ADVI algorithm for posterior approximating

Maxim Kochurov

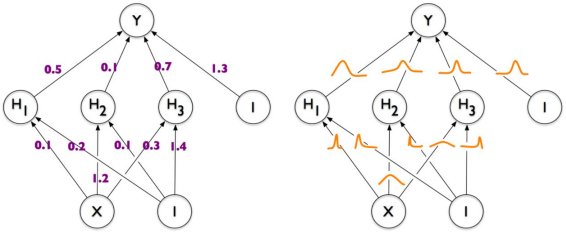
EF MSU

November 15, 2016

Table of Contents

Think Bayes

- Treat weights as distributions
- Choose optimization metrics
- Derive optimization problem
- Choose optimizer



Notations

- Consider we have N independent observations $\mathcal{D} = x_{1:N}$
- We also have k latent variables $\theta = (\theta_1, \ldots, \theta_K)$
- And of course we have some probability model that depends on θ and it relates our $\mathcal{D}$ and θ with likelihood $p(\mathcal{D}|\theta)$
- According to bayesian approach we posit a prior $p(\theta)$ on θ so we have $p(\mathcal{D}, \theta) = p(\mathcal{D}|\theta)p(\theta)$
- We are looking for $p(\theta|\mathcal{D})$ like true bayesians

Model

- We need a differentiable probability model that has continuous $\theta_1, \ldots, \theta_K$
- Then we should be able take gradients $\nabla_\theta \log p(\mathcal{D}, \theta)$ over $supp(p(\theta))$[1]

When we have such model we can start our further investigations and state the optimization problem

[1] $supp(p(\theta)) = \{\theta | \theta \in \mathbb{R}^K \text{ and } p(\theta) > 0\} \subseteq \mathbb{R}^K$

Objective

We need

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)\mathrm{d}\theta}$$

As it is often hard to deriver $p(\theta|\mathcal{D})$ but we can use an approximation $q(\theta|\psi)$. Common objective used for that kind of problem is simplified KL-Divergence

$$\begin{aligned}
KL(q||p) =& \mathbb{E}_{q(\theta|\psi)}\left[\log \frac{q(\theta|\psi)}{p(\theta|\mathcal{D})}\right] = \\
& \mathbb{E}_{q(\theta|\psi)}\left[\log q(\theta|\psi)\right] - \mathbb{E}_{q(\theta|\psi)}\left[\log p(\theta|\mathcal{D})\right] = \\
& \mathbb{E}_{q(\theta|\psi)}\left[\log q(\theta|\psi)\right] - \mathbb{E}_{q(\theta|\psi)}\left[\log \frac{p(\mathcal{D},\theta)}{p(\mathcal{D})}\right] = \\
& \underbrace{\mathbb{E}_{q(\theta|\psi)}\left[\log q(\theta|\psi)\right] - \mathbb{E}_{q(\theta|\psi)}\left[\log p(\mathcal{D},\theta)\right]}_{\text{need to minimize(called variation free energy)}} + \underbrace{\mathbb{E}_{q(\theta|\psi)}\left[\log p(\mathcal{D})\right]}_{const}
\end{aligned}$$

Transformation T

So we have our $ELBO = \mathcal{L} = \mathbb{E}_{q(\theta|\psi)}[\log p(\mathcal{D}, \theta)] - \mathbb{E}_{q(\theta|\psi)}[\log q(\theta|\psi)] \rightarrow \max_{\psi}$.

There is one important constraint on $q(\theta|\psi)$

$$supp(q(\theta|\psi)) \subseteq supp(p(\theta|\mathcal{D})) \text{ or } supp(p(\theta))$$

It is about out prior knowledge about θ, we want our beliefs and what we get to have no conflicts The solution is pretty simple:

$$T : supp(p(\theta)) \rightarrow \mathbb{R}^K$$

Applying it to θ we have $\zeta = T(\theta)$ and

$$g(\mathcal{D}, \zeta) = p(\mathcal{D}, T^{-1}(\zeta)) \left|\det J_{T^{-1}}(\zeta)\right|$$

Why not approximating in new coordinate space where all is pretty good?

Gaussian Approximation

We define approximation family in real coordinate space as diagonal Gaussian. It's easy to work with.

$$q(\zeta|\psi) = \mathcal{N}(\zeta|\mu, diag(\sigma^2)) = \prod_{k=1}^{K} \mathcal{N}(\zeta_k|\mu_k, \sigma_k^2)$$

Recall our objective

$$\mathcal{L} = \mathbb{E}_{q(\theta|\psi)} \left[\log q(\theta|\psi)\right] - \mathbb{E}_{q(\theta|\psi)} \left[\log p(\mathcal{D}, \theta)\right]$$

With some transformations it is now

$$\mathcal{L} = \mathbb{E}_{q(\zeta|\psi)} \left[\log p(\mathcal{D}, \zeta) \left|\det J_{T^{-1}}(\zeta)\right|\right] - \mathbb{E}_{q(\zeta|\psi)} \left[\log q(\zeta|\psi) \left|\det J_{T^{-1}}(\zeta)\right|\right]$$

We can't easily take gradients until expectation depends on ψ, so we use reparametrization trick

$$\zeta = \mu + \exp(\omega) \cdot \eta \quad \text{where} \quad \omega = \log(\sigma),\ \eta \sim \mathcal{N}(\eta|0, \mathrm{I})$$

Call $S_{\mu,\omega} : \zeta \to \eta$

Optimization

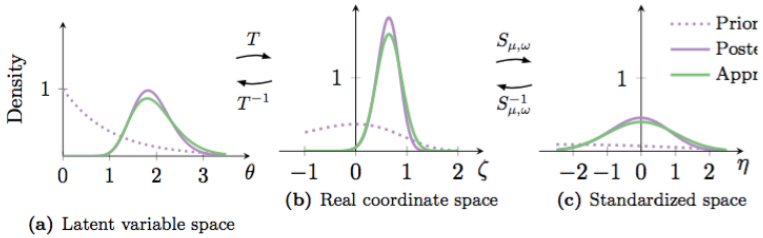


Figure: How it works

Finally

$$\mathcal{L} = \mathbb{E}_{\mathcal{N}(\eta|0,\mathrm{I})}\left[\log p(\mathcal{D}, S^{-1}_{\mu,\omega}(\eta))\left|\det J_{T^{-1}}(S^{-1}_{\mu,\omega}(\eta))\right|\right] - \mathbb{E}_{\mathcal{N}(\eta|0,\mathrm{I})}\left[\log q(S^{-1}_{\mu,\omega}(\eta)|\mu,\omega)\left|\det J_{T^{-1}}(S^{-1}_{\mu,\omega}(\eta))\right|\right] \to \max_{\mu,\omega}$$

Algorithm

Algorithm 1: Automatic Differentiation Variational Inference

Input: Dataset $\mathbf{X} = \mathbf{x}_{1:N}$, model $p(\mathbf{X}, \theta)$.
Set iteration counter $i = 0$ and choose a stepsize sequence $\rho^{(i)}$.
Initialize $\mu^{(0)} = \mathbf{0}$ and $\omega^{(0)} = \mathbf{0}$.

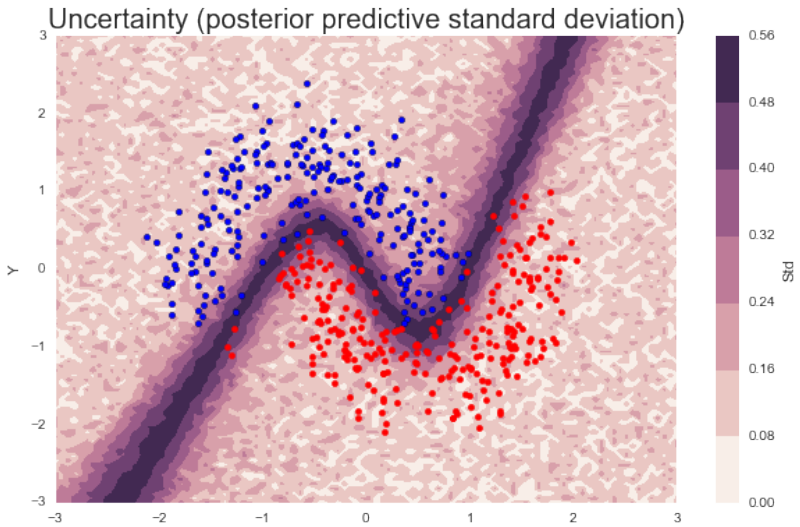
while *change in* ELBO *is above some threshold* **do**

- Draw M samples $\eta_m \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ from the standard multivariate Gaussian.
- Invert the standardization $\zeta_m = \text{diag}(\exp(\omega^{(i)}))\eta_m + \mu^{(i)}$.
- Approximate $\nabla_\mu \mathcal{L}$ and $\nabla_\omega \mathcal{L}$ using MC integration (Equations 5 and 6).
- Update $\mu^{(i+1)} \longleftarrow \mu^{(i)} + \rho^{(i)} \nabla_\mu \mathcal{L}$ and $\omega^{(i+1)} \longleftarrow \omega^{(i)} + \rho^{(i)} \nabla_\omega \mathcal{L}$.
- Increment iteration counter.

end

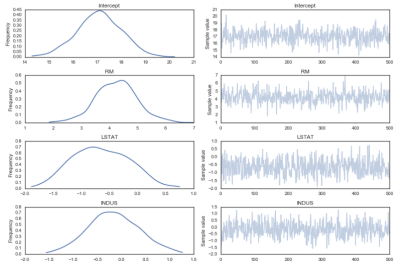
Return $\mu^* \longleftarrow \mu^{(i)}$ and $\omega^* \longleftarrow \omega^{(i)}$.

Toy neural network example (2x5x5x1)



Linear Regression(Boston dataset)

$$y \sim RM + LSTAT + INDUS + NOX + ZN + DIS$$



...

ZN:

Mean	SD	MC Error	95% HPD interval
0.076	0.444	0.021	[-0.711, 1.031]

Posterior quantiles:

2.5	25	50	75	97.5
-0.824	-0.233	0.071	0.382	0.947