Bayesian Neural Networks

Kiana Asgari

Deep Learning

July 15, 2021

Introduction

Bayesian Inference

Bayesian Neural Network

Bayesian Inference Algorithms

Evaluating

All credit goes to the original paper: LAURENT VALENTIN JOSPIN, Hands-on Bayesian Neural Networks - a Tutorial for Deep Learning Users-2020.

Why Bayesian?

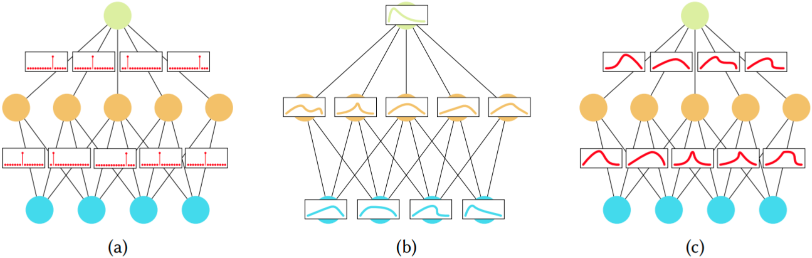
- Bayesian statistics is named after Thomas Bayes as a specific use of Bayes' theorem in 1763.
- Bayesian methods are tempting; owing to their great generality
- using Bayesian methods in deep neural network can help resolving overfitting and overconfidence problems.

How does Bayesian Inference work in theory?

- Baysian methods use the prior blieves as a prior distribution on the set of parameters.
- After receiving iid samples, we can compute posterior distribution using Bayes' formula:

$$p(\boldsymbol{\theta}|D) = \frac{p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta}')p(\boldsymbol{\theta}')d\boldsymbol{\theta}'} \propto p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta}).$$

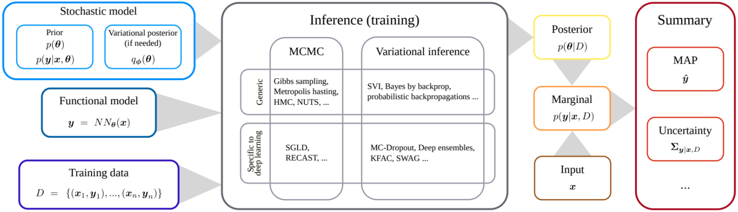
Stochastic neural networks



Bayesian neural network

- A Bayesian Neural Network (BNN) is any stochastic neural network trained using Bayesian inference.

$$\begin{aligned} \boldsymbol{\theta} &\sim p(\boldsymbol{\theta}), \\ \boldsymbol{y} &= NN_{\theta}(\boldsymbol{x}) + \boldsymbol{\epsilon}, \end{aligned}$$


Stochastic model
Prior
$p(\boldsymbol{\theta})$
$p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})$
Variational posterior
(if needed)
$q_{\phi}(\boldsymbol{\theta})$
Functional model
$\boldsymbol{y} = NN_{\boldsymbol{\theta}}(\boldsymbol{x})$
Training data
$D = \{(\boldsymbol{x}_1,\boldsymbol{y}_1),...,(\boldsymbol{x}_n,\boldsymbol{y}_n)\}$
Inference (training)
MCMC
Variational inference
Generic
Gibbs sampling, Metropolis hasting, HMC, NUTS, ...
SVI, Bayes by backprop, probabilistic backpropagations ...
Specific to deep learning
SGLD, RECAST, ...
MC-Dropout, Deep ensembles, KFAC, SWAG ...
Posterior
$p(\boldsymbol{\theta}|D)$
Marginal
$p(\boldsymbol{y}|\boldsymbol{x},D)$
Input
$\boldsymbol{x}$
Summary
MAP
$\hat{\boldsymbol{y}}$
Uncertainty
$\boldsymbol{\Sigma}_{\boldsymbol{y}|\boldsymbol{x},D}$
...

How can one sample from the posterior distribution?

$$p(\boldsymbol{\theta}|D) = \frac{p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta}')p(\boldsymbol{\theta}')d\boldsymbol{\theta}'} \propto p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta}).$$

- computing the exact posterior is intractable due to high dimensionality of the denominator.
- there are several algorithms to sample from posterior without actually computing it.

Markov chain Monte Carlo

Algorithm 1 Metropolis-Hasting

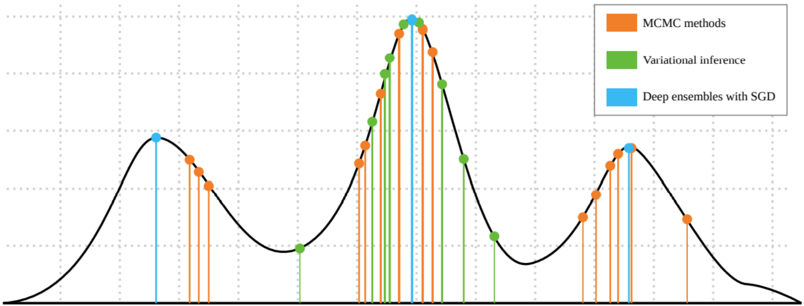
Draw $\boldsymbol{x}_0 \sim \mathit{Initial}$
while $n = 0$ **to** N **do**
  Draw $\boldsymbol{x}' \sim Q(\boldsymbol{x}|\boldsymbol{x}_n)$
  $p = min\left(1, \frac{Q(\boldsymbol{x}'|\boldsymbol{x}_n)}{Q(\boldsymbol{x}_n|\boldsymbol{x}')}\frac{f(\boldsymbol{x}')}{f(\boldsymbol{x}_n)}\right)$
  Draw $k \sim Bernoulli(p)$
  if k **then**
    $\boldsymbol{x}_{n+1} = \boldsymbol{x}'$
    $n = n + 1$
  end if
end while

Variational Inference

- Rather than sampling from the exact posterior, we use the variational distribution parametrized by a set of parameters .
- measure of closeness most readily used is the KL-divergence.
- maximizing the ELBO:

$$ELBO = \int_H q_\phi(H') log\left(\frac{P(H', D)}{q_\phi(H')}\right) dH' = log(P(D)) - D_{KL}(q_\phi || P).$$

Evaluating


MCMC methods
Variational inference
Deep ensembles with SGD

REFERENCES

[1] LAURENT VALENTIN JOSPIN, Hands-on Bayesian Neural Networks - a Tutorial for Deep Learning Users
[2]Nicolas Chopin, ON SOME RECENT ADVANCES ON HIGH DIMENSIONAL BAYESIAN STATISTICS