[Paper review 17]

Dropout as Bayesian Approximation :

Representing Model Uncertainty in Deep Learning

(Yarin Gal, Zoubin Ghahramani, 2016)

[Contents]

0. Abstract

Bayesian models catches model uncertainty

contribution : "model uncertainty with drop out NNs"

- casting Dropout training in NNs as approximate Bayesian inference in deep Gaussian case
- without sacrificing either computational complexity or test accuracy

1. Introduction

Standard deep learning : don't catch model uncertainty

(softmax output are NOT model confidence!)

Passing a distribution through softmax better reflects classification uncertainty

Using dropout in NN

- can be interpreted as a Bayesian approximation of a well known probabilistic model : GP
- avoid over-fitting
- dropout approximately integrates over the models' weight

2. Related Research

- infinitely wide NN = GP (Neal, 1995 & Williams, 1997)
- BNN (Neal, 1995 & Mackay, 1992)
- Variational Inference (Blei et al., 2012)
- Dropout (Blundell et al., 2015)
- Expectation Propagation (Hernandez-Lobato & Adams, 2015)
- Uncertainty estimation on VI approaches (Graves, 2011)

3. Dropout as a Bayesian Approximation

Dropout applied before weight = deep Gaussian Process (Damianou & Lawrence, 2013)

3.1 Dropout

Notation

- $E(\cdot, \cdot)$: error function (softmax loss, Euclidean loss ..)
- $\mathbf{W}_i$: weight matrices of dimension $K_i \times K_{i-1}$,

  (each row of $\mathbf{W}_i$ distribute according to the $p(w)$)
- $\omega = \{\mathbf{W}_i\}_{i=1}^{L}$
- vectors $\mathbf{m}_i$ of dimensions K_i for each GP layer

Objective function

- L_2 regularization (weight decay λ)
- $\mathcal{L}_{\text{dropout}} := \frac{1}{N} \sum_{i=1}^{N} E\left(\mathbf{y}_i, \widehat{\mathbf{y}}_i\right) + \lambda \sum_{i=1}^{L} \left(\|\mathbf{W}_i\|_2^2 + \|\mathbf{b}_i\|_2^2\right)$

Dropout

- With dropout, we sample "Binary Variables" for every input point & for every network unit in each layer
- take value 1 with probability p_i (p_i = 0 : unit is dropped)

3.2 Deep GP

Deep GP

- model distributions over functions
- covariance function : $\mathbf{K}(\mathbf{x}, \mathbf{y}) = \int p(\mathbf{w}) p(b) \sigma\left(\mathbf{w}^T \mathbf{x} + b\right) \sigma\left(\mathbf{w}^T \mathbf{y} + b\right) \mathrm{d}\mathbf{w} \mathrm{d}b$

  (element-wise non-linearity $\sigma(\cdot)$ and distributions $p(\mathbf{w}), p(b)$)

Predictive Probability (of deep GP model)

$p(\mathbf{y} \mid \mathbf{x}, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}) p(\boldsymbol{\omega} \mid \mathbf{X}, \mathbf{Y}) \mathrm{d}\boldsymbol{\omega}$

- $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}) = \mathcal{N}\left(\mathbf{y}; \widehat{\mathbf{y}}(\mathbf{x}, \boldsymbol{\omega}), \tau^{-1} \mathbf{I}_D\right)$
  - $\widehat{\mathbf{y}}(\mathbf{x}, \boldsymbol{\omega}) = \widehat{\mathbf{y}}\left(\mathbf{x}, \boldsymbol{\omega} = \{\mathbf{W}_1, \ldots, \mathbf{W}_L\}\right) = \sqrt{\frac{1}{K_L}} \mathbf{W}_L \sigma\left(\ldots \sqrt{\frac{1}{K_1}} \mathbf{W}_2 \sigma\left(\mathbf{W}_1 \mathbf{x} + \mathbf{m}_1\right) \ldots\right)$
- $p(\boldsymbol{\omega} \mid \mathbf{X}, \mathbf{Y})$: intractable
  - thus, use $q(\omega)$ to approximate

We define $q(\boldsymbol{\omega})$ as

- $\mathbf{W}_i = \mathbf{M}_i \cdot \operatorname{diag}\left(\left[\mathbf{z}_{i,j}\right]_{j=1}^{K_i}\right)$

  where $\mathbf{z}_{i,j} \sim$ Bernoulli (p_i) for $i = 1, \ldots, L, j = 1, \ldots, K_{i-1}$
- meaning of $\mathbf{z}_{i,j}$ = unit j in layer $i-1$ being dropped out as an input to layer i

We minimize KL divergence

- between $q(w)$ and $p(\boldsymbol{\omega} \mid \mathbf{X}, \mathbf{Y})$
- minimize $-\int q(\boldsymbol{\omega}) \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega} + \mathrm{KL}(q(\boldsymbol{\omega}) \| p(\boldsymbol{\omega}))$
- first term
  - $\int q(\boldsymbol{\omega}) \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega} \approx -\sum_{n=1}^{N} \int q(\boldsymbol{\omega}) \log p\left(\mathbf{y}_n \mid \mathbf{x}_n, \boldsymbol{\omega}\right) \mathrm{d}\boldsymbol{\omega}$, where $\widehat{\omega}_n \sim q(\omega)$
- second term
  - $\mathrm{KL}(q(\boldsymbol{\omega}) \| p(\boldsymbol{\omega})) \approx \sum_{i=1}^{L}\left(\frac{p_i l^2}{2}\left\|\mathbf{M}_i\right\|_2^2 + \frac{l^2}{2}\left\|\mathbf{m}_i\right\|_2^2\right)$
- Thus

$$\mathcal{L}_{\mathrm{GP-MC}} \propto \frac{1}{N} \sum_{n=1}^{N} \frac{-\log p\left(\mathbf{y}_n \mid \mathbf{x}_n, \widehat{\boldsymbol{\omega}}_n\right)}{\tau} + \sum_{i=1}^{L}\left(\frac{p_i l^2}{2 \tau N}\left\|\mathbf{M}_i\right\|_2^2 + \frac{l^2}{2 \tau N}\left\|\mathbf{m}_i\right\|_2^2\right)$$

If we set $E\left(\mathbf{y}_n, \widehat{\mathbf{y}}\left(\mathbf{x}_n, \widehat{\boldsymbol{\omega}}_n\right)\right) = -\log p\left(\mathbf{y}_n \mid \mathbf{x}_n, \widehat{\boldsymbol{\omega}}_n\right) / \tau$

$\Rightarrow$ (1)$\mathcal{L}_{\mathrm{GP-MC}}$ = (2) $\mathcal{L}_{\text{dropout}}$

$\Rightarrow$ (1) The sampled $\widehat{\omega}_n$ result in realisations from Bernoulli dist'n $z_{i,j}^n$ = (2) binary variables in the dropout case

4. Obtaining Model Uncertainty

model uncertainty can be obtained from dropout NN

predictive distribution : $q\left(\mathbf{y}^* \mid \mathrm{x}^*\right) = \int p\left(\mathbf{y}^* \mid \mathrm{x}^*, \boldsymbol{\omega}\right) q(\boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega}$

STEPS

- step 1) sample T set of vectors from $\left\{z_1^t, \ldots, z_L^t\right\}_{t=1}^{T}$
- step 2) since $\mathbf{W}_i = \mathbf{M}_i \cdot \operatorname{diag}\left(\left[\mathbf{z}_{i,j}\right]_{j=1}^{K_i}\right)$, find $\left\{\mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right\}_{t=1}^{T}$.
- step 3) MC Dropout : estimate
  - mean : $\mathbb{E}_{q\left(\mathbf{y}^* \mid \mathbf{x}^*\right)}\left(\mathbf{y}^*\right) \approx \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathbf{y}}^*\left(\mathbf{x}^*, \mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right)$
  - second moment : $\mathbb{E}_{q\left(\mathbf{y}^* \mid \mathbf{x}^*\right)}\left(\left(\mathbf{y}^*\right)^T\left(\mathbf{y}^*\right)\right) \approx \tau^{-1} \mathbf{I}_D + \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathbf{y}}^*\left(\mathbf{x}^*, \mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right)^T \widehat{\mathbf{y}}^*\left(\mathbf{x}^*, \mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right)$
  - variance : $\operatorname{Var}_{q\left(\mathbf{y}^* \mid \mathbf{x}^*\right)}\left(\mathbf{y}^*\right) \approx \tau^{-1} \mathbf{I}_D + \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathbf{y}}^*\left(\mathbf{x}^*, \mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right)^T \widehat{\mathbf{y}}^*\left(\mathbf{x}^*, \mathbf{W}_1^t, \ldots, \mathbf{W}_L^t\right) - \mathbb{E}_{q\left(\mathbf{y}^* \mid \mathbf{x}^*\right)}\left(\mathbf{y}^*\right)^T \mathbb{E}_{q\left(\mathbf{y}^* \mid \mathbf{x}^*\right)}\left(\mathbf{y}^*\right)$

    (= (1) sample variance of T stochastic forward passes + (2) inverse model precision)

weight decay : $\lambda \rightarrow$ model precision : $\tau = \frac{p l^2}{2 N \lambda}$

predictive log-likelihood

- predictive likelihood : $p(\mathbf{y} \mid \mathbf{x}, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}) p(\boldsymbol{\omega} \mid \mathbf{X}, \mathbf{Y}) \mathrm{d}\boldsymbol{\omega}$, where $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}) = \mathcal{N}\left(\mathbf{y} ; \widehat{\mathbf{y}}(\mathbf{x}, \boldsymbol{\omega}), \tau^{-1} \mathbf{I}_D\right)$
- predictive log-likelihood : $\log p\left(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{X}, \mathbf{Y}\right) \approx \operatorname{logsumexp}\left(-\frac{1}{2} \tau\left\|\mathbf{y} - \widehat{\mathbf{y}}_t\right\|^2\right) - \log T - \frac{1}{2} \log 2 \pi - \frac{1}{2} \log \tau^{-1}$