[Paper review 3]

Keeping Neural Networks Simple by Minimizing the Description Length of the Weights

(Geoffry E.Hinton and Drew van Camp, 1993)

[Contents]

0. Abstract
1. Introduction
2. Applying the MDL (Minimum Description Length) Principle
3. Coding the data misfit
4. A simple method of coding the weights (= 2. weight penalty)
5. Noisy Weights
6. Summary

0. Abstract

Neural Networks GENERALIZE well , if LESS INFORMATION is in the weights

- keep the weights simple, by penalizing the amount of information they contain!
- add Gaussian noise

Introduce a method of computing...

- 1) derivatives of the expected square loss
- 2) amount of information in the noisy weights

1. Introduction

How to limit the information in the weights (in Neural Network)

- 1) limit the number of connections
- 2) divide the connections into subset & force them within a subset to be identical (= "weight sharing")
- 3) Quantize all the weights, so that a probability mass (p) can be assigned to each quantized value

  (number of bits in a weight = $\log p$)

2. Applying the MDL (Minimum Description Length) Principle

best model is the model that minimizes the combined cost of 1) + 2)

- 1) describing the model
- 2) describing the misfit between model & data

(By adding the discrepancy to the output of the net, receiver can generate exactly the correct output)

3. Coding the data misfit (= 1. train loss)

To apply MDL method,, need to decide coding scheme for data misfits & weights

if data misfits are real number, infinite information is needed $\rightarrow$ Need to quantize (intervals of fixed width t)

particular data misfit : $p\left(d_j^c - y_j^c\right) = t\frac{1}{\sqrt{2\pi}\sigma_j}\exp\left[\frac{-\left(d_j^c - y_j^c\right)^2}{2\sigma_j^2}\right]$

description length of a data misfit (in units of log) :

$-\log p\left(d_j^c - y_j^c\right) = -\log t + \log\sqrt{2\pi} + \log\sigma_j + \frac{\left(d_j^c - y_j^c\right)^2}{2\sigma_j^2}$

find the optimal value of σ_j : $\sigma^* = \sqrt{\frac{1}{N}\sum_c \frac{(d^c - y^c)^2}{\sigma^2}}$

then, $\mathrm{DL}^* = -N\log t + \frac{N}{2}\log\left[\frac{1}{N}\sum_c (d^c - y^c)^2\right] + \frac{N}{2}$

sum over all the training dataset, then

$\therefore$ Data Misfit Cost = $C_{\text{data-misfit}} = kN + \frac{N}{2}\log\left[\frac{1}{N}\sum_c \left(d_j^c - y_j^c\right)^2\right]$

(k only depends on t)

We can see that the description length is minimized by minimizing the usual squared error function

(So, the Gaussian Assumption about coding can be viewed as the MDL justification for this error function!)

4. A simple method of coding the weights (= 2. weight penalty)

code the weights in the same way as above!

description length of the weights is proportional to the sum of their squares (= $\frac{1}{2\sigma_w^2}\sum_{ij} w_{ij}^2$)

∴ Total Description Length : $C = \sum_j \frac{1}{2\sigma_j^2} \sum_c \left(d_j^c - y_j^c\right)^2 + \frac{1}{2\sigma_w^2} \sum_{ij} w_{ij}^2$

(can be seen as just the standard "weight-decay" method)

5. Noisy Weights

How to limit information?

- standard way : add "zero-mean Gaussian Noise"
- MDL framework : allow "very noisy weights" to be communicated very cheaply

5-1. The expected description length of the weights

sender & receiver : have an agreed Gaussian prior = P

sender : has a Gaussian posterior = Q

number of bits required to communicate the posterior distribution of a weight

= asymmetric(KL) Divergence from P to Q

= $G(P, Q) = \int Q(w) \log \frac{Q(w)}{P(w)} du$

5-2. The "bits back" argument

step 1) sender collapses the weights drawn from $Q(w)$ to pick a precise value within the tolerance t

step 2) sender sends each weight for $Q(w)$ (by coding them using $P(w)$) & sends data-misfits

- Communication cost : $C(w) = -\log t - \log P(w)$

  ($\log t$: quantization / $\log P(w)$: for coding w with the prior)

step 3) receiver rcover the exact same posterior $Q(w)$ with correct output & misfits

- # of bits required to collapse weight from Q to a quantized value(w) = $R(w) = -\log t - \log Q(w)$

step 4) True expected description length for a noisy weight :

- $G(P, Q) = \langle C(w) - R(w) \rangle = \int Q(w) \log \frac{Q(w)}{P(w)} dw$

6. Summary

when we have a prior $P(w)$ on weights and the sender has a posterior $Q(w)$,

the expected description length for a noisy (random) weights is $G(P, Q) = \int Q(w) \log \frac{Q(w)}{P(w)} dw$

Goal of lots of variational inference problems (for BNN) :

- "find $Q(w)$ that minimizes $D_{KL}(Q \mid\mid P)$ given some prior $P(w)$"