1 Intorduction Problem Setting

The introduction use polynomial curve fitting as example. We can use the general defined loss function to describe the fitting performance and the normilized RMS loss function.

$$E(\mathbf{w})=\frac{1}{2} \sum_{n=1}^{N}\left\{y\left(x_{n}, \mathbf{w}\right)-t_{n}\right\}^{2}$$

$$E_{\mathrm{RMS}}=\sqrt{2 E\left(\mathbf{w}^{\star}\right) / N}$$

We can use the penalty term to control the model complexity in order to match the problem complexity. (As mentioned in class, the DL model can be overparameterized model, which does not follow the traditional model complexity control theory)

$$\widetilde{E}(\mathbf{w})=\frac{1}{2} \sum_{n=1}^{N}\left\{y\left(x_{n}, \mathbf{w}\right)-t_{n}\right\}^{2}+\frac{\lambda}{2}\|\mathbf{w}\|^{2}$$

2 Probability Theory

[1] This section describe the general probability defintion in descrete conditoin and continuous condition. With Bayes' theorem and the distribution, we can describe the learning processure in probability perspective.

3 The curve fitting revisted in probabilistic perspective

3.1 Piror

Given the value of x, the corresponding value of t has a Gaussian distribution with a mean equal to the value $y(x, \mathbf{w})$

$$p(t \mid x, \mathbf{w}, \beta)=\mathcal{N}\left(t \mid y(x, \mathbf{w}), \beta^{-1}\right)$$

, where precision parameter β corresponding to the inverse variance of the distribution.

If the data point is i.i.d.,

$$p(\mathbf{t} \mid \mathbf{x}, \mathbf{w}, \beta)=\prod_{n=1}^{N} \mathcal{N}\left(t_{n} \mid y\left(x_{n}, \mathbf{w}\right), \beta^{-1}\right),$$

thus we can have the log likelihood,

$$\ln p(\mathbf{t} \mid \mathbf{x}, \mathbf{w}, \beta)=-\frac{\beta}{2} \sum_{n=1}^{N}\left\{y\left(x_{n}, \mathbf{w}\right)-t_{n}\right\}^{2}+\frac{N}{2} \ln \beta-\frac{N}{2} \ln (2 \pi)$$

.

We can remove the term that do not depend on w, and scale the whole equation, then we can minimize the negative log likelihood,

$$\frac{1}{\beta_{\mathrm{ML}}}=\frac{1}{N} \sum_{n=1}^{N}\left\{y\left(x_{n}, \mathbf{w}_{\mathrm{ML}}\right)-t_{n}\right\}^{2}$$

.

For inference, we can have, $p\left(t \mid x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}\right)=\mathcal{N}\left(t \mid y\left(x, \mathbf{w}_{\mathrm{ML}}\right), \beta_{\mathrm{ML}}^{-1}\right)$ and the Prediction distribution, $p(t \mid x, \mathbf{x}, \mathbf{t})=\int p(t \mid x, \mathbf{w}) p(\mathbf{w} \mid \mathbf{x}, \mathbf{t}) \mathrm{d} \mathbf{w}$.

4 Model Selection

cross-validation drawbacks: the training is itself computationally expensive and multiple complexity parameters for a single model.

[1] Limited to the page limit, you can get the full version note at https://github.com/Xiang-Pan/NYU_Baysian_Machine_Learning/blob/master/reading_notes/ch1/note1.pdf

5 The Curse of Dimensionality

We have some observation and assumption for the high-dimensional data, which helps us overcome the curse of dimensionality.

6 Decision Theory

Bayes' theorem, $p\left(\mathcal{C}_k \mid \mathbf{x}\right)=\frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}$. Minimizing the misclassification rate (or maximizing the correct classfication rate) to make the decision. (PS, these two inference methods can be the same at the inference time. However, they MAY be different on the training procedure)

7 Inference and decision

We have three typical ways to describe the inference processure, however, we may prefer those methods which can keep the posterior probability.