Introduction to Statistical Machine Learning
Homework 2

Yota Toyama

November 2, 2016

1.

$$
\begin{aligned}
R(h_r; q) &= \int_{\mathbf{x}} \sum_{c=1}^{C} \sum_{c'=1}^{C} L_{0/1}(c', c) q(c_r = c'|\mathbf{x}) p(\mathbf{x}, y = c) d\mathbf{x} \\
&= \int_{\mathbf{x}} R(h_r|\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \\
\text{where } R(h_r|\mathbf{x}) &= \sum_{c=1}^{C} \sum_{c'=1}^{C} L_{0/1}(c', c) q(c_r = c'|\mathbf{x}) p(y = c|\mathbf{x}) \\
&= \sum_{c=1}^{C} \sum_{c' \neq c}^{C} q(c_r = c'|\mathbf{x}) p(y = c|\mathbf{x}) \\
&= \sum_{c=1}^{C} (1 - q(c_r = c|\mathbf{x})) p(y = c|\mathbf{x}) \\
R(h^*) &= \int_{\mathbf{x}} R(h^*|\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \\
\text{where } R(h^*|\mathbf{x}) &= \sum_{c=1}^{C} L_{0/1}(h^*(\mathbf{x}), c) p(y = c|\mathbf{x}) \\
&= \sum_{c \neq h^*}^{C} p(y = c|\mathbf{x}) \\
&= 1 - p(y = h^*(\mathbf{x})|\mathbf{x}) \\
R(h_r|\mathbf{x}) - R(h^*|\mathbf{x}) &= \sum_{c=1}^{C} (1 - q(c_r = c|\mathbf{x})) p(y = c|\mathbf{x}) - (1 - p(y = h^*(\mathbf{x})|\mathbf{x})) \\
&= p(y = h^*(\mathbf{x})|\mathbf{x}) - \sum_{c=1}^{C} q(c_r = c|\mathbf{x}) p(y = c|\mathbf{x}) \\
&= \sum_{c=1}^{C} q(c_r = c|\mathbf{x}) (p(y = h^*(\mathbf{x})|\mathbf{x}) - p(y = c|\mathbf{x})) \\
&\geq 0 \\
&\therefore\ R(h_r|\mathbf{x}) \geq R(h^*|\mathbf{x}) \\
&\therefore\ R(h_r; q) \geq R(h^*)
\end{aligned}
$$

2. Let M be the number of augmented data points.

$$\sum_{i=1}^{N+M} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 = \sum_{i=1}^{N} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 + \lambda \|\mathbf{w}\|^2$$
$$\sum_{i=N+1}^{N+M} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 = \lambda \|\mathbf{w}\|^2$$

Let $y_i = 0$ and $\mathbf{x}_i = [0, a, \ldots, a]^T$.

$$\sum_{i=N+1}^{N+M} a^2 \|\mathbf{w}\|^2 = \lambda \|\mathbf{w}\|^2$$
$$Ma^2 \|\mathbf{w}\|^2 = \lambda \|\mathbf{w}\|^2$$
$$Ma^2 = \lambda$$
$$\therefore \ \mathbf{y}' = \begin{bmatrix} y_1 \\ \vdots \\ y_N \\ 0 \\ \vdots \\ 0 \end{bmatrix}, X' = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1d} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{N1} & \cdots & x_{Nd} \\ 0 & a & \vdots & a \\ \vdots & \vdots & \vdots & \vdots \\ 0 & a & \vdots & a \end{bmatrix}$$

s.t. $Ma^2 = \lambda$ (where M is the number of augmented data points)

3.

$$\begin{aligned} E_{p(\mathbf{x},y)}\left[y - \mathbf{w}^T\mathbf{x}\right] &= 0 \\ E_{p(\mathbf{x},y)}\left[(y - \mathbf{w}^T\mathbf{x}) E_{p(\mathbf{x})}\left[A\mathbf{x}\right]\right] &= 0 \\ E_{p(\mathbf{x},y)}\left[(y - \mathbf{w}^T\mathbf{x}) A\mathbf{x}\right] - E_{p(\mathbf{x},y)}\left[(y - \mathbf{w}^T\mathbf{x}) E_{p(\mathbf{x})}\left[A\mathbf{x}\right]\right] &= 0 \\ E_{p(\mathbf{x},y)}\left[(y - \mathbf{w}^T\mathbf{x})(A\mathbf{x} - E_{p(\mathbf{x})}\left[A\mathbf{x}\right])\right] &= 0 \end{aligned}$$

$\therefore$ the correration between any linear function of data and prediction errors is 0.

4.

$$
\begin{aligned}
\hat{\mathbf{w}} &= \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 \\
&= (X^T X)^{-1} X^T y
\end{aligned}
$$

$$\text{Let } C \in \mathbb{R}^{(d+1)\times(d+1)} \text{ be a diagonal matrix s.t. } \tilde{X} = XC$$

$$
\begin{aligned}
\hat{\tilde{\mathbf{w}}} &= \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \mathbf{w}^T \tilde{\mathbf{x}}_i)^2 \\
&= (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y \\
&= ((XC)^T XC)^{-1} (XC)^T y \\
&= (CX^T XC)^{-1} CX^T y \\
&= C^{-1} (X^T X)^{-1} C^{-1} C X^T y \\
&= C^{-1} (X^T X)^{-1} X^T y \\
\tilde{X} \hat{\tilde{\mathbf{w}}} &= XCC^{-1} (X^T X)^{-1} X^T y \\
&= X (X^T X)^{-1} X^T y \\
&= X \hat{\mathbf{w}} \text{ as required}
\end{aligned}
$$

5.

$$
\begin{aligned}
\hat{\sigma^2} &= \operatorname*{argmax}_{\sigma^2} \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i; \mathbf{w}, \sigma) \\
&= \operatorname*{argmax}_{\sigma^2} -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 - N \log \sigma \sqrt{2\pi} \\
&= \operatorname*{argmin}_{\sigma^2} \frac{1}{\sigma^2} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 + N \log 2\pi\sigma^2 \\
&= \operatorname*{argmin}_{\sigma^2} \frac{1}{\sigma^2} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 + N \log \sigma^2
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial\sigma^2} \frac{1}{\sigma^2} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 + N \log \sigma^2 &= 0 \\
-\frac{1}{\sigma^4} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 + \frac{N}{\sigma^2} &= 0 \\
\sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 - N\sigma^2 &= 0
\end{aligned}
$$

$$
\begin{aligned}
\sigma^2 &= \frac{1}{N} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2 \\
\therefore \hat{\sigma^2} &= \frac{1}{N} \sum_{i=1}^{N} (y_i - f(\mathbf{x}_i; \mathbf{w}))^2
\end{aligned}
$$

The experimental result on a validation dataset showed that there are huge gaps of loss, σ^2, and log-likelihood between a linear model and, a quadratic and cubic models. The lienar model showed much greater values in terms of loss and σ^2. The others achieved much better log-likelihood values.

Based on the evaluation on a validation dataset, I select a quadratic model as model A. The reasons are listed below.

- Computional efficiency for training and prediction
  There are fewer times of multiplication compared with a cubic one.
- Low complexity
  It has one fewer parameters compared with a cubic one.
- Acceptably low loss value
  While a linear model is more efficient and simpler than a quadratic one, it showed too large loss on both training and validation datasets. That means a linear one is not expressive enough for the data.

6.

$$
\begin{aligned}
\text{Let } \beta_{\hat{y},y} &= \begin{cases} 1 & \text{if } \hat{y} \leq y \\ \alpha & \text{otherwise} \end{cases} \\
l_\alpha(\hat{y}, y) &= \begin{cases} (\hat{y} - y)^2 & \text{if } \hat{y} \leq y \\ \alpha(\hat{y} - y)^2 & \text{otherwise} \end{cases} \\
&= \beta_{\hat{y},y}(\hat{y} - y)^2 \\
\frac{\partial}{\partial \mathbf{w}} L_\alpha &= \frac{\partial}{\partial \mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} l_\alpha(\hat{y}, y) \\
&= \frac{\partial}{\partial \mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} \beta_{\hat{y},y}(\hat{y}_i - y_i)^2 \\
&= \frac{\partial}{\partial \mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} \beta_{\hat{y},y}(\mathbf{w}^T \phi_i(\mathbf{x}) - y_i)^2 \\
&= \frac{1}{N} \sum_{i=1}^{N} 2\beta_{\hat{y},y}(\mathbf{w}^T \phi_i(\mathbf{x}) - y_i)\phi_i(\mathbf{x})
\end{aligned}
$$

Here, I also choose a quadratic model as model B with the exactly same reasons as model A.

While I cannot compare model A and B because I chose the same quadratic one, I think it is not reasonable to compare them because they are evaluated on different tasks with symmetric and asymmetric loss functions. The results of different experiments just explain which model is better on which experiment.