Machine Learning, 2024 Spring
Assignment 7

Notice

Plagiarizer will get 0 points.
LaTeXis highly recommended. Otherwise you should write as legibly as possible.

Problem 1 Referring to Figure 4.10, why are both curves increasing with K? Why do they converge to each other with increasing K? (20pt)

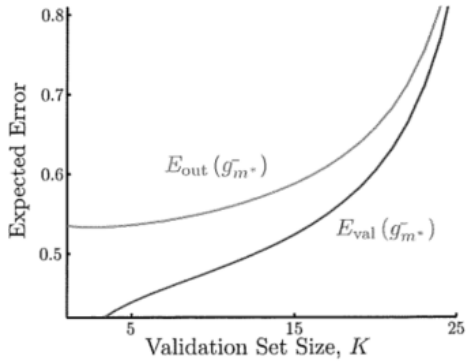

Figure 4.10: Optimistic bias of the validation error when using a validation set for the model selected.

Solution

1. **Why are both curves increasing with** K (10pt): With K increasing, the training set size $N-K$ (N is the number of samples) decreases, therefore, both the validation error $E_{val}(g_{m^*}^-)$ and the true error $E_{out}(g_{m^*}^-)$) increases.
2. **Why do they converge to each other with increasing** K (10pt): Since $E_{out}(g^-) \leq E_{val}(g^-) + \mathcal{O}(\frac{1}{\sqrt{K}})$, when K decreases, the validation error $E_{val}(g_{m^*}^-)$ approaches to true error $E_{out}(g_{m^*}^-)$, two curves converge to each other.

Problem 2

1. From Figure 4.12, $\mathbb{E}[E_{out}(g_{m^*}^-)]$ is initially decreasing. How can this be, if $\mathbb{E}[E_{out}(g_{m^*}^-)]$ is increasing in K for each m? (10pt)
2. From Figure 4.12 we see that $\mathbb{E}[E_{out}(g_{m^*})]$ is initially decreasing, and then it starts to increase. What are the possible reasons for this? (10pt)
3. When $K = 1$, $\mathbb{E}[E_{out}(g_{m^*}^-)] < \mathbb{E}[E_{out}(g_{m^*})]$. How can this be, if the learning curves for both models are decreasing? (10pt)

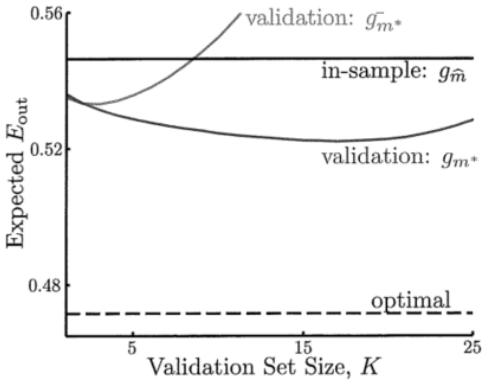

Figure 4.12: Model selection between $\mathcal{H}_2$ and $\mathcal{H}_5$ using a validation set. The solid black line uses E_{in} for model selection, which always selects $\mathcal{H}_5$. The dotted line shows the optimal model selection, if we could select the model based on the true out-of-sample error. This is unachievable, but a useful benchmark. The best performer is clearly the validation set, outputting g_{m^*}. For suitable K, even $g_{m^*}^-$ is better than in-sample selection.

Solution

1. In the beginning, K is small, so the validation effect is weak. It is hard to choose the best model based on the validation error with huge bias. With K growing big, the validation error is able to better reflect the ability of the model. Therefore, the plot increases at first. Then with the K increase, the size of training set $N - K$ decreases, the model has not been trained enough. Therefore, the $\mathbb{E}[E_{out}(g_{m^*}^-)]$ decreases afterwards.
2. g_{m^*} is trained with all data while $g_{m^*}^-$ is trained with $N - K$ data. When K is small, the validation error $g_{m^*}^-$ is close to g_{m^*} since the difference between the training set is small. The tendency of g_{m^*} is similar to the tendency of $g_{m^*}^-$.
3. When the K is small, E_{in} has large bias. Therefore, the $E_{out}[g_{m^*}^-]$ may be bigger than $E_{out}[g_{m^*}]$.

Problem 3

Definition 1 (leave-one-out cross-validation) *Select each training example in turn as the single example to be held-out, train the classifier on the basis of all the remaining training examples, test the resulting classifier on the held-out example, and count the errors.*

Let the superscript -i denote the parameters we would obtain by finding the SVM classifier f without the ith training example. Define the *leave-one-out CV error* as

$$\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i, f(\mathbf{x}_i; \mathbf{w}^{-i}, b^{-i})), \tag{1}$$

where $\mathcal{L}$ is the zero-one loss. Prove that

$$\text{leave-one-out CV error} \leq \frac{\text{number of support vectors}}{\text{n}} \tag{2}$$

(20pt)

Solution

According to this problem, we assume that there are k support vectors, and there are $n-k$ non-support vectors. For non-support vectors, the *leave-one-out CV error* is 0, so we only need to consider the *leave-one-out CV error* of support vectors.

For the support vectors, we can get

$$\mathcal{L}(y_i, f(\mathbf{x}_i; \mathbf{w}^{-i}, b^{-i})) \leq 1, 1 \leq i \leq k$$

In this way,

$$\sum_{i=1}^{n}\mathcal{L}(y_i, f(\mathbf{x}_i; \mathbf{w}^{-i}, b^{-i})) \leq k.$$

Therefore,

$$\text{leave-one-out CV error} \leq \frac{\text{number of support vectors}}{\text{n}}.$$

Problem 4

The l_1-norm SVM can be formulated as follows

$$\begin{aligned} &\min_{\mathbf{w},b} \|\mathbf{w}\|_1 \\ &\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, i = 1, ..., n. \end{aligned} \tag{3}$$

Please deprive the equivalent linear programming formulation of (3) (10pt), give its dual formulation (10pt). Also, please explain how to determine the support vector SV according to the optimal multiplier. (10pt)

Solution

1. **The equivalent linear programming formulation is:**

$$\begin{aligned} &\min_{\mathbf{w},b} \sum_{i=1}^{n} u_i \\ &\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, i = 1, ..., n. \\ &\quad -w_i \leq u_i, w_i \leq u_i, i = 1, ..., n. \end{aligned} \tag{4}$$

2. **Dual formulation**:

$$\begin{aligned} L(\mathbf{w}, b, \mathbf{u}, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\mu}) &= \sum_{i=1}^{n} u_i - \sum_{i=1}^{n} \alpha_i(y_i(\mathbf{w}^T x_i + b) - 1) - \sum_{i=1}^{n} \xi_i(u_i - w_i) - \sum_{i=1}^{n} \mu_i(u_i + w_i), \\ \frac{\partial L}{\partial w_i} &= -\sum_{i=1}^{n} \alpha_i y_i x_i + \xi_i - \mu_i = 0, \\ \frac{\partial L}{\partial b} &= -\sum_{i=1}^{n} \alpha_i y_i = 0, \\ \frac{\partial L}{\partial u_i} &= 1 - \xi_i - \mu_i = 0. \end{aligned} \tag{5}$$

   Then we can get the dual problem,

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i \\ &\text{s.t. } 1 - \sum_{i=1}^{n} \alpha_i y_i x_i = 0, i = 1, ..., n. \\ &\quad \sum_{i=1}^{n} \alpha_i y_i = 0, i = 1, ..., n. \\ &\quad \alpha_i \leq 0, i = 1, ..., n. \end{aligned} \tag{6}$$

3. **How to determine the support vector SV according to the optimal multiplier**: The sample points corresponding to $\alpha_i > 0$ are the support vectors.