Recap: cross validation:

↳ For $n = 1, 2, \cdots N$

$$D_n = \{ (\underline{x}_1, y_1) \dots \cancel{(\underline{x}_n, y_n)} \dots (\underline{x}_N, y_N) \}$$

$$D_n \rightarrow \boxed{A} \rightarrow g_n^-$$

$$e_n = e(g_n^- (\underline{x}_n), y_n)$$

$$E_{cv} = \frac{1}{N} \sum_{n=1}^{N} e_n$$

↳ Question: is $E_{cv} = E_{out}$?

1. is $E_{cv} = E_{out}$?

↳ Note: // $D = D_n \cup \{ (\underline{x}_n, y_n) \}$

① $$\mathbb{E}_D [E_{cv}] = \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_D [e(g_n^- (x_n), y_n)]$$

$$= \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{D_n} [\mathbb{E}_{\underline{x}_n} [e(g_n^- (\underline{x}_n), y_n) \mid D_n]]$$

$$= \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{D_n} [\mathbb{E}_{\underline{x}} [e(g_n^- (\underline{x}_n), y)]]$$

$$= \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{D_n} [E_{out} (g_n^-)]$$

$$\overline{E_{out}} (N) \triangleq \mathbb{E}_D [E_{out} (g)]$$

expected test error over the randomness of a set of N training samples

Then $$\mathbb{E}_D [E_{cv}] = \frac{1}{N} \sum_{n=1}^{N} \overline{E_{out}} (N-1)$$

$$= \overline{E_{out}} (N-1)$$

∴ E_{cv} is an unbiased estimate of $\overline{E_{out}} (N-1) \approx \overline{E_{out}} (N)$

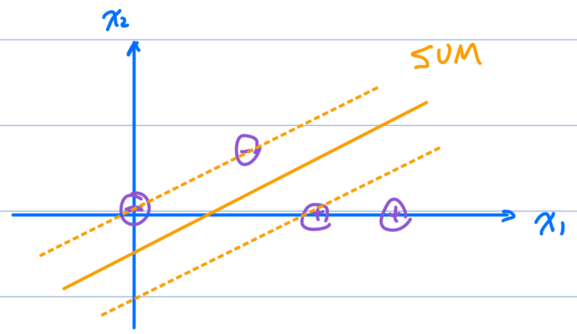
③ However, since $\{e_n\}$ are not independent.

$var[E_{cv}] \neq \frac{1}{N} var[e_n]$

In practice. E_{cv} is small,

$\therefore E_{cv} \approx \overline{E_{out}}(N)$ // mean vaule

e.g. $d=2$



using LOO cross validation,

$E_{cv} = \frac{1}{N} \sum_{n=1}^{N} e_n$

- If (x_n, y_n) is not a support vector, D_n produces the same solution as D does $\Rightarrow e_n = 0$
- If (x_n, y_n) is a support vector,

$e_n = 1(y_n \neq g_n^-(x_n)) \leq 1$

$\therefore E_{cv} \leq \frac{\text{No. of support vectors}}{N}$ — usually small

— indep. with decision d.

2. Problem of LOO:

↳ Total computation : $N \times$ training

↳ V-fold cross validation:

- Given $\mathcal{D}$, split into V equally sized sets

$$\mathcal{D} = \underset{N/V}{\mathcal{D}_1} \cup \underset{N/V}{\mathcal{D}_2} \cup \cdots \cup \underset{N/V}{\mathcal{D}_V}$$

samples

- Let $D_m^{train} = D_1 \cup D_2 \cup \cdots \cancel{D_m} \cdots \cup D_N$

  Use D_m^{train} to train model and obtain g_m^-

  Let $e_m = \frac{V}{N} \sum_{(x_n, y_n) \in D_m} e(g_m^-(x_n), y_n)$

$$E_{cv} = \frac{1}{V} \sum_{n=1}^{V} e_m \approx E_{out}$$