Assignment 8

Darwin Ding

October 27, 2016

Exercise 4.3

(a) If H is fixed, and we increase the complexity of f, deterministic noise will go up. This happens because there is more of f that we cannot model with our hypothesis set. More noise means more overfitting. A higher target complexity also means a higher chance of overfitting, so there is a **higher tendency to overfit**.

(b) If f is fixed and we decrease the complexity of H, deterministic noise will also go up, for the same reason as above. There will be more aspects of the target function that we cannot fit with a simpler hypothesis set. However, the fact that we are simplifying the hypothesis set also means that the chance of overfitting will go down. These are two competing factors, but in practice we will find that the amount that the latter affects the tendency to overfit will strongly outweigh the amount that the former affects the tendency to overfit. Thus, **lower tendency to overfit**.

Exercise 4.5

(a) If Γ is the **identity matrix**, then $w^T \Gamma^T \Gamma w = w^T w = \sum_{q=0}^{Q} w_q^2$, and constraining the original expression to be $\leq C$ will constrain the summation of all the individual sums to be $\leq C$.

(b) This can be done by making the Tikhonov regularization constant an N x N matrix with the top row all 1s and the rest of the values 0, like follows:

$$\begin{bmatrix} 1 & 1 & 1 & ... & 1 \\ 0 & 0 & 0 & ... & 0 \\ & & ... & & \\ 0 & 0 & 0 & ... & 0 \end{bmatrix}$$

We can then figure out this expression as follows:

$$w^T\Gamma^T\Gamma w$$
$$= (w^T\Gamma^T)(\Gamma w)$$

This is a legal operation with matrix multiplication, as long as we preserve the original order of matrices. Γ^T is the N x N matrix with the first column being 1 and rest of values being 0. w^T is a 1 x N matrix. When multiplying these two matrices together we will get:

$$\begin{bmatrix} (w_0 + w_1 + w_2 + ...) & 0 & 0 & ... & 0 \end{bmatrix}$$

We can also resolve the other side the same way. Γ is the N x N matrix with the first row being 1 and the rest of values 0. w is a N x 1 matrix. We will then get:

$$\begin{bmatrix} (w_0 + w_1 + w_2 + ...) \\ 0 \\ 0 \\ ... \\ 0 \end{bmatrix}$$

Then, multiplying both of these matrices together we will get a 1 x 1 matrix that looks like follows:

$$(w_0 + w_1 + w_2 + ...)^2$$
$$= (\sum_{q=0}^{Q} w_q)^2$$

Exercise 4.6

Hard-order constraints are more useful for binary classification. In binary classification, it doesn't matter how far away points are from the perceptron line, as long as they are on the correct side of the line. While a linear hard-order constraint will have a d_{VC} of N + 1, where N is the dimension of the perceptron, a soft-order constraint will have a VC constrained by C, which is a much larger bound in practice.

Additionally, a perceptron in N dimensions is typically represented by the formula $0 = w_0x^N + w_1x^{N-1}y + ... + w_Ny^N$, and to satisfy a soft-order constraint it would be easy to scale down all the weights by a constant factor and not change the positioning of the perceptron.

Exercise 4.7

(a)

$$\begin{aligned}
\sigma^2_{val} &= Var_{D_{val}}[E_{val}(g^-)] \\
&= Var_{D_{val}}[\frac{1}{K}\sum_{x\in D_{val}} e(g^-(x_n), y_n)] \\
&= \frac{1}{K}Var_{D_{val}}[\sum_{x\in D_{val}} e(g^-(x_n), y_n)] \\
&= \frac{1}{K}Var_x[e(g^-(x), y)] \\
&= \frac{1}{K}\sigma^2(g^-)
\end{aligned}$$

(b) For any point n, we know that the probability that g^- classifies it incorrectly is $P[g^-(x) \neq y]$. For binary classification, $e(g^-(x), y)) = 1$ if $g^-(x) \neq y$ and 0 otherwise. Therefore, $\mathbf{E}[e(g^-x, y)] = P[g^-(x) \neq y]$. Another thing to note before going forward is that $\mathbf{E}[e^2] = \mathbf{E}[e]$, since $0^2 = 0$ and $1^2 = 1$.

$$\begin{aligned}
\sigma^2_{val} &= \frac{1}{K}Var_x[e(g^-(x), y)] \\
&= \frac{1}{K}(\mathbf{E}[e^2] - \mathbf{E}[e]^2) \\
&= \boldsymbol{\frac{1}{K}(P[g^-(x) \neq y] - P[g^-(x) \neq y]^2)}
\end{aligned}$$

(c) We can prove this through analysis. Probabilities are strictly ≤ 1 and ≥ 0. First, let's try to maximize $P - P^2$.

$$P - P^2$$

$$\frac{d(P - P^2)}{dP} = 1 - 2P$$

Setting the derivative equal to 0 shows that the maximum difference between a probability and its square occurs at 0.5. $0.5 - 0.5^2 = 0.25$, so we can plug this into the formula:

$$\begin{aligned} \sigma_{val}^2 &= \frac{1}{K}(P[g^-(x) \neq y] - P[g^-(x) \neq y]^2) \\ &= \frac{1}{K}(0.25) \\ &= \frac{1}{4K} \end{aligned}$$

And this is the maximum possible value, so we can conclude that:

$$\sigma_{val}^2 \leq \boldsymbol{\frac{1}{4K}}$$

(d) **No**, there isn't an upper bound, because there is no theoretical bound for squared error. As a result, there is no theoretical bound for variance either, and you can't do the convenient math that we could for binary classification.

(e) **Higher**. If you train with less points, we would expect the regression output to be less accurate. This would naturally imply a generally higher squared error, which would imply a higher variance, thus a higher $\sigma^2(g^-)$.

(f) As a result, if you have a larger validation set, you will typically lose out on your ability to estimate E_{out}. Having a larger validation set means higher variance, after all.

Exercise 4.8

E_m is indeed an unbiased estimate for the out-of-sample error. Just because you have multiple models does not make it biased, as the validation set has had nothing to do with the actual training of any particular hypothesis.