LFD Problem Set 8

John Cohen

November 4, 2024

Exercise 4.3

Deterministic noise depends on H, as some models approximate f better than others.

(a) Assume H is fixed and we increase the complexity of f. Will deterministic noise in general go up or down? Is there a higher or lower tendency to overfit?

Deterministic noise will **INCREASE**. Deterministic noise depends on the hypothesis set as it does not exactly represent the target function. A more complex f means the gap between what H can represent and f also grows. This causes the approximation error to grow and with it deterministic noise. There is also a **LOWER** tendency to overfit which arises when H is more complex than f, not the other way around as we are claiming in this situation.

(b) Assume f is fixed and we decrease the complexity of H. Will deterministic noise in general go up or down? Is there a higher or lower tendency to overfit? *[Hint: There is a race between two factors that affect overfitting in opposite ways, but one wins.]*

The two factors are increase in overfitting from the higher deterministic noise and decrease in overfitting from lower complexity of H reducing the ability to fit random noise in the training data. The winning factor is the decrease in overfitting as the reduction in model complexity has a stronger effect in decreasing overfitting than deterministic noise has in harming generalization. In the case of (b) decreasing the complexity of H while holding f fixed causes deterministic noise to **INCREASE** but still a **LOWER** tendency of overfitting as H becomes less capable of fitting the data and the noise in the data despite the higher deterministic noise.

Exercise 4.5 [Tikhonov regularizer]

A more general soft constraint is the Tikhonov regularization constraint

$$w^T \Gamma^T \Gamma w \leq C$$

which can capture relationships among the w_i (the matrix Γ is the Tikhonov regularizer).

(a) What should Γ be to obtain the constraint $\Sigma_{q=0}^{Q} w_q^2 \leq C$.

$\Sigma_{q=0}^{Q} w_q^2 = \Sigma_{q=0}^{Q} w_q * w_q = w^T * w$. Therefore, for $w^T * w = w^T \Gamma^T \Gamma w$, $\Gamma = I$.

(b) What should Γ be to obtain the constraint $(\Sigma_{q=0}^{Q} w_q)^2 \leq C$?

Similarly to part (a) $(\Sigma_{q=0}^{Q} w_q)^2 = (\Sigma_{q=0}^{Q} w_q)(\Sigma_{q=0}^{Q} w_q) = w^T * w$ and thus $\Gamma = I$.

Exercise 4.6

We have seen both the hard-order constraint and the soft-order constraint. Which do you expect to be more useful for binary classification using the perceptron model? [Hint: $\text{sign}(w^T x) = \text{sign}(\alpha w^T x)$ for any $\alpha > 0$.]

The hard-order constraint is more useful because limiting features directly affects the perceptron's decisions, while soft-order constraints (like penalizing weight sizes) don't influence its predictions—since scaling weights doesn't change the output sign. With soft order, penalizing the magnitude of w does not affect the classification outcome; scaling w does not affect the perceptron's predictions as the hint suggests. It is less effective while hard order is more effect directly impacting the perceptron's decision boundary. The constraint does not matter for this binary classification.

Exercise 4.7

Fix g^- (learned from D_{train} and define $\sigma_{val}^2 = Var_{D_{val}}[E_{val}(g^-)]$. We consider how σ_{val}^2 depends on K. Let

$$\sigma_{val}^2(g^-) = Var_x[e(g^-(x), y)]$$

be the point-wise variance in the out-of-sample error of g^-.

(a) Show that $\sigma_{val}^2 = \frac{1}{K}\sigma^2(g^-)$.

$$\sigma_{val}^2 = \text{Var}_{D_{train}}[E_{\text{Val}}(g^-)]$$

$$\sigma_{val}^2 = \text{Var}_{D_{train}}[\frac{1}{K}\Sigma_{x_n \in D_{val}} e(g^-(x_n), y_n)]$$

$$\sigma_{val}^2 = \frac{1}{K^2}\Sigma_{x_n \in D_{val}} \text{Var}_{x_n} e(g^-(x_n), y_n)$$

$$\sigma^2_{val} = \frac{1}{K^2} K \sigma^2(g^-)$$

$$\sigma^2_{val} = \frac{\sigma^2(g^-)}{K}$$

(b) In a classification problem, where $e(g^-(x), y) = [g^-(x) \neq y]$, express σ^2_{val} in terms of $P[g^-(x) \neq y]$.
Let $e(g^-(x), y) = [g^-(x) \neq y] = 1$ and $e(g^-(x), y) = [g^-(x) = y] = 0$. Here we have:

$$E_x[e(g^-(x), y)] = P[g^-(x) \neq y]$$

$$\sigma^2(g^-) = \text{Var}_x[e(g^-(x), y)]$$

$$\sigma^2(g^-) = E_x[e(g^-(x), y) - E_x[e(g^-(x), y)]]^2$$

$$\sigma^2(g^-) = E_x[e(g^-(x), y) - P[g^-(x) \neq y]]^2$$

$$\sigma^2(g^-) = (1 - P[g^-(x) \neq y]) * P^2[g^-(x) \neq y] + P[g^-(x) \neq y] * (1 - P[g^-(x) \neq y])^2$$

$$\sigma^2(g^-) = (1 - P[g^-(x) \neq y]) * P[g^-(x) \neq y]$$

$$\frac{\sigma^2(g^-)}{K} = \frac{(1 - P[g^-(x) \neq y]) * P[g^-(x) \neq y]}{K}$$

Substitute from part (a):

$$\sigma^2_{val} = \frac{(1 - P[g^-(x) \neq y]) * P[g^-(x) \neq y]}{K}$$

(c) Show that for any g^- in a classification problem, $\sigma^2_{val} \leq \frac{1}{4K}$.
Take the equation $x(1-x)$. The largest value this equation can be is $x(1-x)\frac{d}{dx} = 0 \implies 1 - 2x = 0 \implies x = 0.5$. At $x = 0.5$ $x(1-x) = 0.5 * 0.5 = \frac{1}{4}$ as a maximum. In summary, $x(1-x) \leq \frac{1}{4}$. We can apply the same principle to $\sigma^2_{val} = \frac{(1-P[g^-(x)\neq y])*P[g^-(x)\neq y]}{K}$ from part (b) where x in this case is $P[g^-(x) \neq y]$. We get

$$\sigma^2_{val} = \frac{(1 - P[g^-(x) \neq y]) * P[g^-(x) \neq y]}{K} \leq \frac{1}{4K}$$

(d) Is there a uniform upper bound for $Var[E_{val}(g^-)]$ similar to (c) in the case of regression with squared error $e(g^-(x), y) = (g^-(x) - y)^2$? *[Hint: The squared error is unbounded.]*

No, there is no upper bound for $\text{Var}[E_{val}(g^-)] = E[(e(g^-(x), y))^2] - E[e(g^-(x), y)]^2$ in the case of regression with squared error $e(g^-(x), y) = (g^-(x) - y)^2$. Because the squared error $(g^-(x), y)^2$ is unbounded, the variance is also unbounded.

(e) For regression with squared error, if we train using fewer points (smaller N-K) to get g^-, do you expect $\sigma^2(g^-)$ to be higher or lower? *[Hint: For continuous, non-negative random variables, higher mean often implies higher variance.]*

If we train a regression model on less data points, we should expect $\sigma^2(g^-)$ being the variance of the average hypothesis g^- to be higher. g will produce a worse approximation bounded by a larger number. The squared error will be greater along with the variance.

(f) Conclude that increase the size of the validation set can result in a better or a worse estimate of E_{out}.

Either. Increasing the validation set reduces variance in the error estimate ($\sigma^2_{val} = \frac{\sigma^2(g)}{K}$) giving a better estimate of E_{out}. However, it worsens g and thus the approximation of E_{out} by reducing the training data for the increase in the validation set. The overall effect depends on these two factors.

Larger validation set $\implies$ better E_{val} $\implies$ better E_{out} estimate
Smaller training set $\implies$ worse g $\implies$ Higher E_{out}

Exercise 4.8

Is E_m an unbiased estimate for the out-of-sample error $E_{out}g_m^-$?

g_m was built from an independent training set and completely disjoint from the validation set. Therefore, E_m, being $E_{val}(g_m^-)$, IS an unbiased estimate for the out-of-sample error, $E_{out}g_m^-$.

Problem 4.26

In this problem, derive the formula for the exact expression for the leave-one-out cross validation error for linear regression. Let Z be the data matrix whose rows correspond to the transformed data points $z_n = \Phi(x_n)$.

(a) Show that:

$$Z^TZ = \Sigma_{n=1}^N z_n z_n^T; Z^Ty = \Sigma_{n=1}^N z_n y_n; H_{nm}(\lambda) = z_n^T A^{-1}(\lambda) z_m,$$

where $A = A(\lambda) = Z^TZ + \lambda\Gamma^T\Gamma$ and $H(\lambda) = ZA(\lambda)^{-1}Z^T$. Hence, show that when (z_n, y_n) is left out, $Z^TZ \implies Z^TZ - z_nz_n^T$, and $Z^Ty \implies Z^Ty - z_ny_n$.

The all three equations are trivial and follow the definition of the dot product.

(b) Compute w_n^-, the weight vector learned when the nth data point is left out, and show that:

$$w_n^- = (A^{-1} + \frac{A^{-1}z_n z_n^T A^{-1}}{1 - z_n^T A^{-1} z_n})(Z^T y - z_n y_n).$$

[Hint user the identity $(A - xx^T)^{-1} = A^{-1} + \frac{A^{-1}xx^T A^{-1}}{1 - x^T A^{-1} x}$*.]*

The weight vector learned without the n-th data point is:

$$w_n^- = A_n^{-1}(Z^T y - z_n y_n)$$

With the hint and z_n acting as x,

$$A_n^{-1} = (A - z_n z_n^T)^{-1} = A^{-1} + \frac{A^{-1}z_n z_n^T A^{-1}}{1 - z_n^T A^{-1} z_n}$$

Substituting A^{-1} into the previous equations,

$$w_n^- = (A^{-1} + \frac{A^{-1}z_n z_n^T A^{-1}}{1 - z_n^T A^{-1} z_n})(Z^T y - z_n y_n)$$

(c) Using (a) and (b), show that $w_n^- = w + \frac{y_n^- - y_n}{1 - H_{nm}} A^{-1} z_n$, where w is the regression weight vector using all the data.

$$w_n^- = A^{-1}(Z^T y - z_n y_n) + (\frac{A^{-1}z_n z_n^T A^{-1}}{1 - H_{nm}})(Z^T y - z_n y_n)$$

$$A^{-1}(Z^T y - z_n y_n) = A^{-1}Z^T y - A^{-1} z_n y_n = w - A^{-1} z_n y_n$$

$$z_n^T A^{-1} z_n y_n = H_{nm} y_n$$

$$(\frac{A^{-1}z_n z_n^T A^{-1}}{1 - H_{nm}})(Z^T y - z_n y_n) = \frac{A^{-1}z_n (w^T z_n - H_{nm} y_n)}{1 - H_{nm}}$$

$$w_n^- = w - A^{-1} z_n y_n + \frac{A^{-1}z_n (w^T z_n - H_{nm} y_n)}{1 - H_{nm}}$$

$$w_n^- = w + A^{-1} z_n (\frac{w^T z_n - y_n}{1 - H_{nm}})$$

$$w^T z_n - y_n = y_n^- - y_n$$

$$w_n^- = w + A^{-1} z_n (\frac{y_n^- - y_n}{1 - H_{nm}})$$

or

$$w_n^- = w + \frac{y_n^- - y_n}{1 - H_{nm}} A^{-1} z_n$$

(d) The prediction on the validation point is given by $z_n^T w_n^-$. Show that

$$z_n^T w_n^- = \frac{y_n^- - H_{nm} y_n}{1 - H_{nm}}$$

(e) Show the $e_n = (\frac{y_n^- - y_n}{1 - H_{nm}})^2$, and hence prove Equation (4.13).

Equation 4.13) $E_{cv} = \frac{1}{N} \Sigma_{n=1}^N (\frac{y_n^- - y_n}{1 - H_{nm}(\lambda})^2$