

Discussion 5

This week's resources:

- StatQuest videos on Logistic Regression videos
- Claude discovers a new enzyme system link
- Jev: a new model architecture built to make decisions link
- Jev plays Catan link

1 Bias-Variance Trade-off

As we saw in lecture, the expected error for a model f_w at a test point x can be decomposed as:

$$\mathbb{E}[(t - f_w(x))^2] = \underbrace{\mathbb{E}[(t - h(x))^2]}_{(i)} + \underbrace{\mathbb{E}[(h(x) - \mathbb{E}[f_w(x)])^2]}_{(ii)} + \underbrace{\mathbb{E}[(\mathbb{E}[f_w(x)] - f_w(x))^2]}_{(iii)}, \quad (1)$$

where $f_w(x)$ is the prediction of our model, parameterized by w , trained on a particular dataset.

We set t to be the true label for the test point, and assume it comes from some underlying ground-truth function h such that

$$t = h(x) + \epsilon,$$

where ϵ is random noise inherent in the system. Assume $\mathbb{E}[\epsilon] = 0$ and $\text{Var}(\epsilon) = \sigma^2$.

(a) From the decomposition above, consider the following symbols:

- t : the true test label
- w : the weights of the fitted model
- x : the test input

Which of these are random variables, and which are not? Briefly explain.

Solution:

- t : Random variable due to the noise ϵ added to the ground-truth function $h(x)$.
- w : Random variable, since the weights are fitted to randomly sampled training data.
- x : Not a random variable (in this context); we are interested in the test error at a fixed x .

- (b) Rewrite equation (1) in terms of σ^2 , b (the **model's bias**), and v (the **model's variance**).

Solution: Term (i) can be rewritten as $\mathbb{E}[\epsilon^2] = \text{Var}(\epsilon) + \mathbb{E}[\epsilon]^2 = \sigma^2 + 0^2 = \sigma^2$.

Term (ii) is the square of the difference between the ground-truth function and the model's expected prediction. Thus, it represents the model's bias squared, b^2 .

Recall that $f_w(x)$ is a random variable because it is dependent on w , which is random. From that perspective, term (iii) is just the definition of variance.

Our final equation is then

$$\mathbb{E}[(t - f_w(x))^2] = \sigma^2 + b^2 + v$$

- (c) Suppose you're tasked with predicting housing prices using a regression model.
- (i) If you move from linear regression to a neural network model with many more parameters, how do you expect the bias and variance to change?
 - (ii) What if you keep linear regression but add a ridge (L_2) regularization term. How does this affect bias and variance?

Solution:

- (i) Moving to a neural network with more parameters:
 - **Bias:** *Decreases* (model is more flexible, can fit more complex patterns)
 - **Variance:** *Increases* (More parameters so model's predictions change a lot depending on the training data)
- (ii) Adding ridge (L_2) regularization to linear regression:
 - **Bias:** *Increases* (model is more restricted, less flexible)
 - **Variance:** *Decreases* (Restriction on change in parameters, so model's predictions become less sensitive to training data variations)

2 Statistical Justification of Logistic Regression

Assume that we have N i.i.d. data points $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, where each y_n is a binary label in $\{0, 1\}$. We model the posterior probability of the labels given the observed features as a Bernoulli distribution, where the probability of a positive sample is given by the sigmoid function, meaning

$$p(Y = y | \mathbf{x}; \mathbf{w}) = p^y(1 - p)^{1-y}, \quad \text{where } p = \sigma(\mathbf{w}^\top \mathbf{x}) \text{ and } \sigma(a) = \frac{1}{1 + e^{-a}}.$$

- (a) Show that for a given data point $\mathbf{x}$, the log ratio of the conditional probabilities, or *log odds*, is linear in $\mathbf{x}$. More specifically, show that

$$\log \frac{p(Y = 1 | \mathbf{x}; \mathbf{w})}{p(Y = 0 | \mathbf{x}; \mathbf{w})} = \mathbf{w}^\top \mathbf{x}.$$

Solution:

$$\begin{aligned} \log \frac{p(Y = 1 | \mathbf{x}; \mathbf{w})}{p(Y = 0 | \mathbf{x}; \mathbf{w})} &= \log \frac{p}{1 - p} \\ &= \log \frac{\frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}}}}{\frac{e^{-\mathbf{w}^\top \mathbf{x}}}{1 + e^{-\mathbf{w}^\top \mathbf{x}}}} \\ &= \log \frac{1}{e^{-\mathbf{w}^\top \mathbf{x}}} \\ &= \mathbf{w}^\top \mathbf{x} \end{aligned}$$

- (b) Using the assumption that the labels are conditionally independent given the features, write the conditional likelihood of the observed labels $\mathbf{y} = (y_1, \dots, y_N)$ given the features $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$. This is the likelihood of our dataset $\mathcal{D}$ given $\mathbf{w}$, $p(\mathcal{D} | \mathbf{w})$.

Solution: Let $p_n = \sigma(\mathbf{w}^\top \mathbf{x}_n)$. Then

$$p(\mathbf{y} | \mathbf{X}; \mathbf{w}) = \prod_{n=1}^N p(Y_n = y_n | \mathbf{x}_n; \mathbf{w}) = \prod_{n=1}^N p_n^{y_n} (1 - p_n)^{1-y_n}.$$

- (c) Take the logarithm of the likelihood from the previous part to obtain the log-likelihood.

Solution: Using the fact that the logarithm of a product is a sum,

$$\log p(\mathbf{y} | \mathbf{X}; \mathbf{w}) = \sum_{n=1}^N [y_n \log p_n + (1 - y_n) \log(1 - p_n)].$$

- (d) Explain why maximum likelihood estimation for this model is equivalent to minimizing the binary cross-entropy loss. Write the resulting optimization problem.

Solution:

Earlier, we found that the likelihood of our data was:

$$\log p(\mathcal{D}|\mathbf{w}) = \log p(\mathbf{y} \mid \mathbf{X}; \mathbf{w}) = \sum_{n=1}^N [y_n \log p_n + (1 - y_n) \log(1 - p_n)].$$

Recall that the binary cross-entropy loss is given by:

$$\ell(\mathbf{w}) = - \sum_{n=1}^N [y_n \log p_n + (1 - y_n) \log(1 - p_n)].$$

Notice that the binary cross-entropy loss is exactly the likelihood of our data, but negated! Maximizing one function is equivalent to minimizing its negation:

$$\arg \max_{\mathbf{w}} f(\mathbf{w}) = \arg \min_{\mathbf{w}} [-f(\mathbf{w})].$$

Conclusion: Thus, maximizing the likelihood under the assumption that labels follow independent Bernoulli distributions is equivalent to minimizing the cross-entropy loss. The cross-entropy loss is often also called the logistic loss or the negative log-likelihood.