Machine Data and Learning

Assignment-2: Bias-Variance Trade-off

Moida Praneeth Jain, 2022101093

Task 1: Gradient Descent

(a)

Gradient descent is an optimization algorithm used to fit a model to its training data.

Assume we have a dataset of the form

$$X = [x_1, x_2, ..., x_n]$$

$$Y = [y_1, y_2, ..., y_n]$$

Where x_i are the independent variables and y_i are the dependent variables

To find the line of best fit (m is the slope and c is the signed y-intercept)

$$y = mx + c$$

We first define a cost function such as Mean-Squared Error (MSE)

$$J = \text{cost} = \text{MSE} = \frac{1}{n}\sum_i (y_i - (mx_i + c))^2$$

We randomly choose initial values for m and c, and iteratively move towards the correct values as follows:

First, find the gradient of the cost function J with respect to m and c

$$\frac{\partial J}{\partial m} = \frac{1}{n}\sum_i (2(y_i - (mx_i + c))(-x_i))$$

$$\frac{\partial J}{\partial m} = -\frac{2}{n}\sum_i x_i(y_i - (mx_i + c))$$

$$\frac{\partial J}{\partial c} = \frac{1}{n}\sum_i (2(y_i - (mx_i + c))(-1))$$

$$\frac{\partial J}{\partial c} = -\frac{2}{n}\sum_i (y_i - (mx_i + c))$$

Let us define L to be the learning rate. It determines the step size in each iteration. A small value would increase accuracy but require more iterations, while a high value would decrease accuracy but require less iterations. Now, we update m and c in the opposite direction of the gradient, with the step size L

$$m = m - L \times \frac{\partial J}{\partial m}$$

$$c = c - L \times \frac{\partial J}{\partial c}$$

We continue this process until our loss function reaches an almost constant value. After we get our final m and c, we have our fitted model.

(b)

For a mutivariable model with q independent and one dependent variable, we have

$$y = (x_1 \;\; x_2 \;\; x_3 \;\; \dots \;\; x_q) \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_q \end{pmatrix} + c$$

$$y = \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q + c$$

The cost function is

$$J = \frac{1}{n} \sum_i \left(y_i - \left(\sum_{j=1}^{q} (\beta_i x_i) + c \right) \right)^2$$

The coefficients (β) can be found simply by extending the single variable case.

$$\beta_i = \beta_i - L \frac{\partial J}{\partial \beta_i}$$

$$c = c - L \frac{\partial J}{\partial c}$$

We iteratively update all β_i and c until the cost function converges, and then we get our fitted model.

Task 2: Numerical on Bias and Variance

Given

$$x = [-2, -1, 0, 1, 2, 3]$$

$$y = f(x) = [5, 0, 1, 4, 11, 22]$$

$$f_1(x) = 2x^2 + 3x + 1$$

$$f_2(x) = x^2 + 3x$$

$$f_3(x) = 2x^2 + 2x + 1$$

We need to find the bias squared, variance and MSE

First, we apply the model the input values

$$\widehat{f_1}(x) = [3, 0, 1, 6, 15, 28]$$

$$\widehat{f_2}(x) = [-2, -2, 0, 4, 10, 18]$$

$$\widehat{f_3}(x) = [5, 1, 1, 5, 13, 25]$$

To get bias

$$\text{Bias} = E_i\left[\hat{f}_i(x)\right] - f(x)$$

$$\text{Bias} = \frac{1}{3}\sum_i \hat{f}_i(x) - f(x)$$

$$\text{Bias} = \frac{1}{3}[6, -1, 2, 15, 38, 71] - [5, 0, 1, 4, 11, 22]$$

$$\text{Bias} = [-3, -0.33, -0.33, 1, 1.67, 1.67]$$

$$\text{Bias}^2 = [9, 0.11, 0.11, 1, 2.77, 2.77]$$

To get variance

$$\text{Variance} = E_i\left[\left(\hat{f}_i(x) - E_i\left[\hat{f}_i(x)\right]\right)^2\right]$$

$$\text{Variance} = E_i\left[\left(\hat{f}_i(x) - [2, -0.3, 0.67, 5, 12.67, 23.67]\right)^2\right]$$

$$\begin{gathered}\text{Variance} = \frac{1}{3}\Big(([3, 0, 1, 6, 15, 28] - [2, -0.3, 0.67, 5, 12.67, 23.67])^2 + \\ ([-2, -2, 0, 4, 10, 18] - [2, -0.3, 0.67, 5, 12.67, 23.67])^2 + \\ ([9, 0.11, 0.11, 1, 2.77, 2.77] - [2, -0.3, 0.67, 5, 12.67, 23.67])^2\Big)\end{gathered}$$

$$\text{Variance} = [8.67, 1.56, 0.22, 0.67, 4.22, 17.56]$$

To get MSE

$$\text{MSE} = E_i\left[\left(f(x) - \hat{f}_i(x)\right)^2\right]$$

$$\begin{gathered}\text{MSE} = \frac{1}{3}\Big(([5, 0, 1, 4, 11, 22] - [3, 0, 1, 6, 15, 28])^2 + \\ ([5, 0, 1, 4, 11, 22] - [-2, -2, 0, 4, 10, 18])^2 + \\ ([5, 0, 1, 4, 11, 22] - [5, 1, 1, 5, 13, 25])^2\Big)\end{gathered}$$

$$\text{MSE} = [17.67, 1.67, 0.33, 1.67, 7, 20.33]$$

Note that

$$\text{Bias}^2 + \text{Variance} = [9, 0.11, 0.11, 1, 2.77, 2.77] + [8.67, 1.56, 0.22, 0.67, 4.22, 17.56]$$

$$\text{Bias}^2 + \text{Variance} = [17.67, 1.67, 0.33, 1.67, 7, 20.33]$$

$$\text{Bias}^2 + \text{Variance} = \text{MSE}$$

Thus, the formula has been verified.

Task 3: Calculating Bias and Variance

Degree	Bias	Bias Square	Variance	MSE
1	0.2365	1.0075	0.0389	1.0464
2	0.2292	0.9486	0.0493	0.9979
3	−0.0122	0.0143	0.0834	0.0978
4	0.0008	0.0132	0.13	0.1432
5	0.0063	0.0122	0.149	0.1611
6	0.0061	0.0127	0.1691	0.1818
7	0.0099	0.0131	0.2163	0.2294
8	0.025	0.0279	0.2239	0.2519
9	0.0562	0.1019	0.2136	0.3155
10	0.0853	0.2653	0.2009	0.4661

As the degree of the polynomial increases, the complexity of the model increases. Since models with a large number of parameters tend to have higher variance, we would expect the variance to monotonically increase with the degree of the polynomial. This can be emperically verified by observing the trend of variance increasing with degree in the above table.

For smaller degrees, the data is underfit, thus the bias square is high but the variance is low. As complexity increases, the bias square is expected to monotonically decrease with the degree of the polynomial. This is mostly true for the data tabulated above.

Task 4: Calculating Irreducible Error

Degree	Irreducible Error
1	0
2	0
3	0
4	0
5	0
6	0

7	0
8	0
9	0

The irreducible error

$$\sigma^2 = E_i\left[\left(y - \hat{f}_i(x)\right)^2\right] - (\mathrm{Bias}^2 + \mathrm{Variance})$$

is a property of the data itself, and is independent of the model chosen. This error can not be reduced by creating better models. This error may not always be 0, but for this dataset, its value is 0, and thus remains constant throughout.