CS7.301: Assignment 2

Bias-Variance Tradeoff

Himanshu Singh (2023121013)

February 9, 2024

1 Gradient Descent

Gradient descent is an iterative algorithm used in optimization, specifically to minimize the so called "cost function". In the case of linear regression, we typically seek to minimize the mean square error, given by the following function:

$$\begin{aligned} L &= \frac{1}{n}\sum_{i=1}^{n}(\hat{f}(x_i) - y_i)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x} - y_i)^2 \end{aligned}$$

We assume the input vector is a $d+1$ dimensional vector, starting with a constant 1, to accomodate for the bias term w_0. We start with an initial estimate for the optimization variables w_i and update them according to the following equation:

$$w_i = w_i - \alpha\frac{\partial L}{\partial w_i}$$

Here, α is called the step size, and the partial derivatives give us the descent direction. This can equivalently be written in vector notation, i.e. in terms of the gradient vector.

$$\mathbf{w} = \mathbf{w} - \alpha\nabla_w L$$

We apply this iteratively until a stopping condition is satisfied, say we have achieved the minima, i.e. $\nabla_w L = 0$, or more practically $\|\nabla_w L\| \leq \epsilon$, for some $\epsilon > 0$.

The analytical solution for linear regression, obtained by solving $\nabla_w L = 0$, is given by:

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Here, $\mathbf{X}$ and $\mathbf{y}$ represent the vector of (the possibly multivariate) training input and output points respectively.

2 Numerical on Bias and Variance

x	y	$f_1(x)$	$f_2(x)$	$f_3(x)$	$\mathbf{E}_i[f_i(x)]$	$\mathbf{E}_i[f_i(x)] - y$	$[\mathbf{E}_i[f_i(x)] - y]^2$
-2	5	3	-2	5	2	-3	9
-1	0	0	-2	1	-0.33	-0.33	0.11
0	1	1	0	1	0.66	-0.33	0.11
1	4	6	4	5	5	1	1
2	11	15	10	13	12.66	1.66	2.77
3	22	28	18	25	23.66	1.66	2.77
							2.63

Table 1: Calculation of Bias^2

x	$(f_1(x) - \mathbf{E}_i[f_i(x)])^2$	$(f_2(x) - \mathbf{E}_i[f_i(x)])^2$	$(f_3(x) - \mathbf{E}_i[f_i(x)])^2$	$\mathbf{E}_i[(f_i(x) - \mathbf{E}_i[f_i(x)])^2]$
-2	1	16	9	8.66
-1	0.11	2.77	1.77	1.55
0	0.11	0.44	0.11	0.22
1	1	1	0	0.66
2	5.44	7.11	0.11	4.22
3	18.77	32.11	1.77	17.55
				5.48

Table 2: Calculation of Variance

x	$(y - f_1(x))^2$	$(y - f_2(x))^2$	$(y - f_3(x))^2$	$\mathbf{E}_i[(y - f_i(x))^2]$
-2	4	49	0	17.66
-1	0	4	1	1.66
0	0	1	0	0.33
1	4	0	1	1.66
2	16	1	4	7
3	36	16	9	20.33
				8.11

Table 3: Calculation of MSE

See Table 1, Table 2 and Table 3. Thus, we conclude that MSE = Bias^2 + Variance, for the given data.

3 Calculating Bias and Variance

We can observe that bias decreases rapidly, as we bring the degree of polynomial down to 3. This is also accompanied with a decrease in MSE (see Table 5). This points to a strong possibility that the test data can most accurately be modeled using a polynomial of degree 3.

Variance, as one would expect, increases with increase in model complexity. This results in a regular increase in MSE (see Table 5), after degree = 3, when the sharp decrease in Bias^2 had stagnated. Notably, bias increases rapidly, after degree = 8, possibly as the model tries to overfit on the noise.

Degree of Polynomial	Average Bias	Average Bias^2	Average Variance
1	-0.23648566043247068	0.9936640602861002	0.07953765467730094
2	-0.23065115241058906	0.9481028389983889	0.10697227621961831
3	0.022023412996105995	0.017391283982556432	0.11912286770019158
4	-0.008031184511968848	0.027668586044199573	0.19250257249294855
5	-0.01009935641279951	0.028914548854186456	0.20951057588851082
6	-0.005727057309104389	0.02906706046367728	0.25042806449044935
7	-0.009345566968466335	0.031021600796763794	0.30112437488602656
8	-0.02610912867775783	0.04824938814996045	0.24500478677746512
9	-0.057305823747988094	0.12234505440502912	0.24395941553756498
10	-0.08794196460108164	0.2908053168874439	0.2539041076577705

Table 4: Average Bias and Variance of models, versus their Degree of Polynomial

4 Calculating Irreducible Error

Irreducible error is associated with the inherent randomness/noise/variability in the data that cannot be explained by the features or predictors in the model. The minor fluctuations in the irreducible error, tabulated here, with change in model, are indeed a manifestation of this randomness.

Degree of Polynomial	Average MSE	Average Irreducible Error
1	1.0732017149634019	7.355227538141662e-16
2	1.0550751152180062	-9.575673587391975e-16
3	0.13651415168274794	-6.938893903907228e-17
4	0.22017115853714825	1.1102230246251565e-16
5	0.23842512474269725	-2.7755575615628914e-17
6	0.2794951249541265	-1.6653345369377348e-16
7	0.33214597568279014	-2.220446049250313e-16
8	0.29325417492742545	-1.1102230246251565e-16
9	0.36630446994259414	2.7755575615628914e-17
10	0.5447094245452146	2.7755575615628914e-16

Table 5: MSE and Irreducible Error of models, versus their Degree of Polynomial

5 Plotting Bias^2 - Variance Graph

Similar observations from section 3 follow. The underlying test data seems to be more aligned with polynomial of degree 3. The high bias for degree < 3 indicate underfitting of data. On the other hand, the high variance and still increasing bias, for degree > 8 indicate overfitting, with the model not generalizing on the larger patterns, and trying to fit on the noise present in the training data. The reasonably good bias and variance for degree $\in [3, 8]$ suggest that models with higher degree polynomials are choosing to simulate the effect of a cubic function, to a great extent, rather than exploring drastically different combinations.

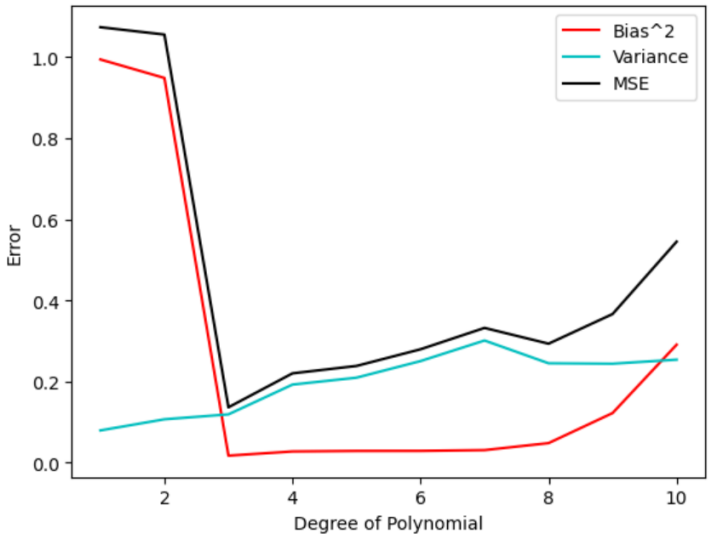

Figure 1: Plot variation of Bias^2, Variance and MSE, versus Degree of Polynomial