Indian Institute of Technology, Kharagpur

PGDBA Programme

Department of Computer Science Engineering

Fundamentals of Algorithm Design and Machine Learning

Sushant Shekhar: 24BM6JP54

Instructor: Dr. Aritra Hazra

(1st April 2025, 11:00AM to 11:55AM) Scribe

Approximation VS Generalization

Let Hypothesis set **H** = {h_1,h_2,h_3,…….., h_n}

Our learning algorithm takes a hypothesis from **H** so that it can approximate properly the unknown target function **f.**

The **VC analysis** showed that the choice of **H** needs to strike a balance between approximating for the training data and generalizing on new data. The ideal **H** is a singleton hypothesis set containing only the target function. Since we do not know the target function, we resort to a larger model hoping that it will contain a good hypothesis, and hoping that the data will pin down that hypothesis. When you select your hypothesis set, you should balance these two conflicting goals; to have some hypothesis in H that can approximate f, and to enable the data to zoom in on the right hypothesis.

The VC generalization bound is one way to look at this trade-off.

Case 1

If H is too simple, we may fail to approximate f well and end up with a large in-sample error term.

Case 2

If H is too complex, we may fail to generalize well because of the large model complexity term.

There is another way to look at the approximation-generalization trade-off which we will present in this section. It is particularly suited for squared error measures, rather than the binary error used in the VC analysis. The new way provides a different angle; instead of bounding E_{out} by E_{in} plus a penalty term Ω, we will decompose E_{out} into two different error terms.

Bias VS Variance

The bias-variance decomposition of out of sample error is based on squared error measures. The out-of-sample error is

$$E_D[E_{out}(g^D)] = Ex\,[(g^D(x) - f(x))^2] \qquad D = \text{dataset}$$

The hypothesis g(x) is obtained by the help of training dataset D for which $E_{in}(g)$ was minimum.

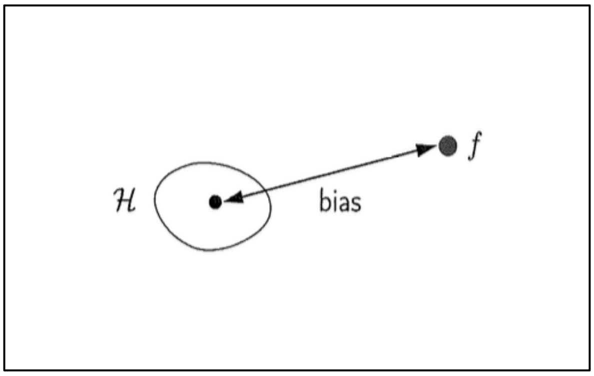
$$\begin{aligned}
\mathbb{E}_D[\mathbb{E}_{\text{out}}(g_D)] &= \mathbb{E}_D\left[\mathbb{E}_x\left((g_D(x) - f(x))^2\right)\right] \\
&= \mathbb{E}_x\left[\mathbb{E}_D\left((g_D(x) - f(x))^2\right)\right] \\
&= \mathbb{E}_x\left[\mathbb{E}_D\left((g_D(x) - \bar{g}(x) + \bar{g}(x) - f(x))^2\right)\right] \\
&= \mathbb{E}_x\left[\mathbb{E}_D\left((g_D(x) - \bar{g}(x))^2 + 2(g_D(x) - \bar{g}(x))(\bar{g}(x) - f(x)) + (\bar{g}(x) - f(x))^2\right)\right] \\
&= \mathbb{E}_x\left[\mathbb{E}_D\left((g_D(x) - \bar{g}(x))^2\right) + 2(\bar{g}(x) - f(x))\mathbb{E}_D(g_D(x) - \bar{g}(x)) + (\bar{g}(x) - f(x))^2\right] \\
&= \mathbb{E}_x\left[\mathbb{E}_D\left((g_D(x) - \bar{g}(x))^2\right) + 0 + (\bar{g}(x) - f(x))^2\right] \\
&= \mathbb{E}_x\left[\text{Variance}(x) + \text{Bias}(x)\right]
\end{aligned}$$

where:

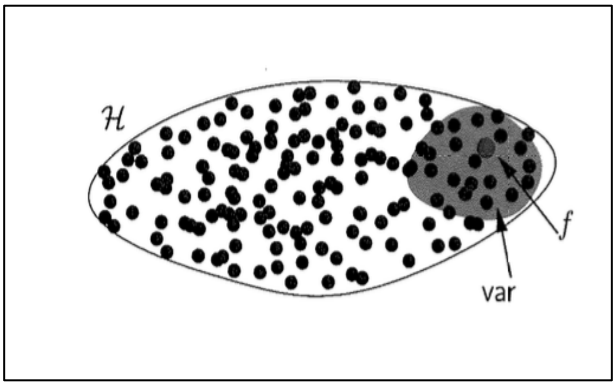
Variance = $E_D[(g^D(x) - \bar{g}(x))^2]$ and Bias(x) = $(\bar{g}(x) - f(x))^2$

The expected value of the out-of-sample error depends on both the variance and the bias of the learning algorithm. If we choose a hypothesis set that generalizes well and is flexible enough to closely approximate the true function, the bias tends to decrease, as the learned hypothesis can better capture the underlying pattern. However, increasing the flexibility of the hypothesis set can also lead to higher variance, since the learned function may vary significantly with different training datasets. This trade-off between reducing bias and increasing variance is known as the *bias-variance trade-off*.

To illustrate let us consider two extreme cases: a very small model (with one hypothesis) and a very large one with all hypothesis.



Very small model. Since there is only one hypothesis, both the average function $\bar{g}$ and the final hypothesis g^D will be the same, for any data set. Thus, variance= 0. The bias will depend solely on how well this single hypothesis approximates the target f, and unless we are extremely lucky, we expect a large bias.



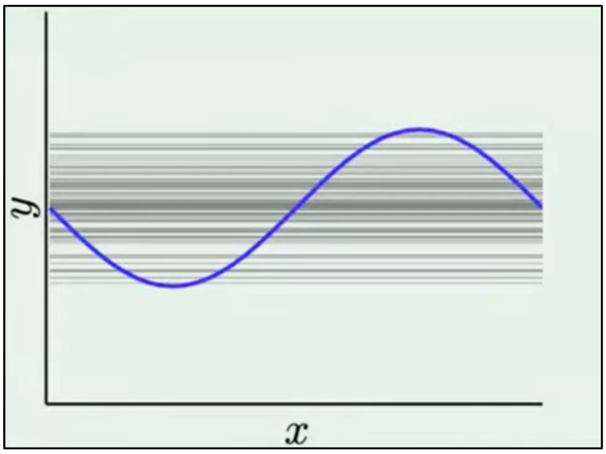
Very large model. The target function is in H. Different data sets will lead to different hypotheses that agree with f on the data set, and are spread around f in the shaded region. Thus, bias ≈ 0 because $\bar{g}$ is likely to be close to f. The var is large (heuristically represented by the size of the shaded region in the figure).

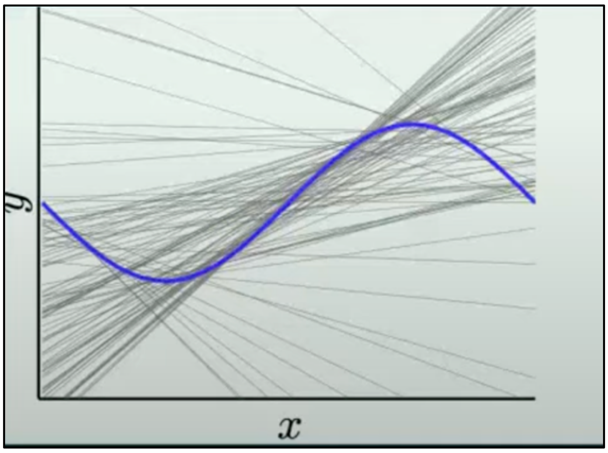
EXAMPLE : Consider the target function f(x)=sin(πx) and a dataset of size N=2 .We sample x uniformly in [−1,1] and fit the data using one of the following two models:

$$\mathbf{H_0}\text{: } h(x) = b$$

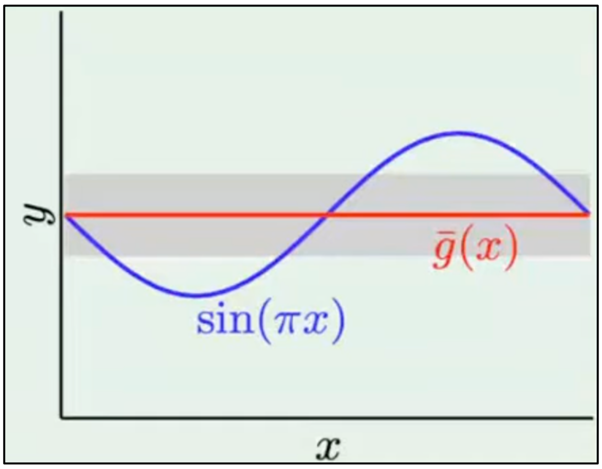
$$\mathbf{H_1}\text{: } h(x) = ax + b$$

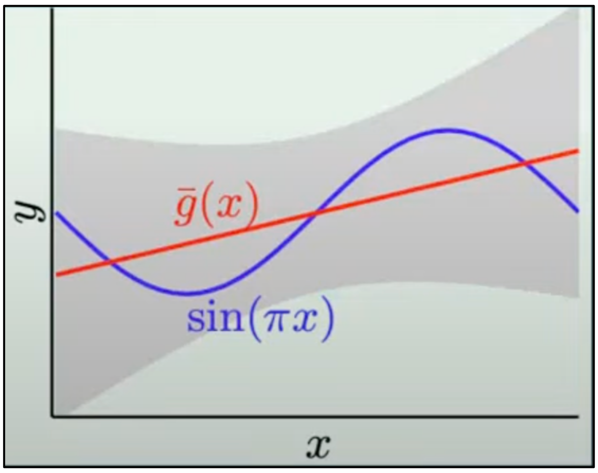
⇒ For **H₀** we choose the constant hypothesis that best fits the data (the horizontal line at the midpoint, b = $(y1 + y2)/2$) For **H₁** , we choose the line that passes through the two data points (x1,y1) and (x2, y2) Repeating this process with many data sets, we can estimate the bias and the variance. The figures which follow show the resulting fits on the same (random) data sets for both models.





With **H₁** the learned hypothesis is wilder and where is extensively depending on the data set the bias variance analysis is summarised in the next figure





For $\mathbf{H_0}$ Bias = 0.5, Variance = 0.25

For $\mathbf{H_1}$, Bias = 0.21, Variance = 1.69

For $\mathbf{H_1}$ the average hypothesis $\bar{g}$ (red line) is a reasonable fit fairly small bias of 0.21. However the large variability lead to **high variance** of **1.69** resulting in a large **expected out of sample error of 1.90**. With the simpler model $\mathbf{H_0}$, the fits are less volatile and we have significantly **lower variance of 0.25**, as indicated by the shaded region . However the average fit is now the zero function resulting in a **higher bias of 0.50**. The **total out of sample error** has a much smaller expected value of **0.75**. The simpler model wins by significantly decreasing the variance at the expense of a smaller increase in bias.

NOTE :

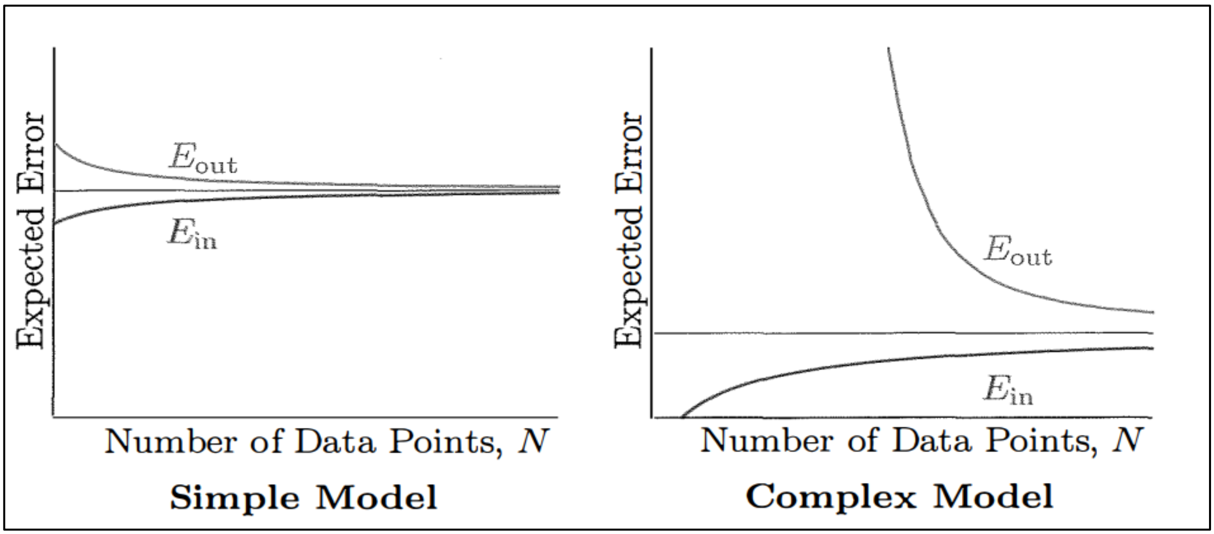
1. It is not only about how best your hypothesis is performing in train data, as same hypothesis has to perform well in out of sample also that means $\mathbf{E_{out}}$ should also be minimum
2. If the chosen hypothesis is very good but further if we can't choose best hypothesis among them then variance will become high which will ultimately increase $\mathbf{E_{out}}$.

The Learning Curve

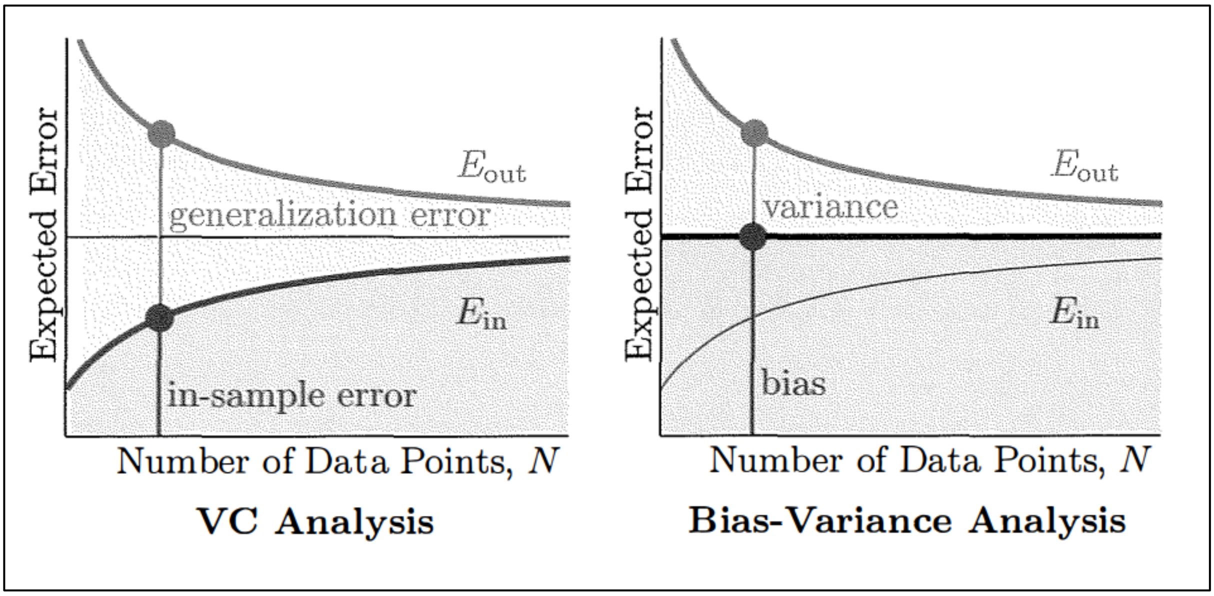
The learning curves summarize the behaviour of the in-sample and out-of-sample errors as we vary the size of the training set.

After learning with a particular data set D of size N, the final hypothesis $\mathbf{g^{(D)}}$ has in-sample error $\mathbf{E_{in}}$ ($\mathbf{g^{(D)}}$) and out-of-sample error $\mathbf{E_{out}}$ ($\mathbf{g^{(D)}}$) both of which depend on D. As we saw in the bias-variance analysis, the expectation with respect to all data sets of size

N gives the expected errors: $\mathbf{E_D[E_{in}(g^{(D)})]}$ and $\mathbf{E_D[E_{out}(g^{(D)})]}$. These expected errors are functions of N, and are called the learning curves of the model.



In the VC analysis, $\mathbf{E_{out}}$ was expressed as the sum of $\mathbf{E_{in}}$ and a generalization error that was bounded by $\mathbf{\Omega}$, the penalty for model complexity. In the bias-variance analysis, $\mathbf{E_{out}}$ was expressed as the sum of a bias and a variance. The following learning curves illustrate these two approaches side by side.



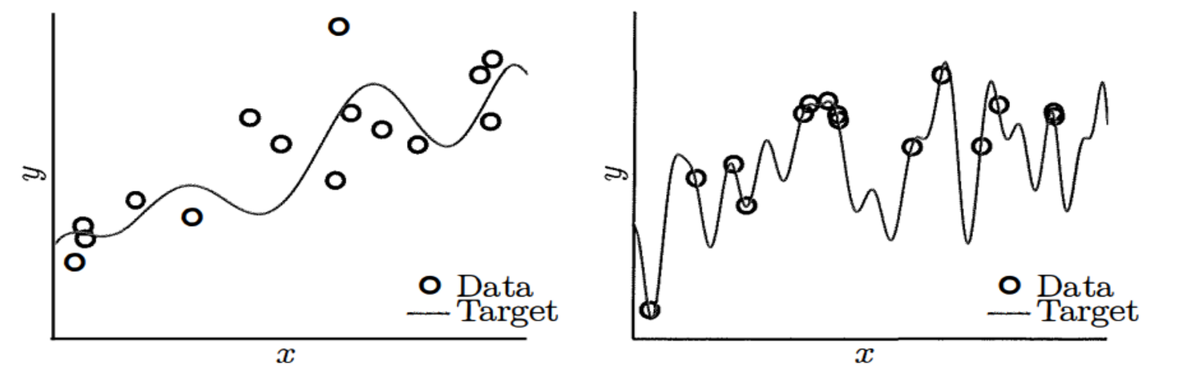
From the above VC Analysis we can infer that $\mathbf{E_{out} \leq E_{in} + \Omega}$ and by looking at the bias variance analysis we can infer that $\mathbf{E_{out} \leq}$ **Variance + Bias.**

Overfitting

Overfitting literally means "Fitting the data more than is warranted." The main case of overfitting is when you pick the hypothesis with lower $\mathbf{E_{in}}$, and it results in higher $\mathbf{E_{out}}$.

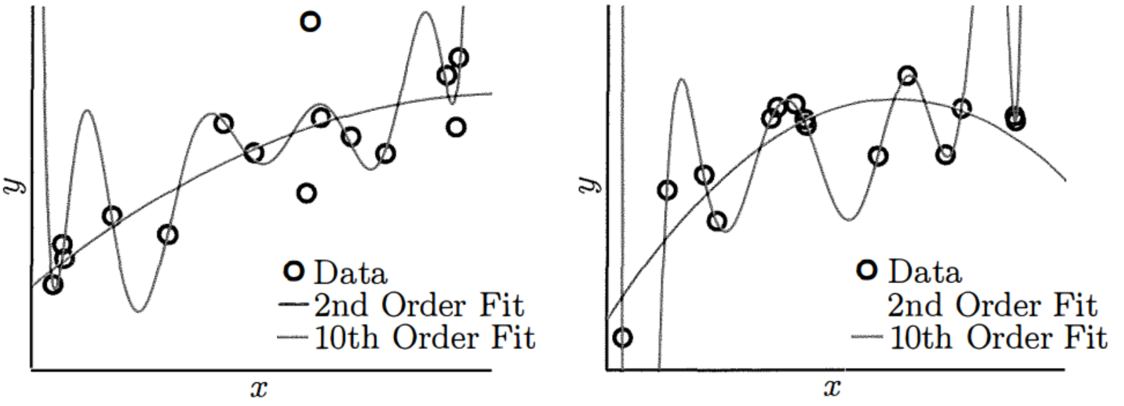
This means that $\mathbf{E_{in}}$ alone is no longer a good guide for learning.

Let's dig deeper to gain a better understanding of when overfitting occurs. We will illustrate the main concepts using data in one-dimension and polynomial regression.



(a) 10th order target function (b) 50th order target function

In both problems, the target function is a polynomial and the data set **D** contains 15 data points. In (a), the target function is a 10th order polynomial and the sampled data are noisy (the data do not lie on the target function curve). In (b) , the target function is a 50th order polynomial and the data are noiseless.



(a) Noisy low order target (b) Noiseless high order target

The best 2nd and 10th order fits are shown in above figure, and the in-sample and out-of-sample errors are given in the following table.

10th order noisy target			50th order noiseless target		
	2nd Order	10th Order		2nd Order	10th Order
E_{in}	0.50	0.034	**E_{in}**	0.029	10^{-5}
E_{out}	0.127	9.00	**E_{out}**	0.120	7680

What the learning algorithm sees is the data, not the target function. In both cases, the 10th order polynomial heavily overfits the data, and results in a final hypothesis which does not resemble the target function. The 2nd order fits do not capture the full nature of the target function either, but they do at least capture its general trend, resulting in significantly lower out-of-sample error. The 10th order fits have lower in-sample error and higher out-of- sample error, so this is indeed a case of overfitting that results in bad generalization.

The figure(a) has the **stochastic noise** and the 10th order polynomial is trying to fit the data that is why the **E_{out}** is very high . But can we say that figure (b) has no noise ?

Actually the 10th Order polynomial is incapable to fit the 50th Order polynomial as a result noise is present and this noise is known as **Deterministic Noise**.

Expected out-of-**sample error** :

$$\text{Assume: } y = f(x) + \varepsilon(x), \quad \mathbb{E}[\varepsilon(x)] = 0, \quad \mathrm{Var}[\varepsilon(x)] = \sigma^2$$

$$\mathbb{E}_{D,\varepsilon}[(g^{(D)}(x) - y)^2] = \mathbb{E}_{D,\varepsilon}[(g^{(D)}(x) - f(x) - \varepsilon(x))^2]$$

$$= \mathbb{E}_{D,\varepsilon}[(g^{(D)}(x) - \bar{g}(x) + \bar{g}(x) - f(x) - \varepsilon(x))^2]$$

$$= \mathbb{E}_{D,\varepsilon}[(g^{(D)}(x) - \bar{g}(x))^2 + (\bar{g}(x) - f(x))^2 + \varepsilon(x)^2]$$

$$= \underbrace{\mathbb{E}_D[(g^{(D)}(x) - \bar{g}(x))^2]}_{\text{Variance}} + \underbrace{(\bar{g}(x) - f(x))^2}_{\text{Bias}} + \underbrace{\mathbb{E}[\varepsilon(x)^2]}_{\text{Stochastic Noise}}$$

$$\Rightarrow \mathbb{E}_{x,D,\varepsilon}[(g^{(D)}(x) - y)^2] = \mathbb{E}_x[\text{Variance}(x)] + \mathbb{E}_x[\text{Bias}(x)] + \text{Stochastic Noise}$$