Machine Learning CS60050
Instructor: Dr. Aritra Hazra
Department of Computer Science and Engineering
Indian Institute of Technology,Kharagpur

Scribed by: Prasanta Kr Sen (20RJ92R05)

Notes for the class on - 25^{th} February 2021
Part -I

1 Recap

From the brief overview of the previous class, our main target to make E_{out} to be zero. In that case, we did it two ways, make our in-sample error zero by our learning algorithms, and second, we want to track how the in sample behaviour is tracking the out of sample. With these two, we get a generalised sample behaviour as well.

$$E_{out} \approx 0 \Rightarrow (E_{in} \approx 0) + (E_{in} \approx E_{out})$$

$$Prob[|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 4 * m_H(2N) * e^{-1/8\epsilon^2 N}$$

Now E_{in} and E_{out} difference is more than ϵ that means E_{in} is not correctly tracking E_{out} and we make abound with respect to the growth function, where our growth function is nothing but

$$m_H(N) = O(N_{vc}^d)$$

$$m_H(N) \leq \sum_{i=0}^{d_{vc}} \binom{N}{i} \; as \; d_{vc} = k - 1$$

with break point k and we compute vc dimention such as a way.

In an example of d dimension perception we have seen and prove that.

$$d_{vc} = d + 1$$

Now we deduce the generalization bound in this way that we considered to be δ where

$$\delta = 4 * m_H(2N) * e^{-1/8\epsilon^2 N}$$

that mean the bad thing should be bounded by δ. In the other word, we could see with probability / confidence greater than 1 - δ and we can find the error bound less than equals to δ. It is call PAC (Probably approximately correct learning).

$$with \; probability \; / \; confidence \geq 1 - \delta$$

$$we\ find, E_{out} - E_{in} \leq \epsilon$$

So, Generalization bound are written as -

$$\epsilon = \sqrt{\frac{8}{N} ln(\frac{4m_H(2N)}{\delta})}$$

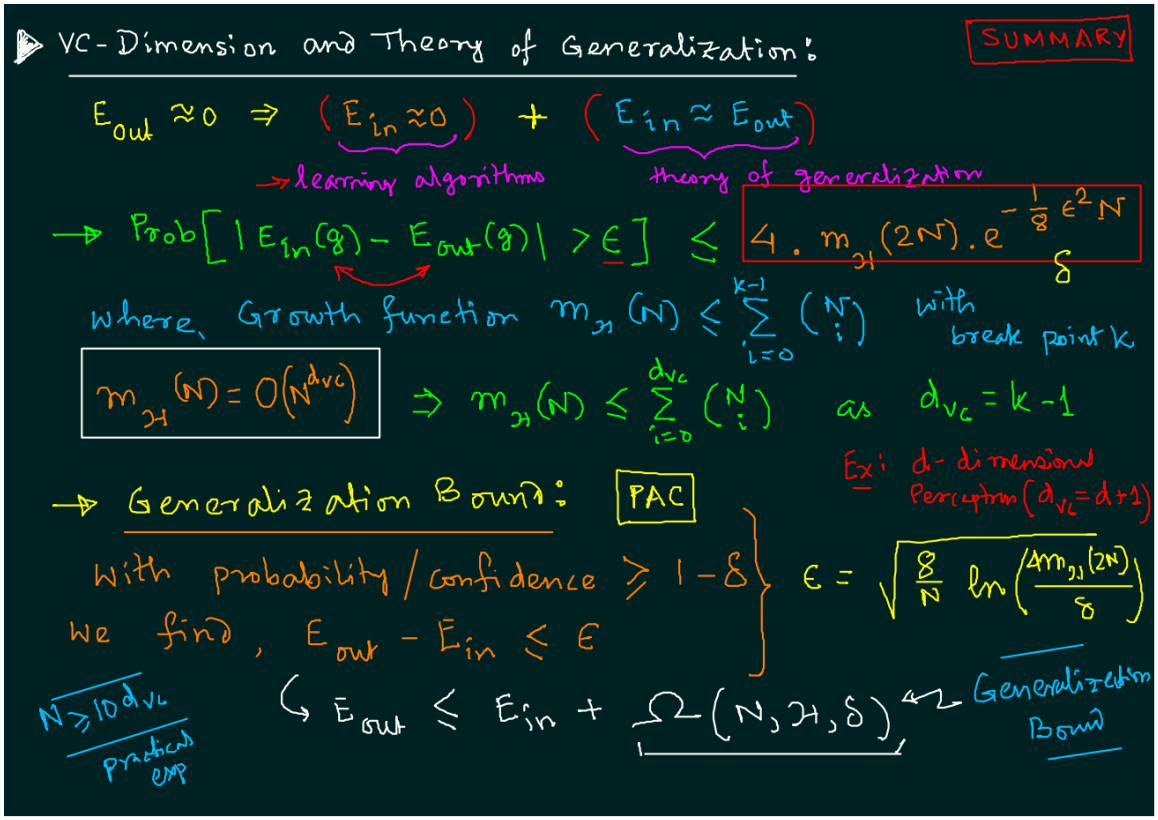
$$E_{out} \leq E_{in} + \Omega(N, H, \delta)$$

Here $\Omega(N, H, \delta)$ is the maximum error that we learn and after our learning algorithm and minimize, the E_{in} is bound and track E_{out} within this range.

Also we know that from the practical experience

$$N \geq 10 d_{vc}$$

where d_{vc} *is* $d-$dimension perception



2 Approximation vs Generalization:

We have to try to learn a function f (unknown)

$$f : X \to Y$$

. and we are only learning it with

$$< X_1, Y_1 > .... < X_N, Y_N >$$

. We don't have access function f. However, we have only access to the training example

$$< X_1, Y_1 > .... < X_N, Y_N >$$

Now we have some Hypothesis Set

$$H = \{h_1, h_2 .....\}$$

Then the learning algorithm

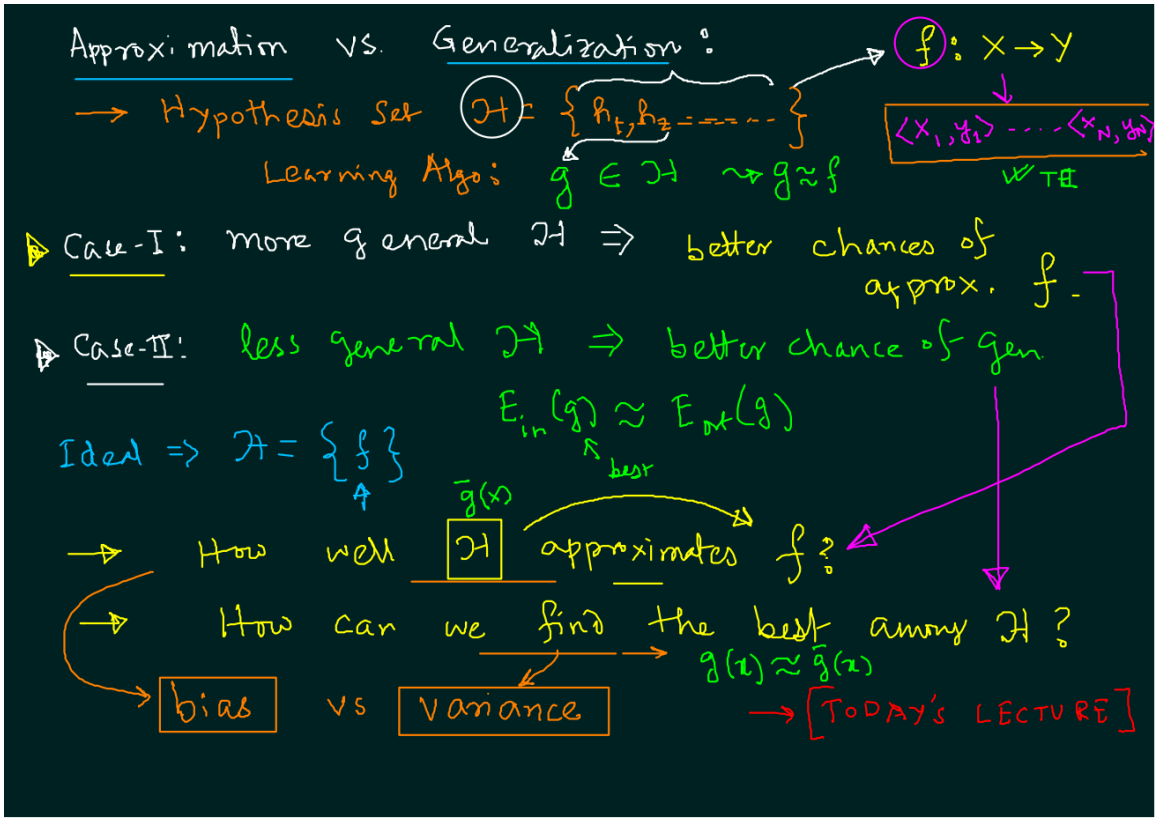
$$g \in H$$

and the theory of generalization says

$$g \approx f$$

There are two aspects: how good your hypothesis set is, and the second aspect is that good of our hypothesis set is the close we get this function "f". Now the consideration is
Case-I: more general H $\Rightarrow$ better changes of approx function f
Case-II: less general H $\Rightarrow$ better changes of generalizing it that means $E_{in}(g) \approx E_{out}(g)$ for best g.

So, the ideal is that H = $\{f\}$



So, two concepts are marge when we try approximation vs generalization. and this leads to an interesting point that raises two questions -
1) Hence How well H approximate f?
2) How can we find the best among H?
The first question comes from Case-I and the second question comes from Case-II. The first question comes from bias, and the second called variance. It is a trade-off between bias vs variance to determine how good our learning can be generalized.

3 Bias vs Variance

$$E_{out}(g^d) = E_x[g^d(x) - f(x)]^2$$
$$d' \to g'$$

It means a point x how far it is from f(x) where training data-set d. So, we give the training example. Our learning algorithm answers hypothesis g, and if we find another data-set d', we could have a different hypothesis as the answer. That is the d data-set we converge into g. and E_x is the expectation over all the point in the out of sample space.

We have nullified the impact of g because it has an expected value concerning all the data-set we had here. So the equation modified as

$$E_d[E_{out}(g^d)] = E_d[E_x[g^d(x) - f(x)]^2]$$

From the equation, we see the inside left-hand term is squared term, so it is positive. For that, we can turn the equation in a reversed way.

$$E_d[E_{out}(g^d)] = E_x[E_d[g^d(x) - f(x)]^2]$$

Now to evaluate the left hand inside aspect, we define the avg hypothesis.

$$H = \{h_1, h_2 \ldots\ldots h_M\}$$

We want to make sure that in this hypothesis set where is the mean hypothesis resides, which means that the avg hypothesis would be like

$$\bar{g}(x) = E_d\left[g^d(x)\right]$$

Let us imagine we have a distribution d_1, d_2 ..... d_k where $d_1 - d_k$ is the k discrete distribution. so the equation is

$$\bar{g}(x) \approx \frac{1}{k}\sum_{k=1}^{K} g^{d_k}(x)$$

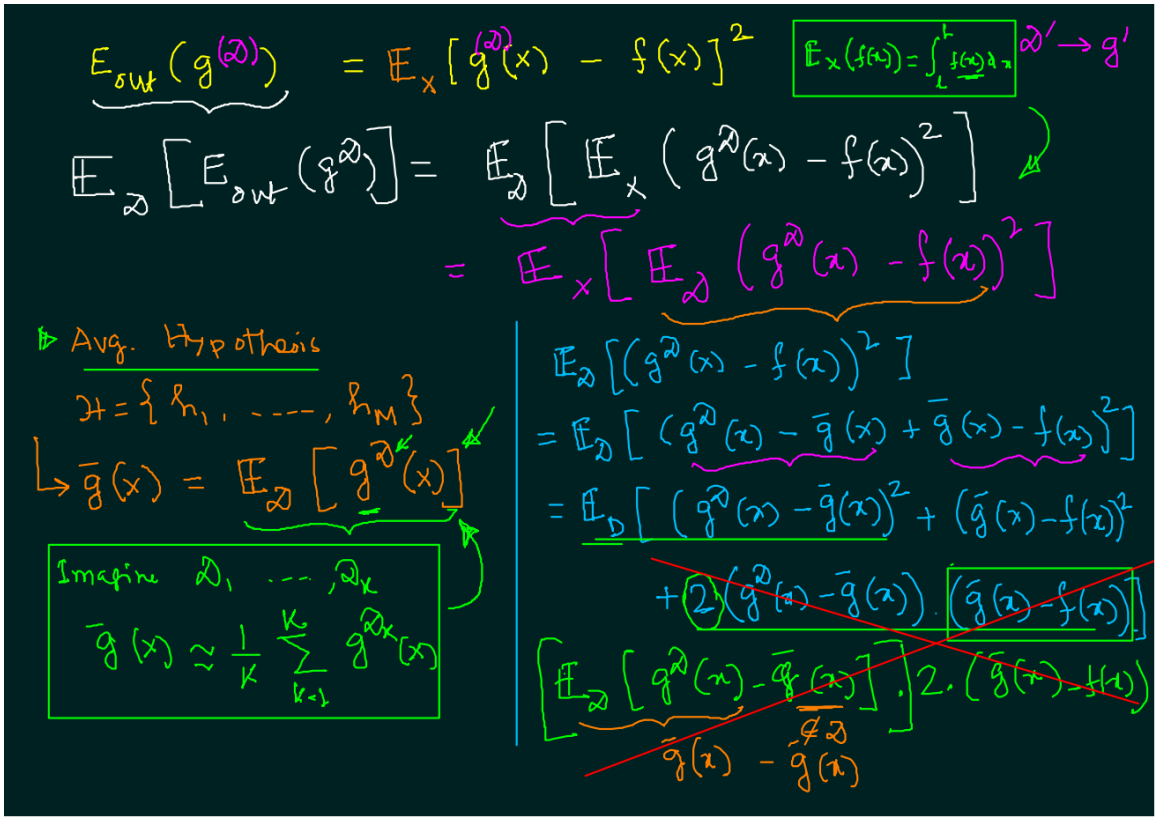
This is called Avg Hypothesis.

Now we extracting the inside part and rewrite it

$$E_d[(g^d(x) - f(x))^2]$$
$$= E_d[(g^d(x) - \bar{g}(x) + \bar{g}(x) - f(x))^2]$$
$$= E_d[(g^d(x) - \bar{g}(x))^2 + (\bar{g}(x) - f(x))^2 + 2 * (g^d(x) - \bar{g}(x)) * (\bar{g}(x) - f(x))]$$

So, after simplifying this, we get

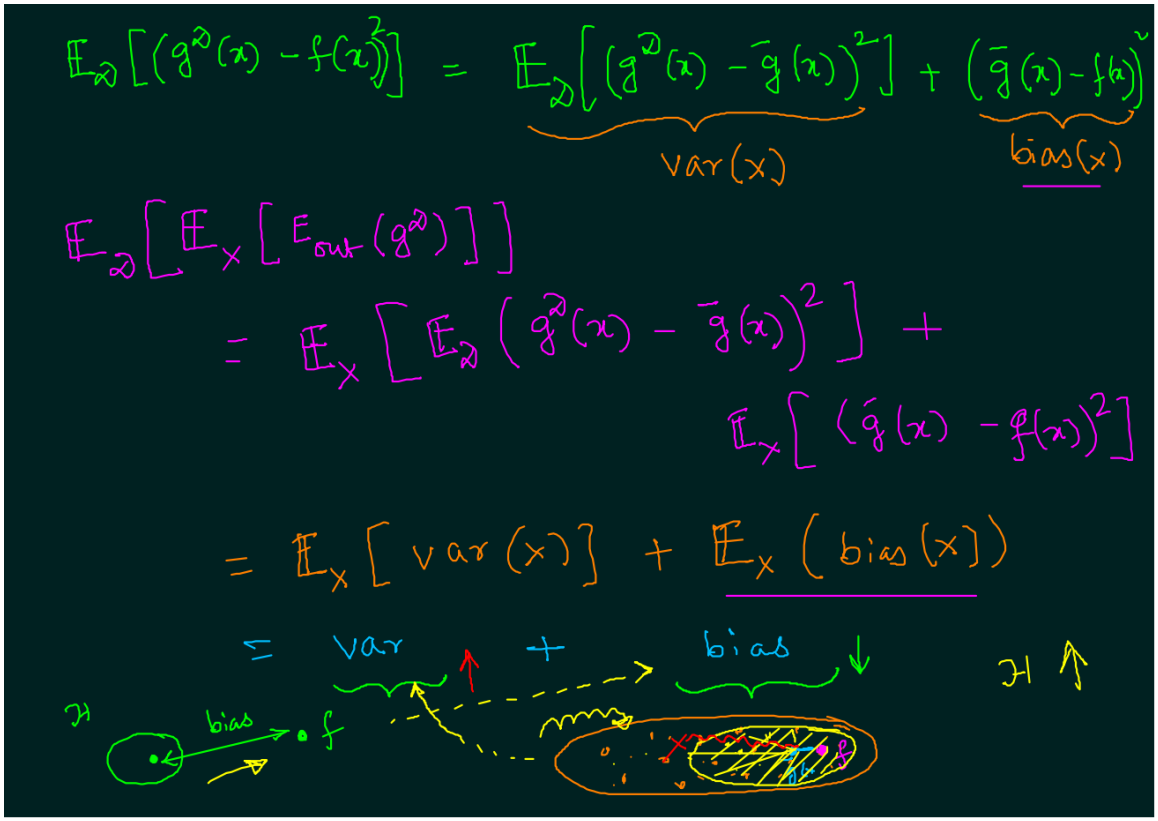
$$E_d[(g^d(x) - f(x))^2] = E_d[(g^d(x) - \bar{g}(x))^2] + (\bar{g}(x) - f(x))^2$$



here first part is the Var(x) and the second part is the Bias(x)

$$E_d[E_x[E_{out}(g^d)]] = E_x[E_d[(g^d(x) - \bar{g}(x))^2]] + E_x[(\bar{g}(x) - f(x))^2]$$
$$= E_x[var(x)] + E_x[bias(x)]$$
$$= var + bias$$

So, we can see the expected error value out of the sample will depend not only on the bias but also on the variance. So, the algorithm's ethics concerning the hypothesis set to increase where your bias is reduced and the variance will be high. It is called bias vs variance trade-off.



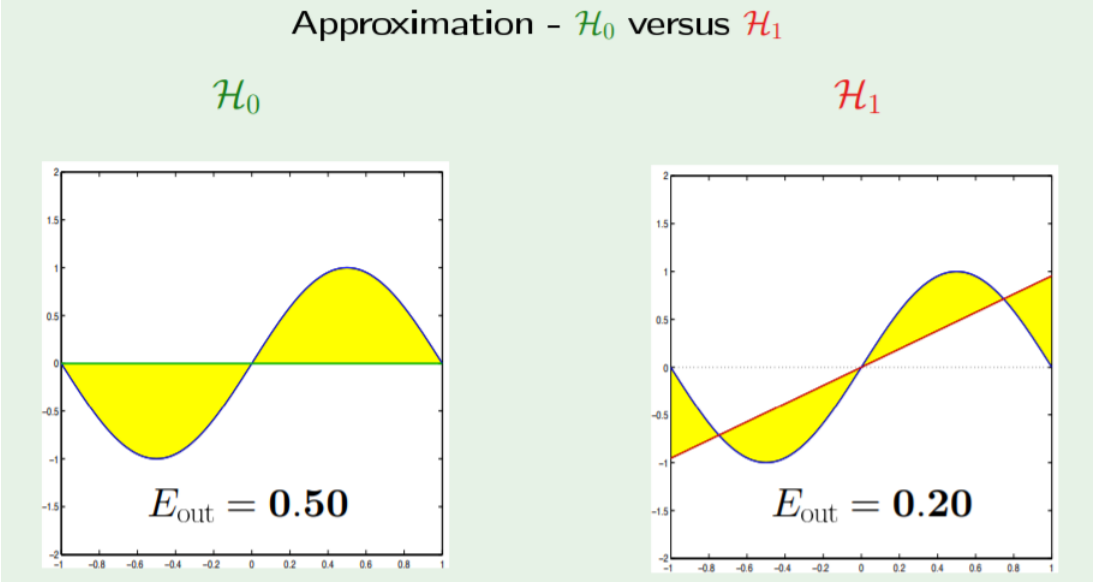
Machine Learning CS60050
Instructor: Dr. Aritra Hazra
Department of Computer Science and Engineering
Indian Institute of Technology,Kharagpur

Scribed by: Somalee Mitra(20RJ91R07)

Notes for the class on - 25^{th} February 2021
Part -II

1 Example

Suppose, the target function is a sine function. So, the output is a real number in the range (-1,+1). Say, there are only two training examples to predict the model.The two hypothesis are used to predict the model are
H0: h(x) = b and H1: h(x) = ax + b
Here we can see that H0 is a special case of H1 with a=0.
Now we have to determine which of these two hypothesis approximates better.



[The green line and red lines represents the approximated target functions using H0 and H1 respectively, whereas the black sine curve is the actual target function. The yellow area represents the error]

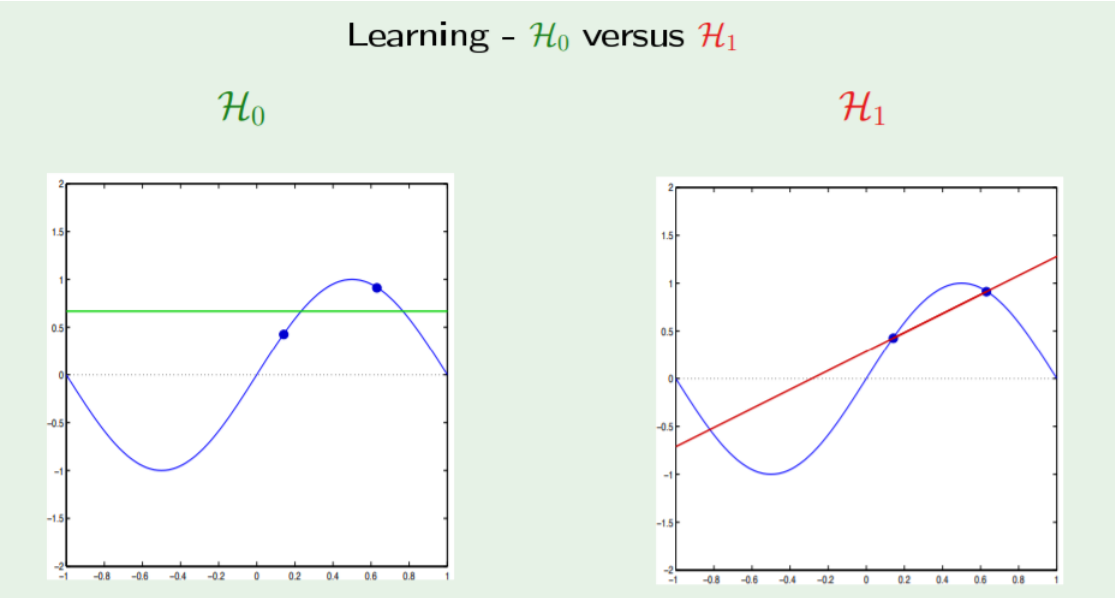
Now, as H0 is parallel to the X axis it gives error(E0H0)=0.50 and H1 is a straight line with gradient it gives much less error(E0H1)=0.20 than H0, given the function.

But in reality, we do not know what the unknown target function is.

So, instead, we try to learn the target function by fitting the given training points in a curve.

In the picture given above the green and red straight line represents the model which minimizes the error for the given two training examples for hypothesis 0 and hypothesis 1 respectively. So both the straight lines are good fit for the given two points.

But the target function is not a straight line but a sine curve.

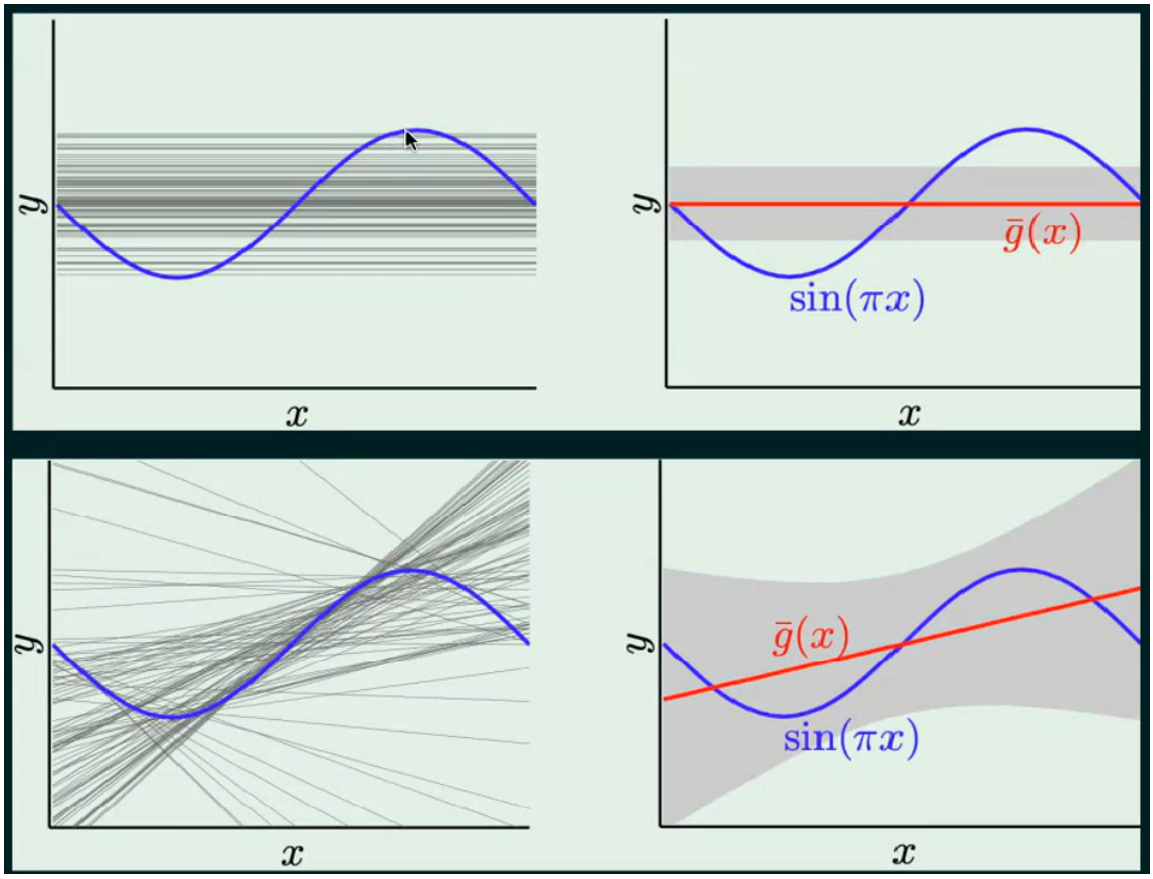


Now suppose we have D data points.

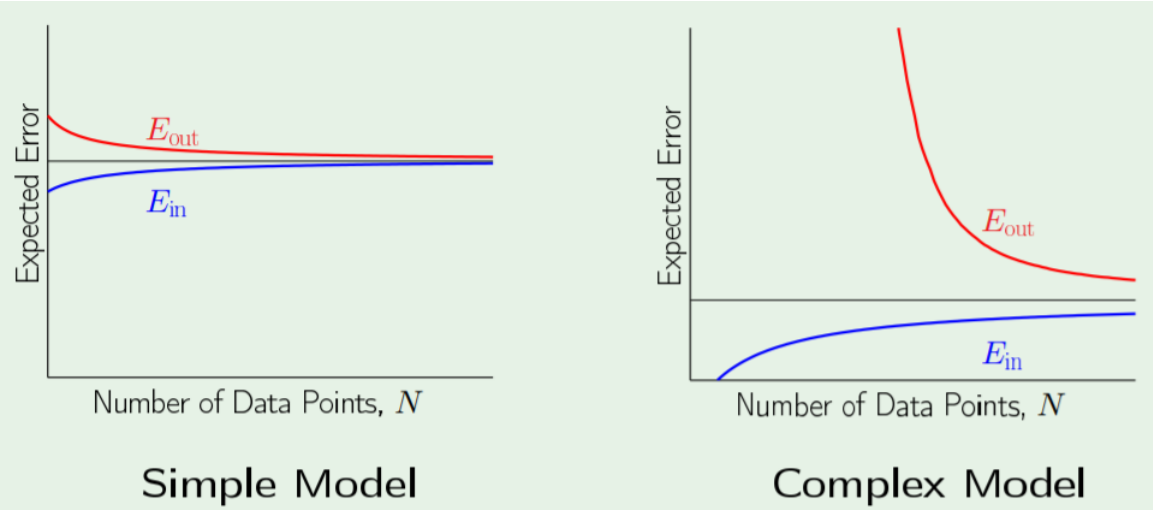
The figure above represents the bias and variance for two different hypothesis H0 and H1. The grey lines indicates how the given data points are distributed. If they are uniformly distributed at the both ends of the target curve, then we get a shaded region as given above (for hypothesis H0). Thus the density of the grey lines represents the bias of the data points. If they are not uniformly distributed , that is, if the density of the grey lines are more near any one end of the curve (instead of being denser in the middle region) it indicates the data points are not balanced. We take the mean of grey spectrum and find the average hypothesis $\bar{g}(x)$ in the shaded region.This shaded region represents the variance. We try to find out the best hypothesis among the possible region of average hypothesis. For hypothesis H1, we get a large variance when we consider the shaded region around the average hypothesis.If we analyse the bias and variance for the two hypothesis we get:

For H0: bias=0.5 variance=0.25 Eout(expected)= 0.75 For H1: bias=0.21 variance=1.69 Eout (expected)= 1.90

Eout(expected) is the out-of-sample error given by bias+variance.


y
x
y
$\bar{g}(x)$
$\sin(\pi x)$
x
y
x
y
$\bar{g}(x)$
$\sin(\pi x)$
x

So we can see that even if the bias is less for hypothesis H1, the variance is much larger for this hypothesis than the other.The expected outcome of out -of-sample error (which we want to minimize as much as we can) is much less for H0 than H1.
So, if we are given two data points and using two hypothesis we need to find out the best fitting curve the hypothesis H0 wins in a large margin than hypothesis H1. That is the probability of getting chosen as best fitting curve is much higher for H0.

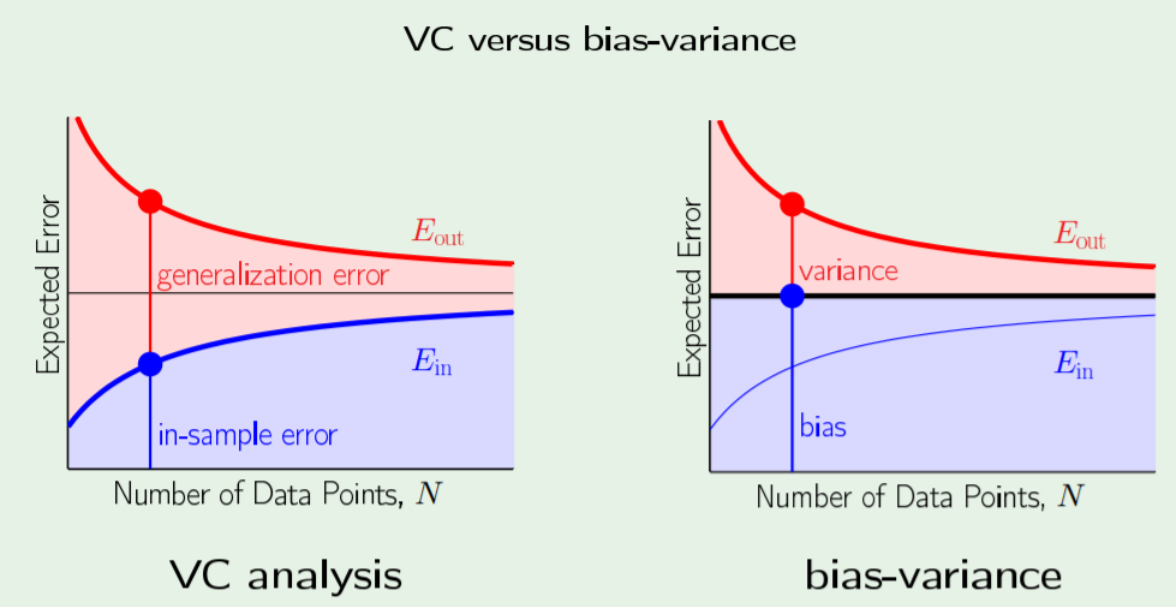


This signifies, that for hypothesis H1, we may have a good hypothesis, but finding that good hypothesis over a larger space is much more difficult. So, it is important how easily we find the best fit from the restricted set of chosen hypothesis, not how expressive the set of hypothesis is (if it is difficult to find the best among them).

Now we will compare the performance of a simple and a more complex model
In the figure above, it is shown that for a simple model the expected error is higher than a complex model.

For the complex model the in-sample-error for less number of data the model fits perfectly well, So at first we get Ein=0 and when it crosses the order of the complexity of the model,it starts to increase. For a few number of data points the out-of-sample error is infinite because here the machine here is memorising instead of learning and cannot get a generalised hypothesis.

How will it look if we compare the the same curve for VC (Vapnik-Chervonenkis) analysis and bias-variance diagram?



In the figure above, for the VC analysis the blue region represents in-sample error and the red region represents generalisation error. For the bias-variance diagram the blue region represents the bias (difference between the expected error created by the average hypothesis and target function) and the red region represents the variance (difference between the average hypothesis and the chosen hypothesis).

[1]

[1]Note 1. This scribe is based on lecture taught by Prof. Aritra Hazra on 25.02.2021 (11.00-11.55 am) in Machine Learning(CS60050) course.
2. All the figures in this document are taken either from slide-10e or handout-10d uploaded on Machine Learning(CS60050) course website.