Bias and Variance and Overfitting

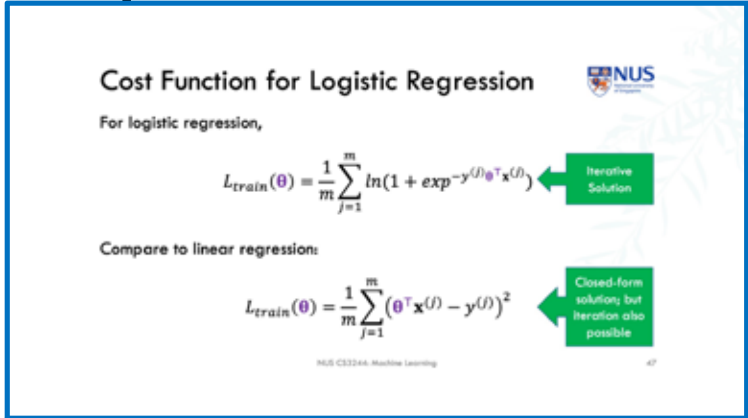
5

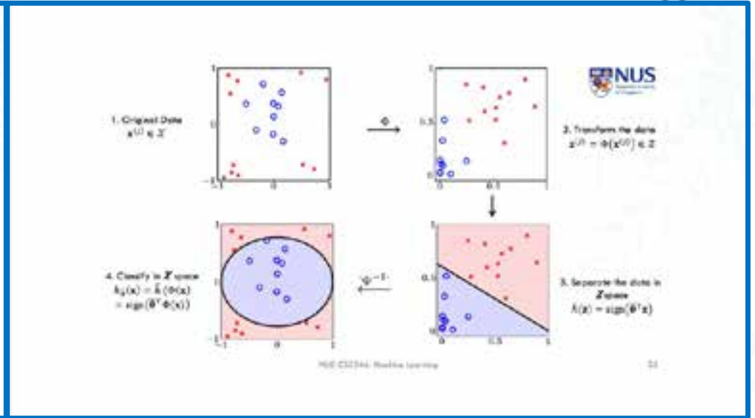
Post

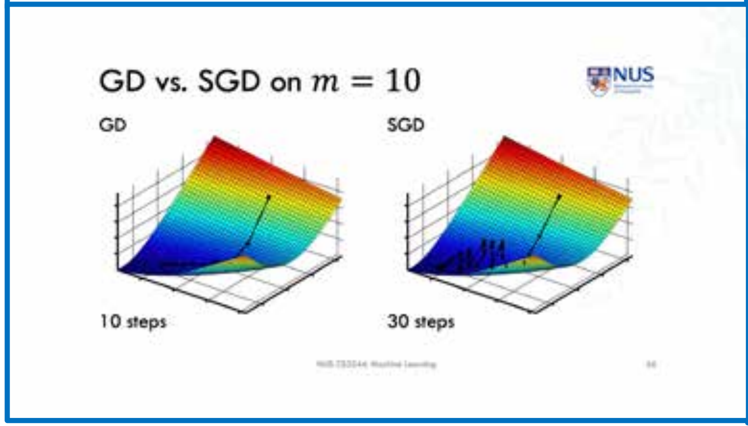
CS 3244
Machine Learning

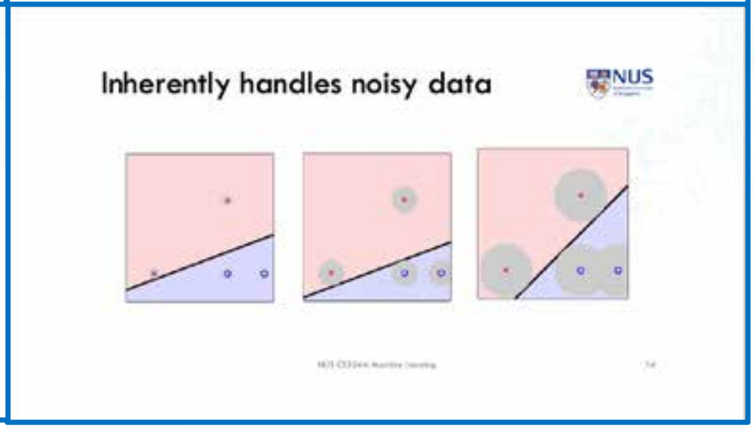
NUS National University of Singapore | Computing

Recap from Week 04









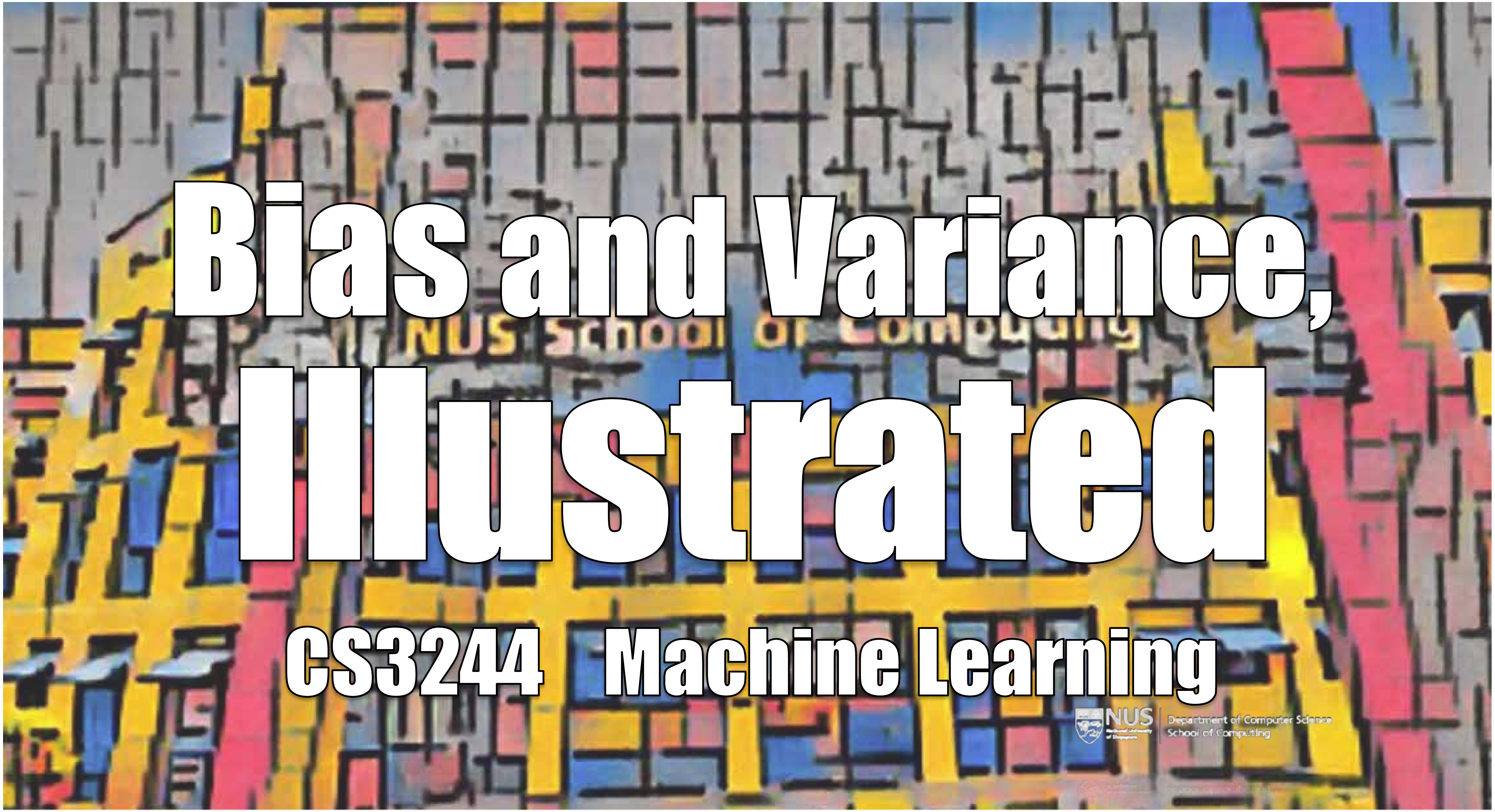
Forecast for Week 05

Learning Outcomes for this week:

- Understand the bias–variance tradeoff
- Understand why overfitting occurs: the role of model complexity and the sampled dataset
- Apply bias and variance decomposition in simple scenarios

Bias and Variance, Illustrated

CS3244 Machine Learning



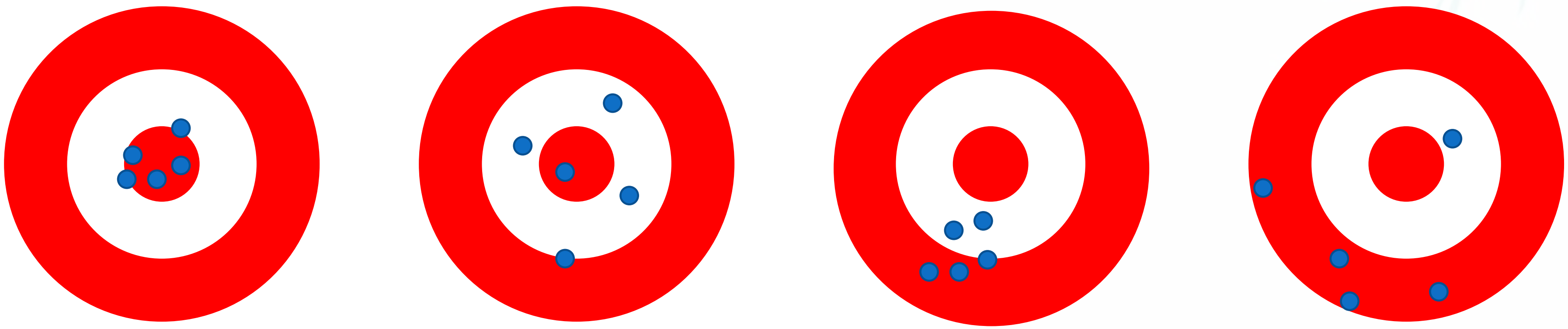
NUS
Department of Computer Science
School of Computing

Bias

The difference between the average prediction and the true value.

Variance

The variability of the model prediction, for given data. Tells us spread of our data.



Regressing the sine function

$f: [-1,1] \to \mathbb{R}$

$f(x) = \sin(\pi x)$

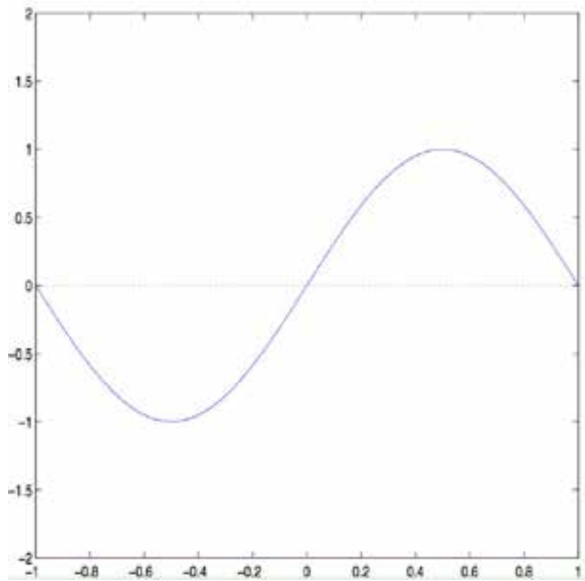
Only two training examples! $m = 2$

Two models used for learning:

🥇 $\mathcal{H}_1: h(x) = \theta_1 x + \theta_0$

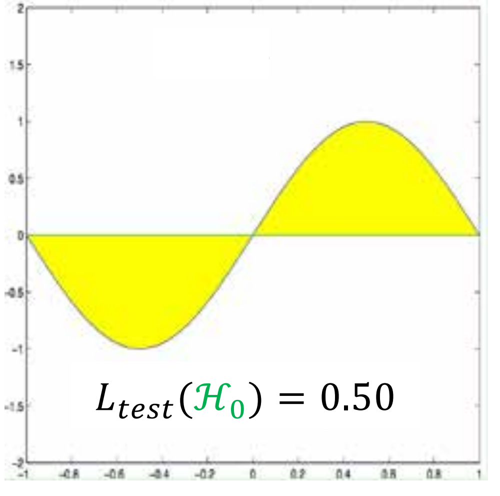
🥈 $\mathcal{H}_0: h(x) = \theta_0$

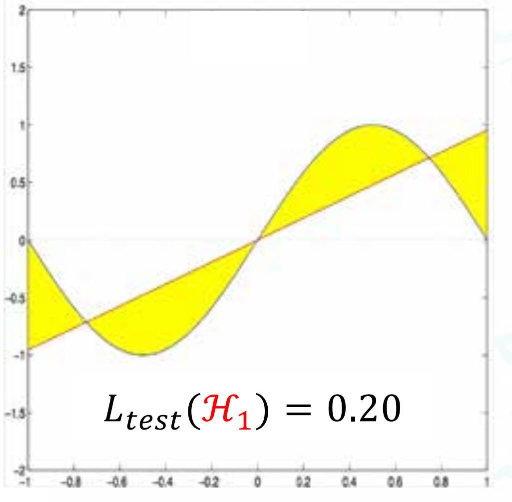
Your turn Q1: Which is better, $\mathcal{H}_0$ or $\mathcal{H}_1$?



Approximation – $\mathcal{H}_0$ versus $\mathcal{H}_1$

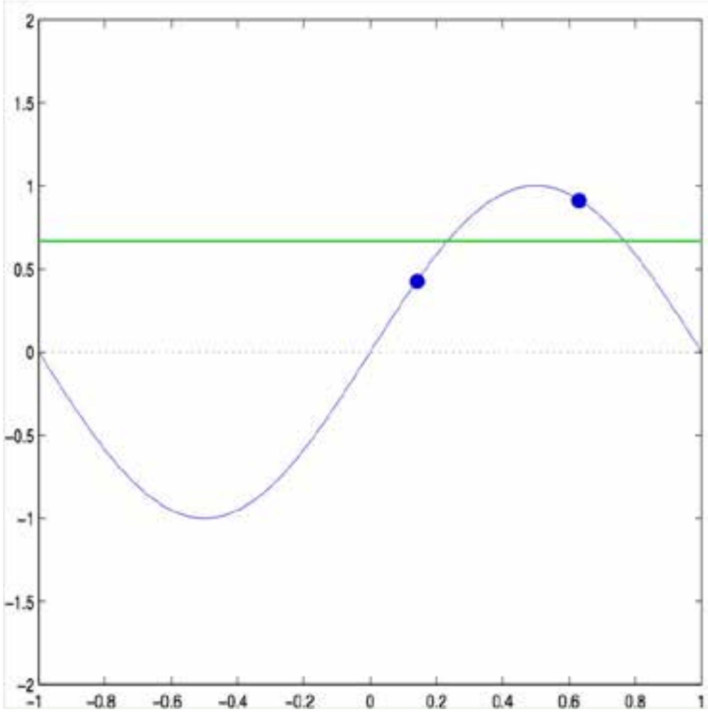
Use the full power of the model

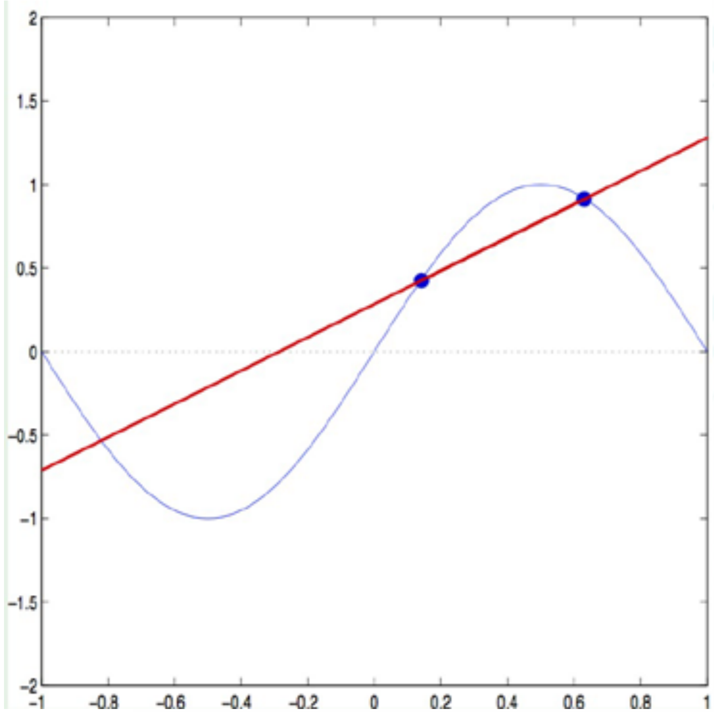




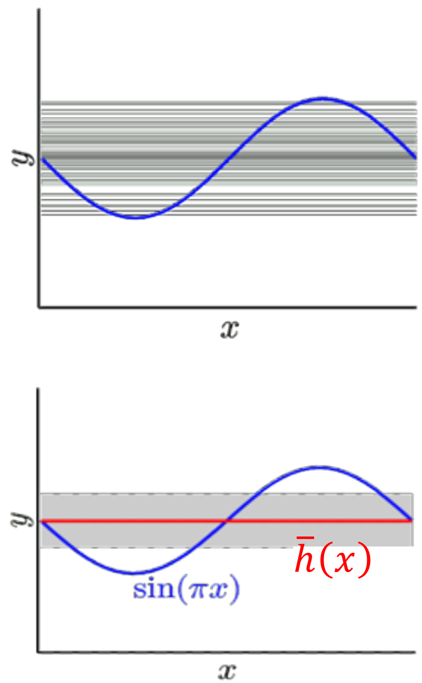
where — and — are the $\bar{h}(x)$ of each $\mathcal{H}$.

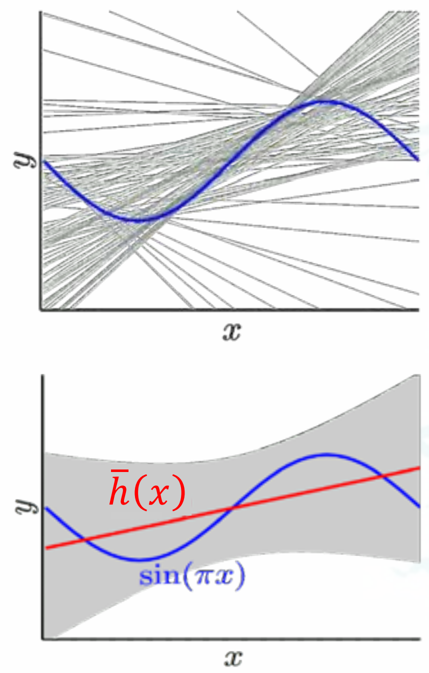
Learning – $\mathcal{H}_0$ versus $\mathcal{H}_1$





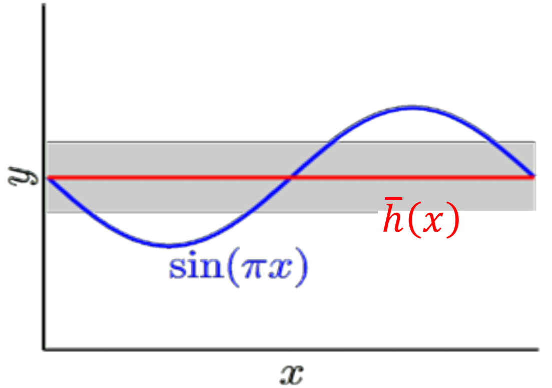
Bias & variance - $\mathcal{H}_0$ versus $\mathcal{H}_1$



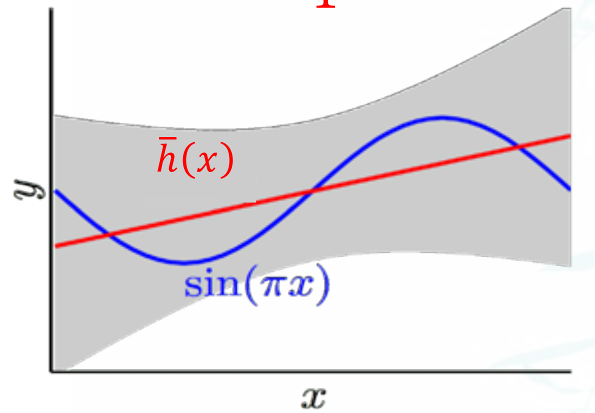


And the winner is …

$\mathcal{H}_0$



$\mathcal{H}_1$



Bias = 0.50

Var = ?

Bias = 0.21

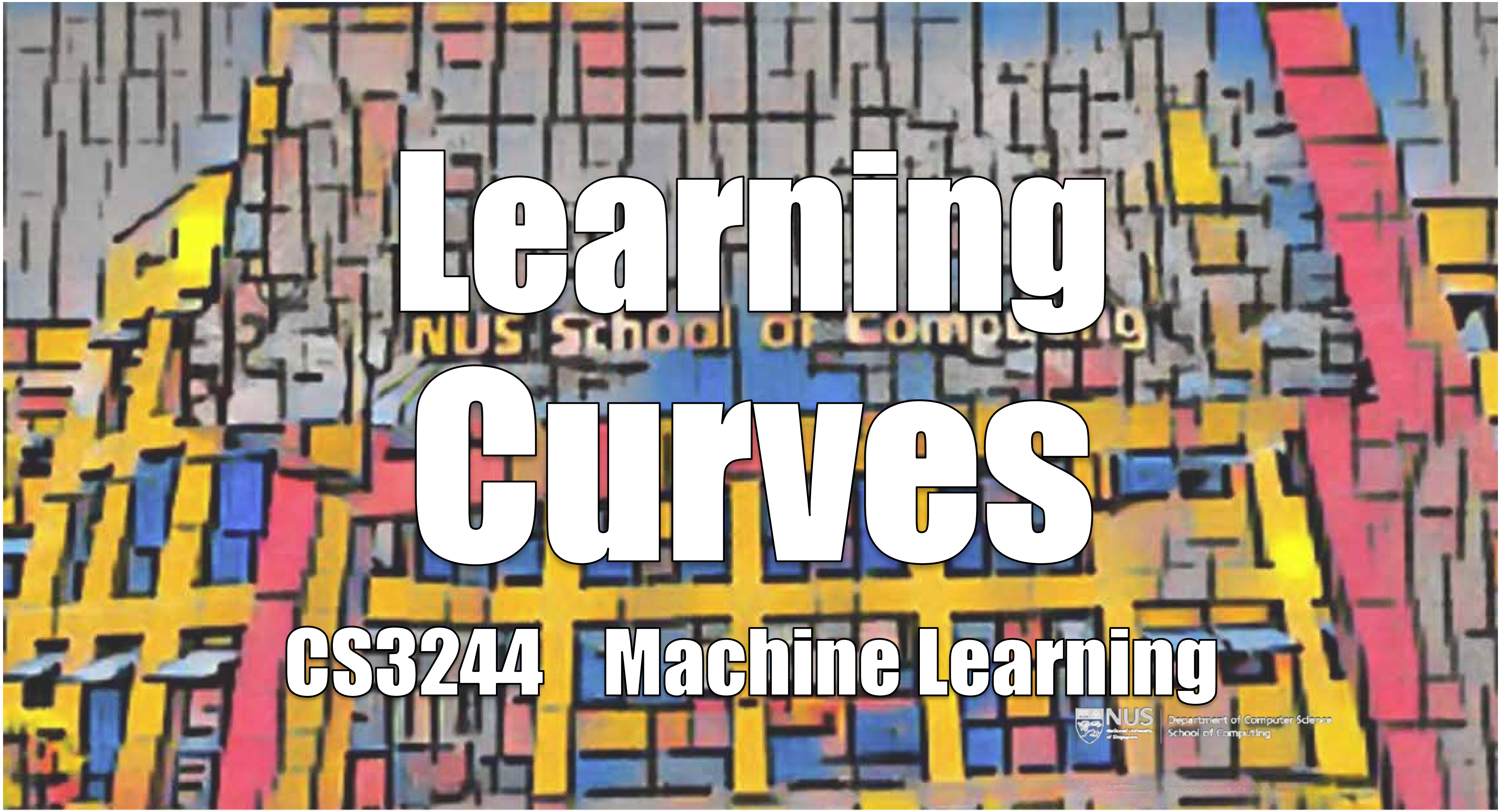
Var = ?

Lesson learned

Match the 'model complexity' to the

data resources, not to the **target complexity**.

Quality and Quantity


Learning Curves
CS3244 Machine Learning
NUS
Department of Computer Science
School of Computing

Learning Curves

Plot expected L_{test} and L_{train} …

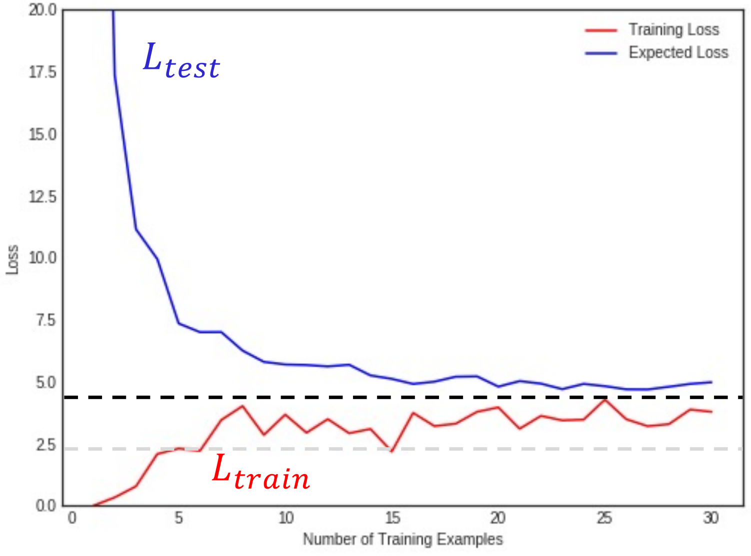
…as we vary size m

Data Resources: Quality and Quantity

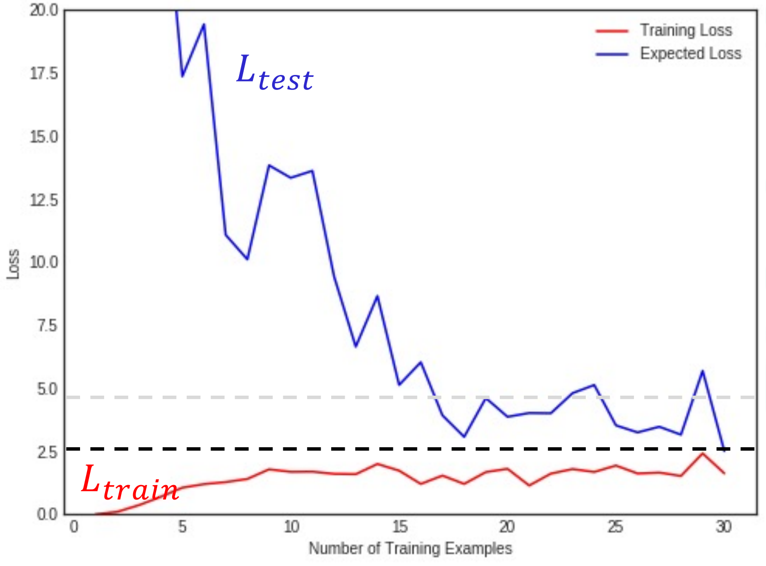
Expected test cost $\mathbb{E}_{\mathcal{D}}[L_{test}(f_{\mathcal{D}})]$

Expected training cost $\mathbb{E}_{\mathcal{D}}[L_{train}(f_{\mathcal{D}})]$

Simple versus Complex Models

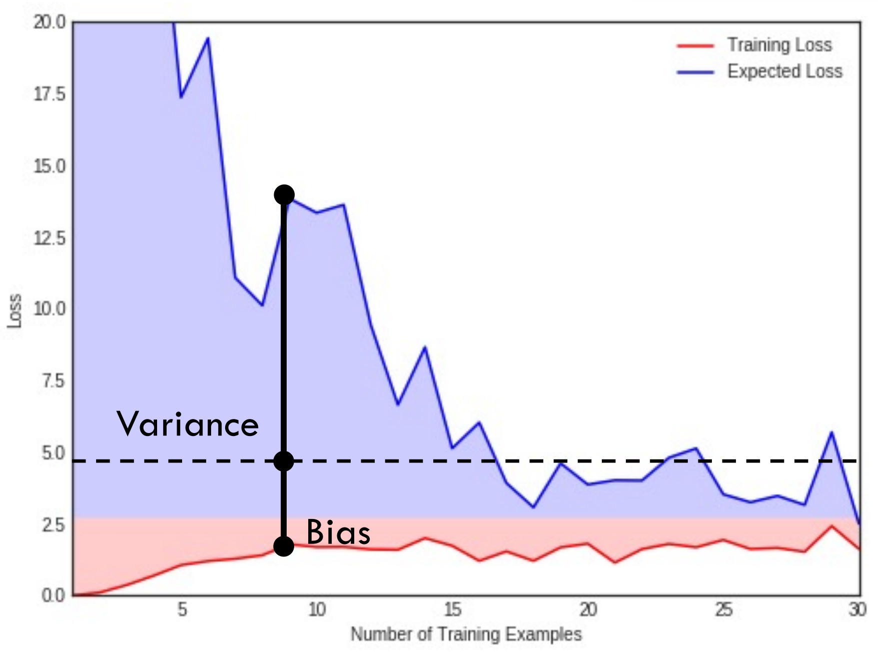


Simple Model



Complex Model

Bias–Variance on a Curve



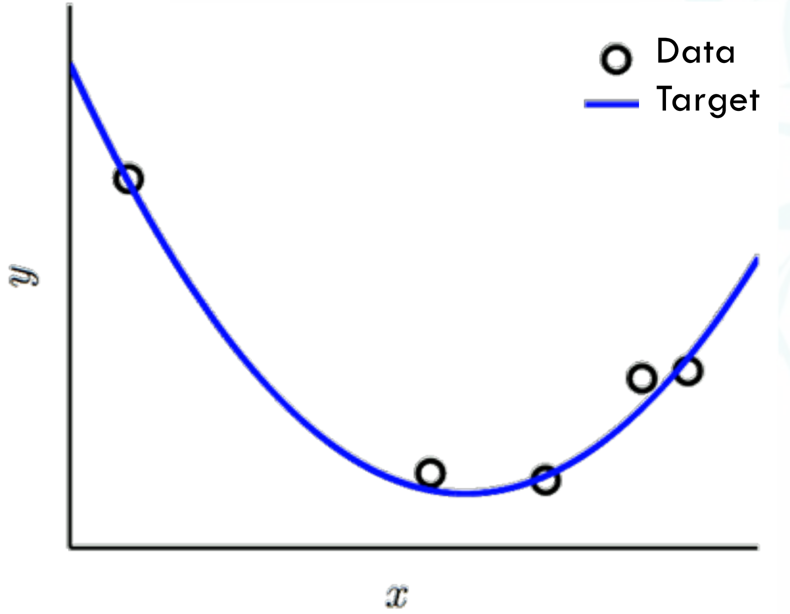
Overfitting

CS3244 Machine Learning

Overfitting, Illustrated

Simple target function

5 data points – **slightly noisy**



4^{th}-order polynomial fit

$L_{train} = 0$ but L_{test} is huge

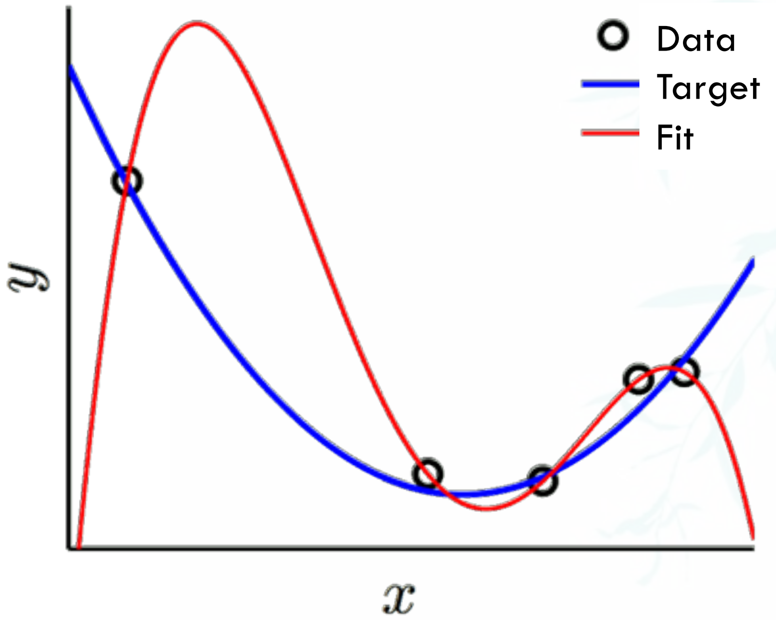
Overfitting, Illustrated

Simple target function

5 data points – **slightly noisy**

4^{th}-order polynomial fit

$L_{train} = 0$ but L_{test} is huge



The culprit

Overfitting:
"Fitting the data more than is warranted"

Culprit: fitting the noise.

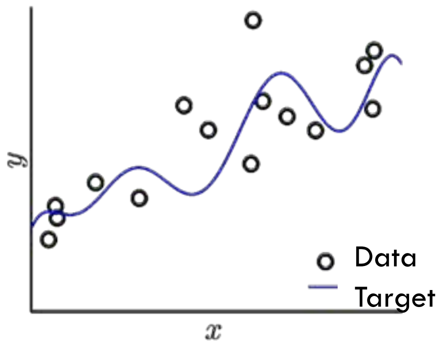
Not just useless,

but harmful

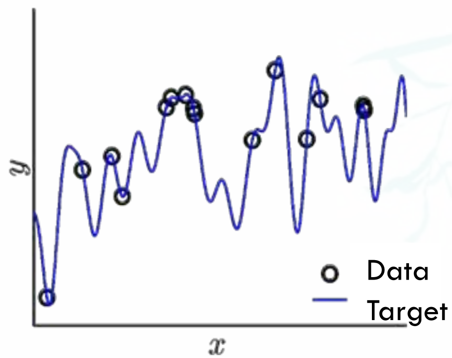
🥈 2nd versus 🥇 10th order polynomials

In Zoom breakout or physical subgroups, discuss and react 🏃🏃🏃:
(5 mins): Pick your model of choice for each scenario. In each scenario, you are given 15 data points. Justify your choice.

Q2: 10th-order target + noise

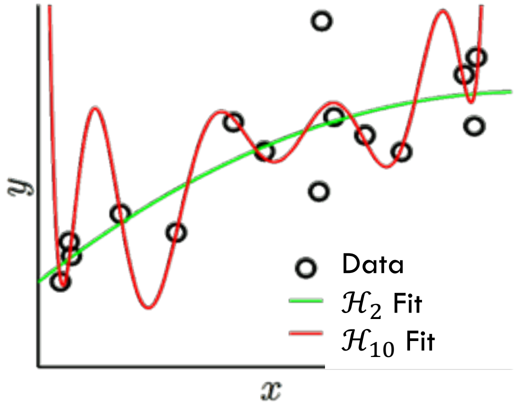


Q3: 50th order target, noiseless



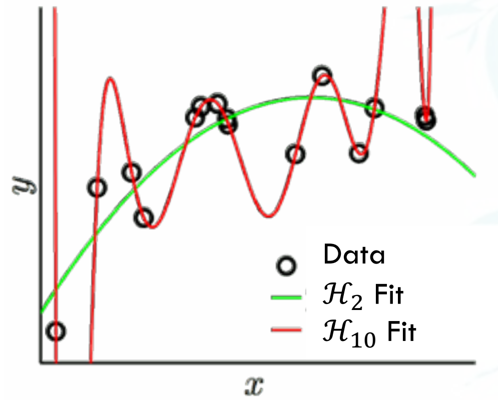
Two fits for each function

Noisy low-order target



	2nd Order	
L_{train}	0.050	
L_{test}	0.127	

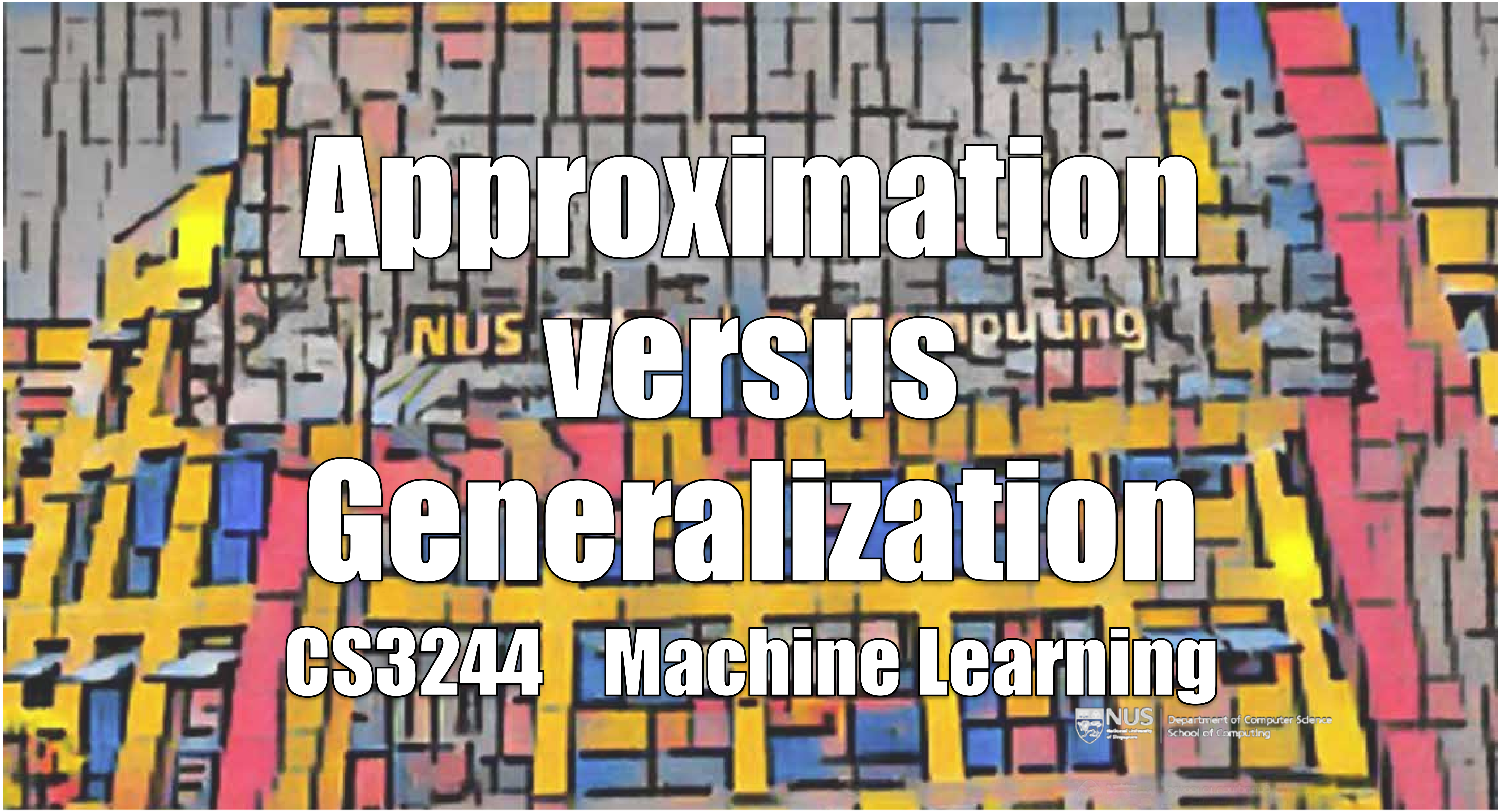
Noiseless high-order target



	2nd Order	
L_{train}	0.029	
L_{test}	0.120	

Approximation versus Generalization

CS3244 Machine Learning



Approximation–generalization tradeoff

Small L_{test}: good approximation of f on unseen test data

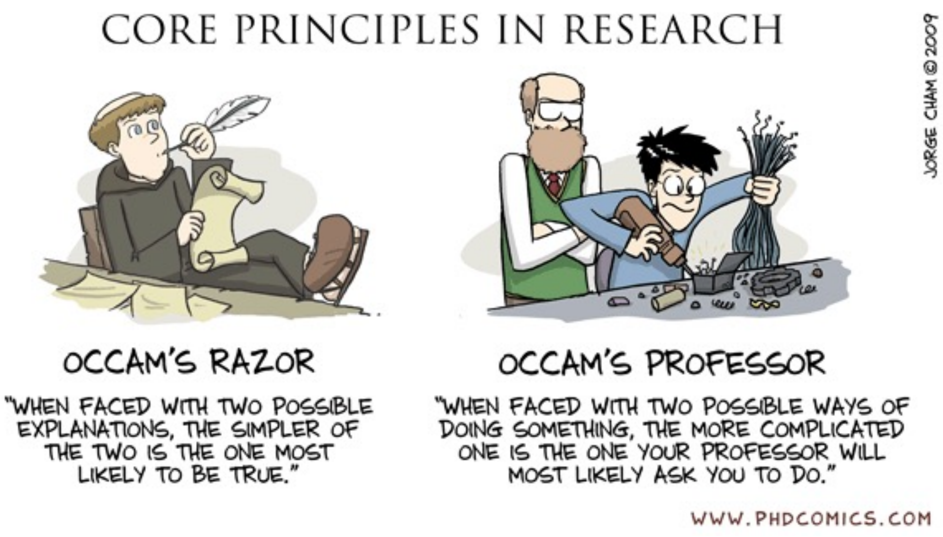
More complex $\mathcal{H}$ → better chance of **approximating** f

Less complex $\mathcal{H}$ → better chance of **generalizing** on test data

Ideal $\mathcal{H} = \{f\}$

Good luck with that!

Occam's Razor



"Simpler is better"

Formalized as Minimum Description Length (MDL)

The shortest description of the data is the best model (related to Entropy)

Last Week's Assigned Task

Post a 1–2 sentence answer to the topic in your tutorial group:
#tg-xx

Describe kNN or decision trees with respect to variance.

Pre-Lecture Activity from last week

The kNN model is very susceptible to variance in the training data especially for low values of k since the algorithm only considers the nearest datapoints, it is easily affected by variance in the region, not being able to recognise the bigger trend to be able to generalise well. While the impact of variance can be reduced by increasing the value of k, the tradeoff here will be that the model will less likely to draw a boundary when necessary.

kNN: As discussed in previous lectures, there seems to be an inverse correlation with the value of `k` and the variance of the model. For instance, a low value of `k` e.g. 1-NN results in a model which has very high variance, meaning it performs the best on training data but does not generalize well to test data.

for kNN with small k, the variance tends to be high; the reverse happens to larger k. This is because relying on fewer neighbors is more risky and may give high variance in results

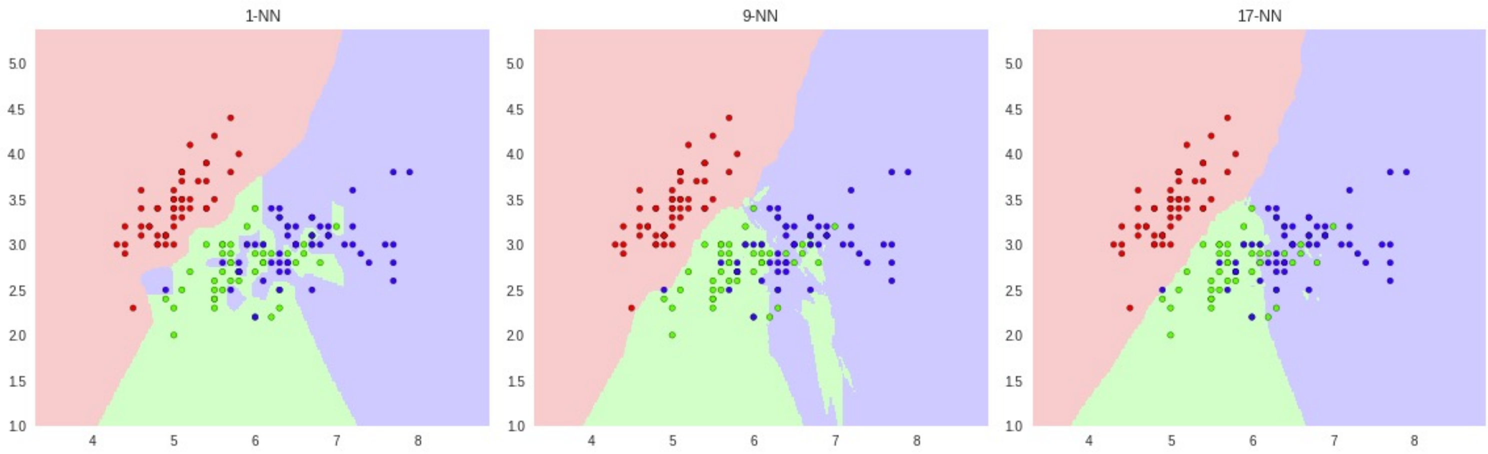
The variance of kNN depends on the value of K chosen. If K is 1, the model essentially remembers the training data resulting in overfitting since it would not be able to generalise to new data, hence, variance will be high. If K becomes bigger, the model would have to consider more neighbours and thus better able to discover underlying pattern that is generalisable to new data, hence, resulting in lower variance. (edited)

For kNN, a model with high variance pays a lot of attention to training data. Hence when k increases, variance decreases as it will look for more points near it and generalize it more while compared to k=1 where it will only look for the nearest point and increases its sensitivity to training data.

In k NN

Model free: Data completely dictates the model

Higher k lowers the dependence of the model on a particular data point, makes model more robust and lowers variance



A decision tree which is not pruned will be very prone to overfitting as noises or small changes to the input data can cause very different outputs. Hence, causing a high variance in its output prediction.

M

DTs are prone to overfitting when are not pruned, leading to high variance as the noise from the dataset is captured.

For decision trees, it seems that they are prone to overfitting with high variance. This is because as we build deeper, it will fit the training data better, but generalizes less to new test data. To reduce this variance, we can perform pre-pruning to stop growing the tree earlier before it perfectly classifies the training set or perform post-pruning according to an optimization metric.

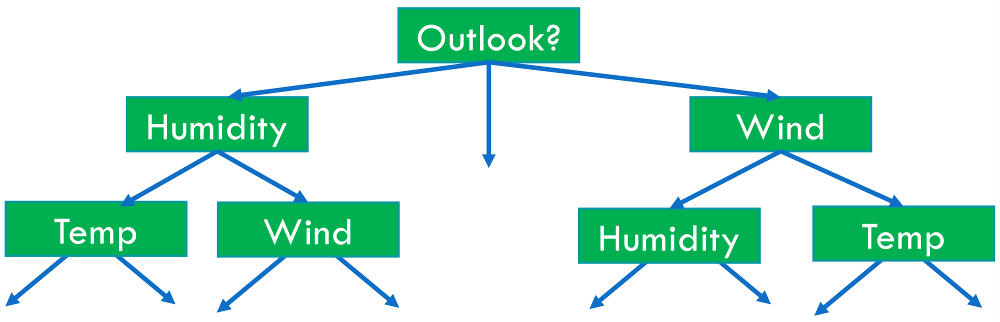
One factor that may affect the variance of decision trees is the tree depth. As tree depth increases, we may be overfitting, with the data spaces being segmented into smaller and smaller areas. This represents higher variance. Source

> **saas.berkeley.edu**
> SAAS - Education Committee
> SAAS is here to provide Berkeley undergraduates interested in Statistics with resources and a statistics-driven community

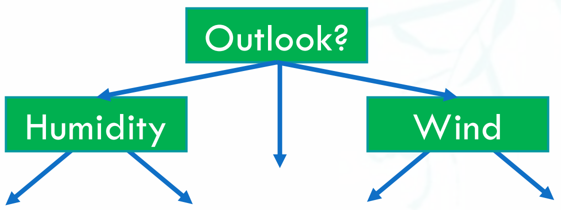
...In Decision Trees

Pruning discards detailed tests that may use criteria non-essential for classification in test data.

Before



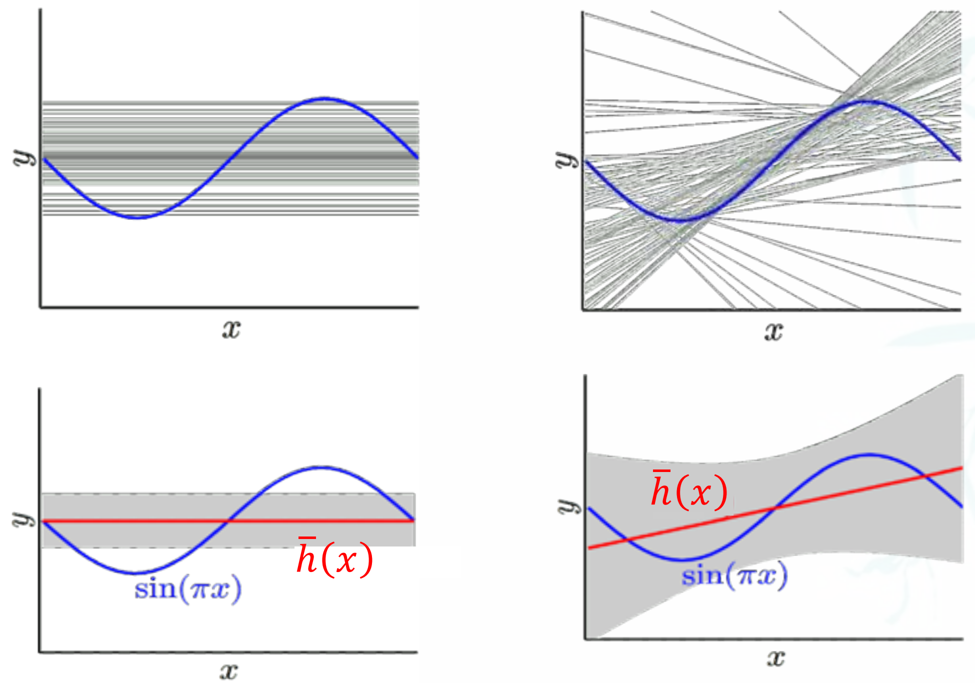
After



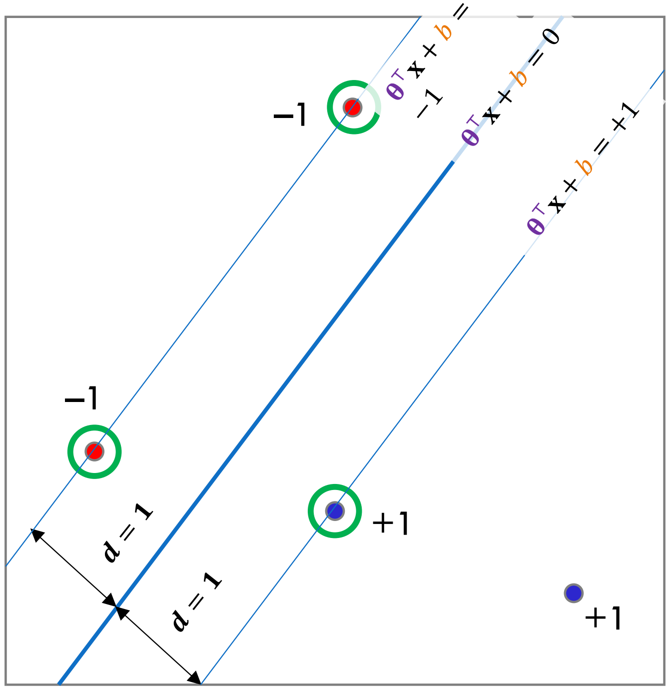
…In Linear Models

We just did this!

Each additional parameter θ_i adds a degree of freedom in the model.



…In SVM



The SVM determines its separating hyperplane by virtue of its support vectors.

# of support vectors ∝ complexity of the model

… in Ensembles

We have T reports $h_1, h_2, \ldots, h_T$ predicting whether a stock will go up as $h(x)$.

We can:

1. Select the most trustworthy of them based on their usual performance
   Training performance: $h(x) = h_{t^*}(x)$ with $t^* = argmin_{t\in\{1,2,\ldots,T\}} L_{train}(h_t^-)$

2. Let each report have a vote
   Uniform Vote: $h(x) = \text{sign}\left(\sum_{t=1}^{T} 1 \cdot h_t(x)\right)$

3. Weight the reports non-uniformly
   Weighted Vote: $h(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t \cdot h_t(x)\right)$ where $\alpha_t \geq 0$.

4. Combine the predictions conditionally
   This is decision trees, let's finish it up now!

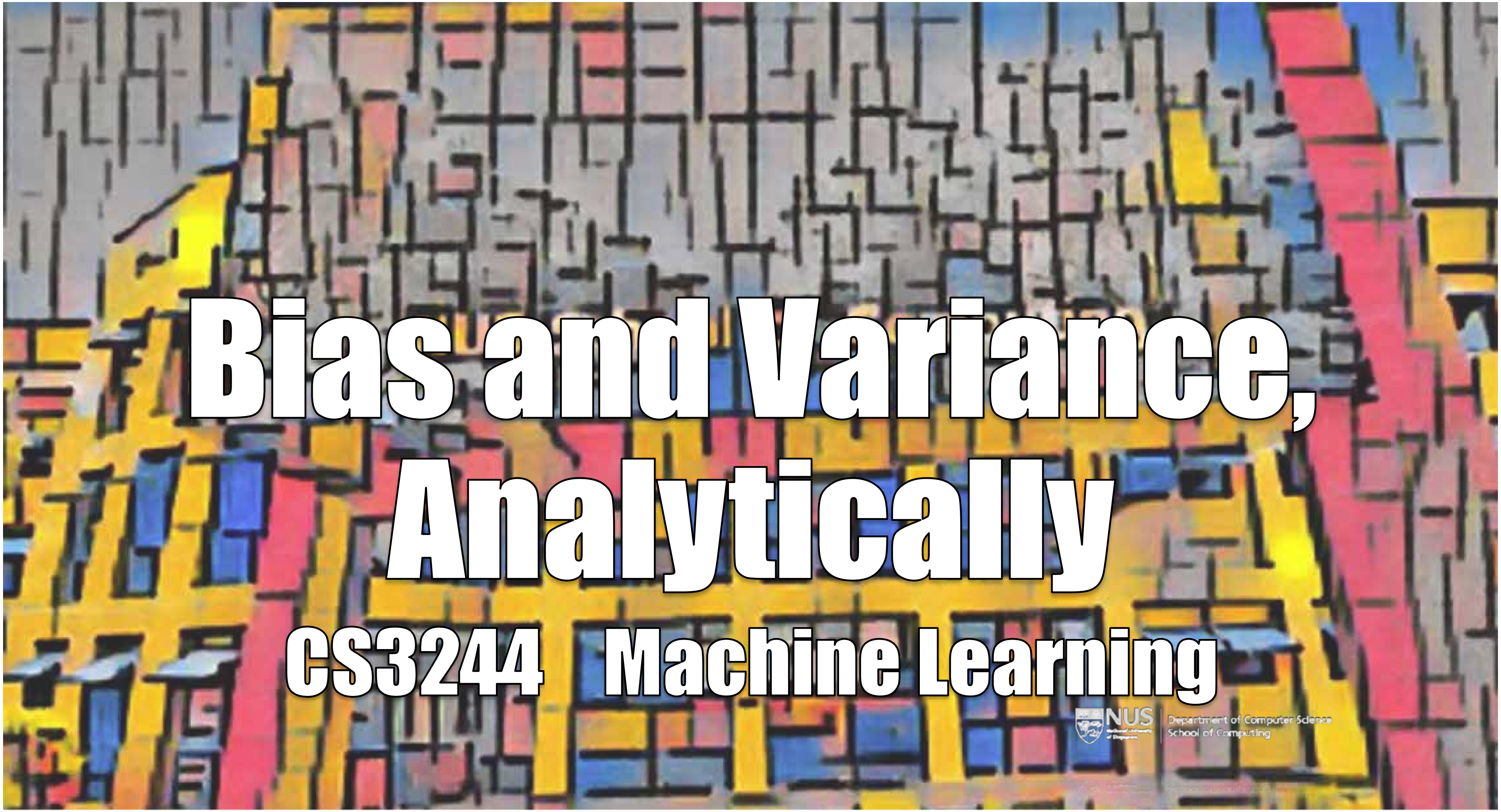
Key: component hypotheses are
Diverse.

Ensembling diverse h generalizes better.

🆒 **fact:** Decision trees have a native version: **Random forests.**

Bias and Variance, Analytically

CS3244 Machine Learning



NUS National University of Singapore | Department of Computer Science School of Computing

Quantifying the tradeoff

Decomposing L_{test} into

1. How well $\mathcal{H}$ can approximate f overall
2. How well we can zoom in on a good $h_\theta \in \mathcal{H}$

Applies to **real-valued** targets and uses **squared error**

There is an equivalent binary version through the lens of VC analyses (related to SVM)

Recap: Expected Value

We'll need to reason about values of data, irrespective of particular samples, so we'll need to work with expected values of random variable.

Expected Value: Average over all possible outcome of a random variable X.

$$\mathbb{E}[X] = \sum_{x \in X} x \Pr[X = x]$$

Example

Let X be the outcome of a fair dice roll. Then the expected value is $\mathbb{E}[X] = \frac{1}{6}(1 + 2 + 3 + 4 + 5 + 6) = 3.5$.

Variance

Intuition

Quantify how much deviation from expected value.

$$\mathrm{Var}[x] = \mathbb{E}[(x - \mathbb{E}[x])^2]$$

Example

Let x be the outcome of a fair dice roll. Then the variance is $\mathrm{Var}[x] = \frac{1}{6}\left((1 - 3.5)^2 + \cdots + (6 - 3.5)^2\right) = \frac{35}{12}$.

Some Terminology

- $\mathbb{E}_x[z(x)] \equiv$ Expected value of $z()$, given the distribution of values of x.

- $h_{\mathcal{D}} \equiv$ Hypothesis of learner when learning from Dataset $\mathcal{D}$.

- $\mathbb{E}_{\mathcal{D}}[z()] \equiv$ Expected value of $z()$, given distribution of possible Datasets $\mathcal{D}$.

Start with L_{test}

$$L_{test}(h_{\mathcal{D}}) = \mathbb{E}_x[(h_{\mathcal{D}}(x) - f(x))^2]$$

$h(x)$ depends on the specific dataset $\mathcal{D}$.

Start with L_{test}

$$L_{test}(h_{\mathcal{D}}) = \mathbb{E}_x[(h_{\mathcal{D}}(x) - f(x))^2]$$

$h(x)$ depends on the specific dataset $\mathcal{D}$.

$$\mathbb{E}_{\mathcal{D}}[L_{test}(h_{\mathcal{D}})]$$

Generalizing over all $\mathcal{D}$ with same m

$$= \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x[(h_{\mathcal{D}}(x) - f(x))^2]\right]$$

$$= \mathbb{E}_x\left[\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2]\right]$$

Swap ok, as integrand is strictly non-negative

Focus on $\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2]$, get the outer term $\mathbb{E}_x[\dots]$ later.

The average hypothesis

To evaluate $\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2]$,
we define the 'average' hypothesis $\bar{h}(x) = \mathbb{E}_{\mathcal{D}}[h_{\mathcal{D}}(x)]$

Imagine many, many data sets $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K$ drawn:

$$\bar{h}(x) \approx \frac{1}{K} \sum_{k=1}^{K} h_{\mathcal{D}}(x)$$

Using $\bar{h}(x)$

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] =$$

Using $\bar{h}(x)$

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] =$$

Add $-\bar{h}(x) + \bar{h}(x)$. Associate first two and second two terms.

$$= \mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x) + \bar{h}(x) - f(x)\right)^2\right]$$

$$= \mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2 + \left(\bar{h}(x) - f(x)\right)^2 + 2\underbrace{\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)}_{0}\underbrace{\left(\bar{h}(x) - f(x)\right)}_{\text{const w.r.t. } \mathcal{D}}\right]$$

Expand out, cross terms drop as first part of term is 0.

$$= \mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right] + \left(\bar{h}(x) - f(x)\right)^2$$

2nd term is constant with respect to $\mathcal{D}$

Bias and Variance

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] = \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right]}_{\text{Q4}} + \underbrace{\left(\bar{h}(x) - f(x)\right)^2}_{\text{Q5}}$$

Quick Quiz: Which is which?

1️⃣ Bias
2️⃣ Variance
🤷‍♀️🤷🤷‍♂️ I give up

Bias and Variance

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] = \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right]} + \underbrace{\left(\bar{h}(x) - f(x)\right)^2}$$

Therefore,

$$\mathbb{E}_{\mathcal{D}}[L_{test}(h_{\mathcal{D}})] = \mathbb{E}_{x}\left[\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2]\right]$$

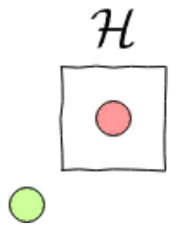
$$= \mathbb{E}_{x}[\mathrm{bias}(x) + \mathrm{var}(x)]$$

$$= \mathrm{bias} + \mathrm{var}$$

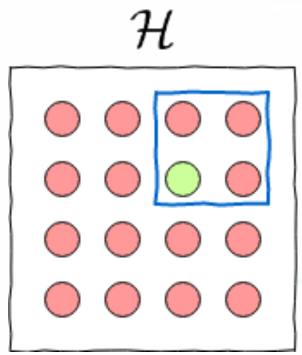
The tradeoff

$$\mathrm{Bias}[h(x)] = \left|\bar{h}(x) - f(x)\right|$$

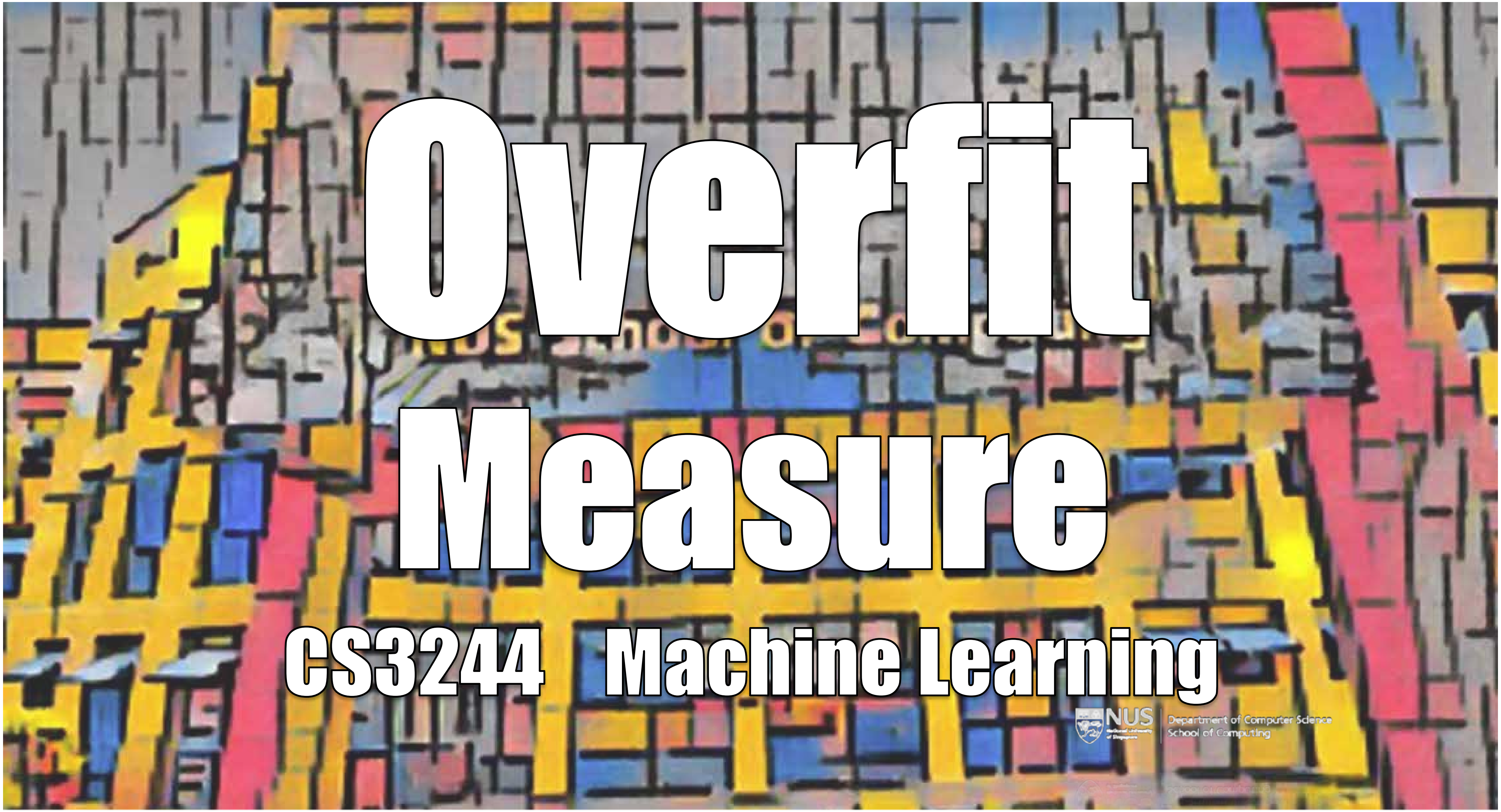
$$\mathrm{Var}[h(x)] = \mathbb{E}\left[\left(h(x) - \bar{h}(x)\right)^2\right]$$



$\mathcal{H}_{small}$: just one hypothesis.



$\mathcal{H}_{big}$: big, contains the target function f. But you must find it.


Overfit
Measure
CS3244 Machine Learning
NUS
Department of Computer Science
School of Computing

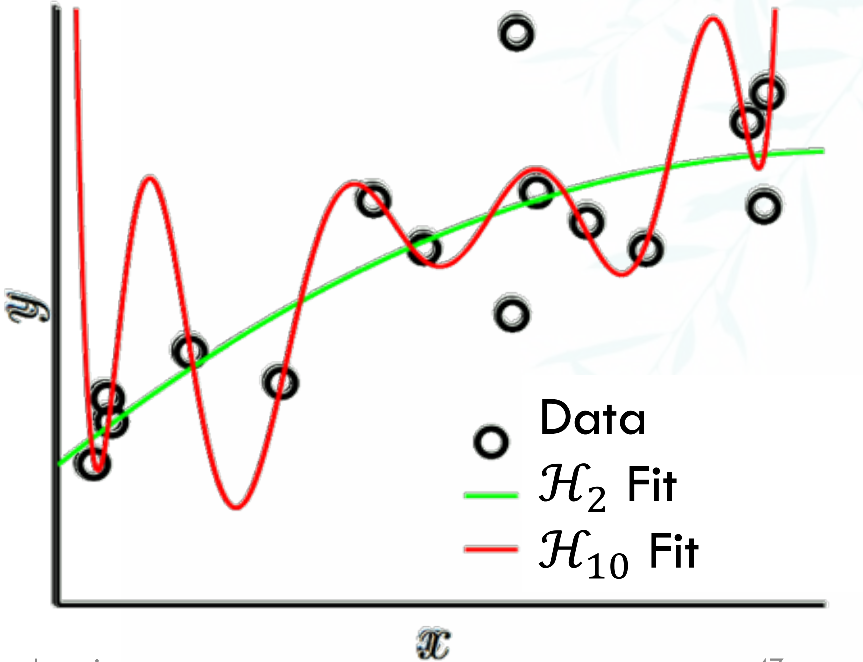
10^{th} order polynomial with noise

Two learners Overfitting and Restrained:

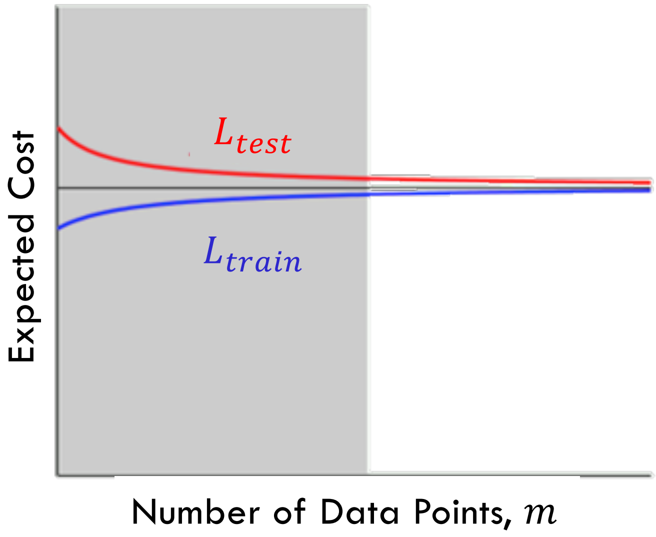
Both know the target is 10^{th} order, you get 15 data points.

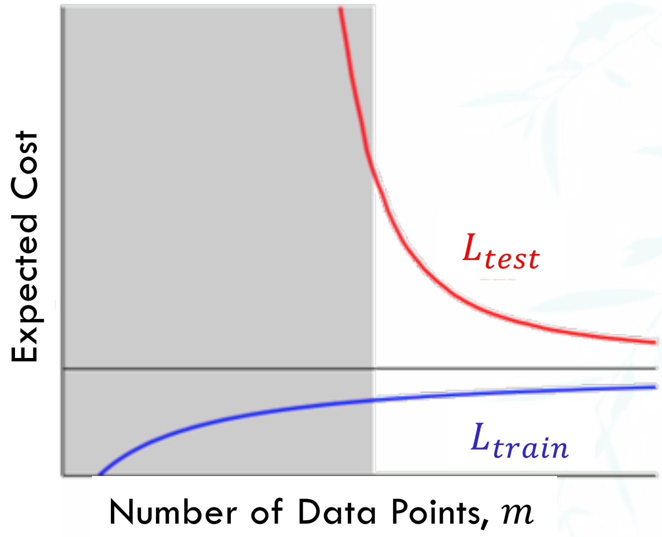
- O chooses $\mathcal{H}_{10}$
- R chooses $\mathcal{H}_2$

Your Turn: Which fits better?



When is $\mathcal{H}_2$ better than $\mathcal{H}_{10}$?





Overfitting (grey region): $L_{test}(\mathcal{H}_{10}) > L_{test}(\mathcal{H}_2)$

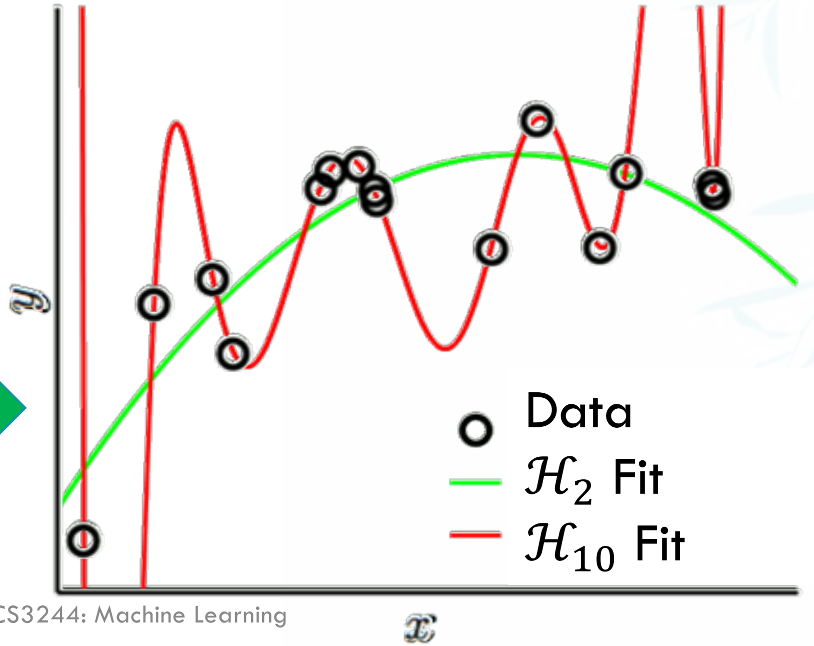
Without noise, with complex f

The two learners $\mathcal{H}_{10}$ and $\mathcal{H}_2$

They <u>know</u> there is no noise

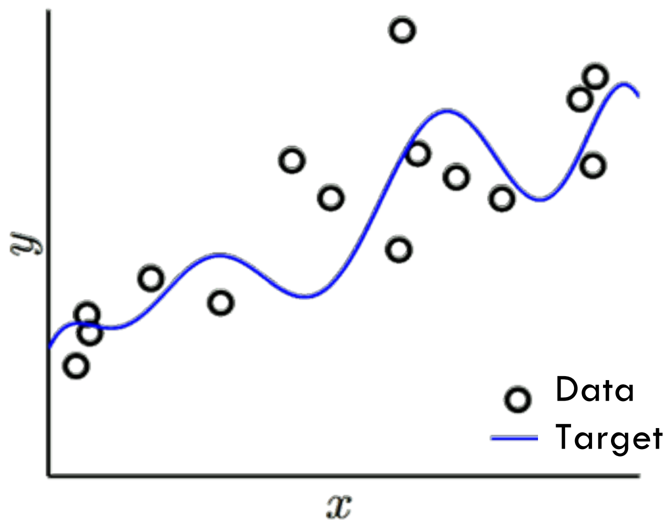
Is there really no noise?

Learning a 50th order target



A detailed experiment

Impact of **noise level** and **target complexity**



$$y = f(x) + \epsilon(x) = \underbrace{\sum_{q=0}^{Q_f} \alpha_q x^q}_{*} + \underbrace{\epsilon(x)}_{\sigma^2}$$

Noise level: σ^2

Target complexity: Q_f

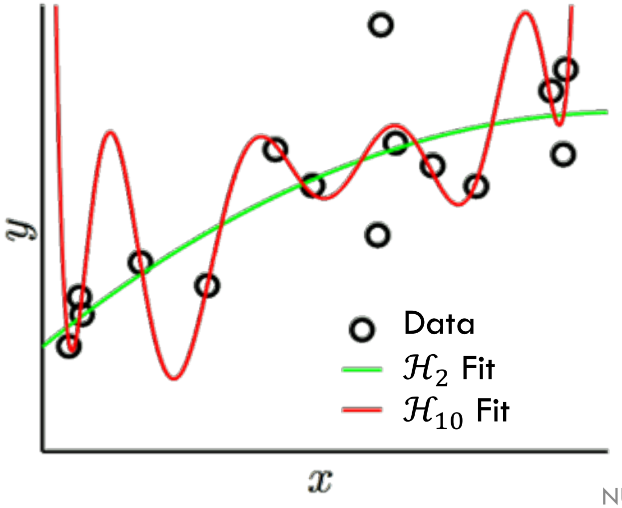
Data set size: m

The overfit measure

We fit the data set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$ using our 2 models:

$\mathcal{H}_2$: 2nd order polynomials

$\mathcal{H}_{10}$: 10th order polynomials

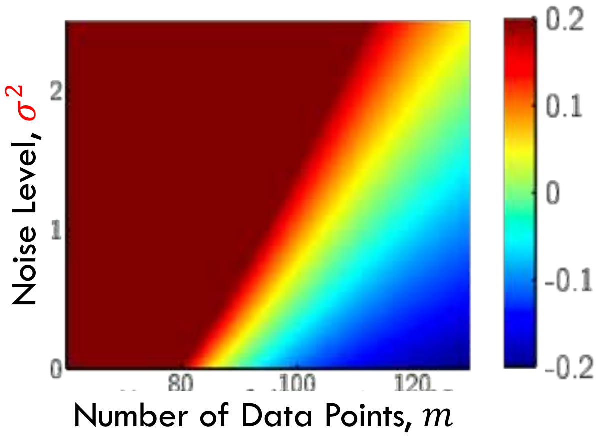


Compare out-of-sample errors of $h_2 \in \mathcal{H}_2$ and $h_{10} \in \mathcal{H}_{10}$

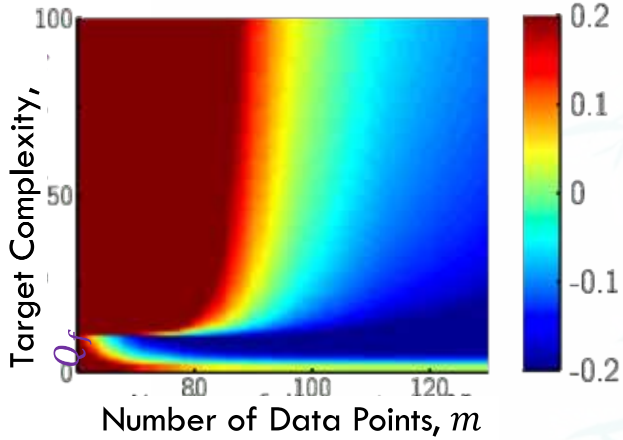
Overfit measure

$L_{test}(h_{10}) - L_{test}(h_2)$

Overfit measure: $L_{test}(h_{10}) - L_{test}(h_2)$

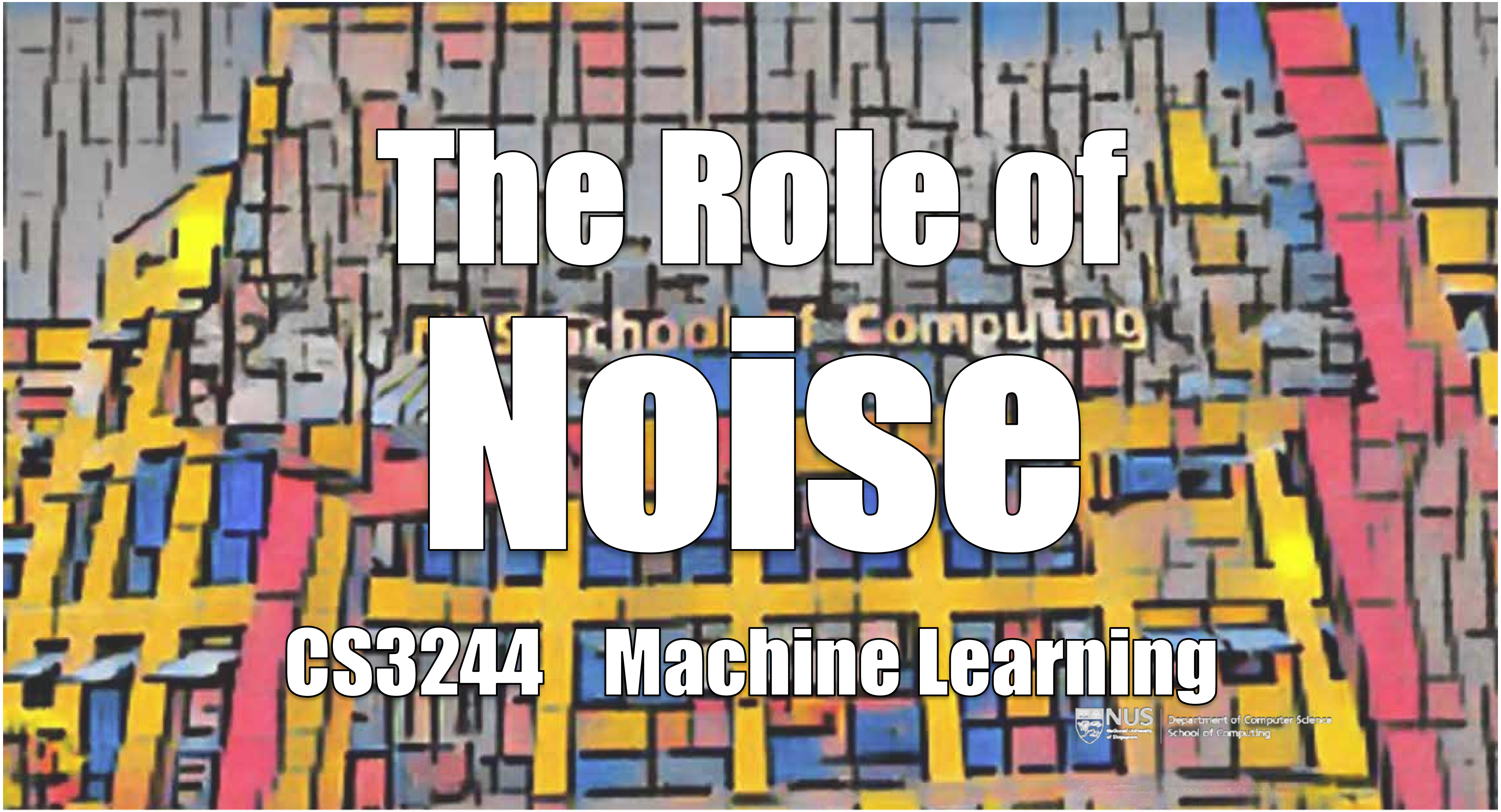


Stochastic Noise



Deterministic Noise

Number of data points	↑	Overfitting ↓
Stochastic Noise	↑	Overfitting ↑
Deterministic Noise	↑	Overfitting ↑


The Role of
Noise
CS3244 Machine Learning
School of Computing
NUS
Department of Computer Science
School of Computing

Noise

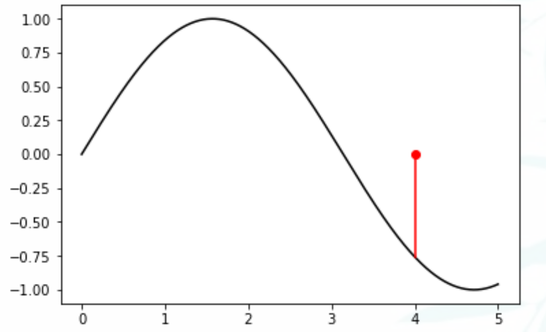
That part of y that we **cannot** model

It has two sources …

Stochastic Noise: Data Error

We would like to learn f

$$y_j = f(\mathbf{x}_j)$$



Unfortunately, we actually observe $\mathcal{D}$

$$y_j = f(\mathbf{x}_j) + \texttt{stochastic\_noise}$$

Stochastic Noise
Fluctuations that we cannot model

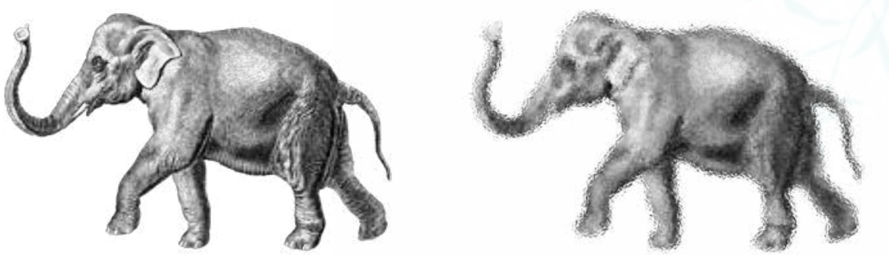


Photo Credits: rawpixel.com

Deterministic Noise: Model Error

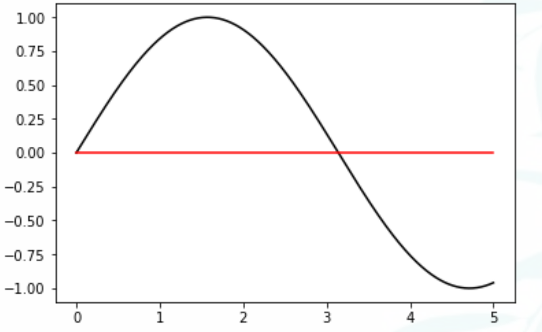
We would like to learn from $\bigcirc$

$$y_j = \bar{h}(\mathbf{x}_j)$$

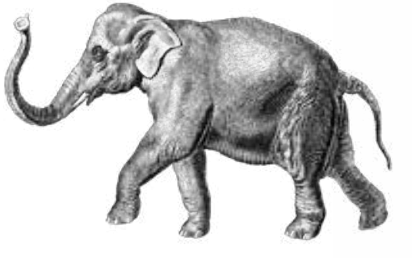
Unfortunately, we only observe $\mathcal{D}$

$$y_j = f(\mathbf{x}_j)$$

$$= \bar{h}(\mathbf{x}_j) + \texttt{deterministic\_noise}$$



Deterministic Noise
The part of f we lack the capacity to model



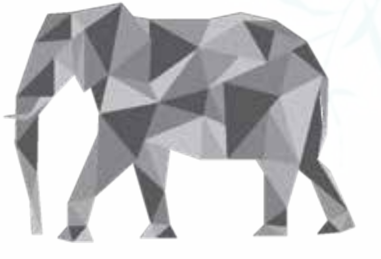
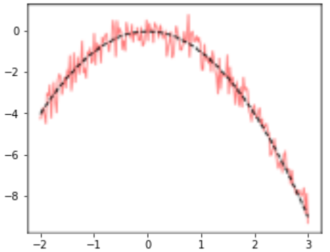
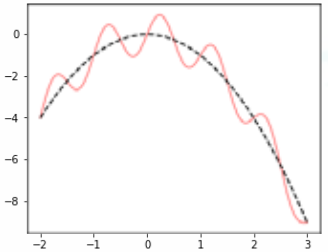


Photo Credits: rawpixel.com

Both sources of noise hurt learning

Stochastic Noise	Deterministic Noise
	
Source: random measurement errors	**Source**: learner's $\mathcal{H}$ cannot model f
Re-measure y: Stochastic noise changes	Re-measure y: Deterministic noise the same
Change $\mathcal{H}$: Stochastic noise still the same	Change $\mathcal{H}$: Deterministic noise changes

We have a single $\mathcal{D}$ and fixed $\mathcal{H}$ so we cannot distinguish between either; with finite m, $\mathcal{H}$ will try to fit the noise.

Noise and Bias-Variance

Recall the decomposition:

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] = \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right]}_{\text{variance}} + \underbrace{\left(\bar{h}(x) - f(x)\right)^2}_{\text{bias}}$$

What if f is a noisy target?

$$y = f(x) + \epsilon(x) \qquad\qquad \mathbb{E}[\epsilon(x)] = 0$$

A noise term

$\mathbb{E}_{\mathcal{D},\epsilon}[(h_{\mathcal{D}}(x) - y)^2] =$

A noise term

$$\mathbb{E}_{\mathcal{D},\epsilon}[(h_{\mathcal{D}}(x) - y)^2] =$$

$$= \mathbb{E}_{\mathcal{D},\epsilon}[(h_{\mathcal{D}}(x) - f(x) - \epsilon(x))^2]$$

Expand observed y

$$= \mathbb{E}_{\mathcal{D},\epsilon}\left[\left(\underbrace{h_{\mathcal{D}}(x) - \bar{h}(x)}_{①} + \underbrace{\bar{h}(x) - f(x)}_{②} - \underbrace{\epsilon(x)}_{③}\right)^2\right]$$

Associate terms

$$= \mathbb{E}_{\mathcal{D},\epsilon}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2 + \left(\bar{h}(x) - f(x)\right)^2 + (\epsilon(x))^2 + \cancel{\text{cross terms}}\right]$$

Additional cross terms disappear as $\mathbb{E}(\epsilon(x)) = 0$

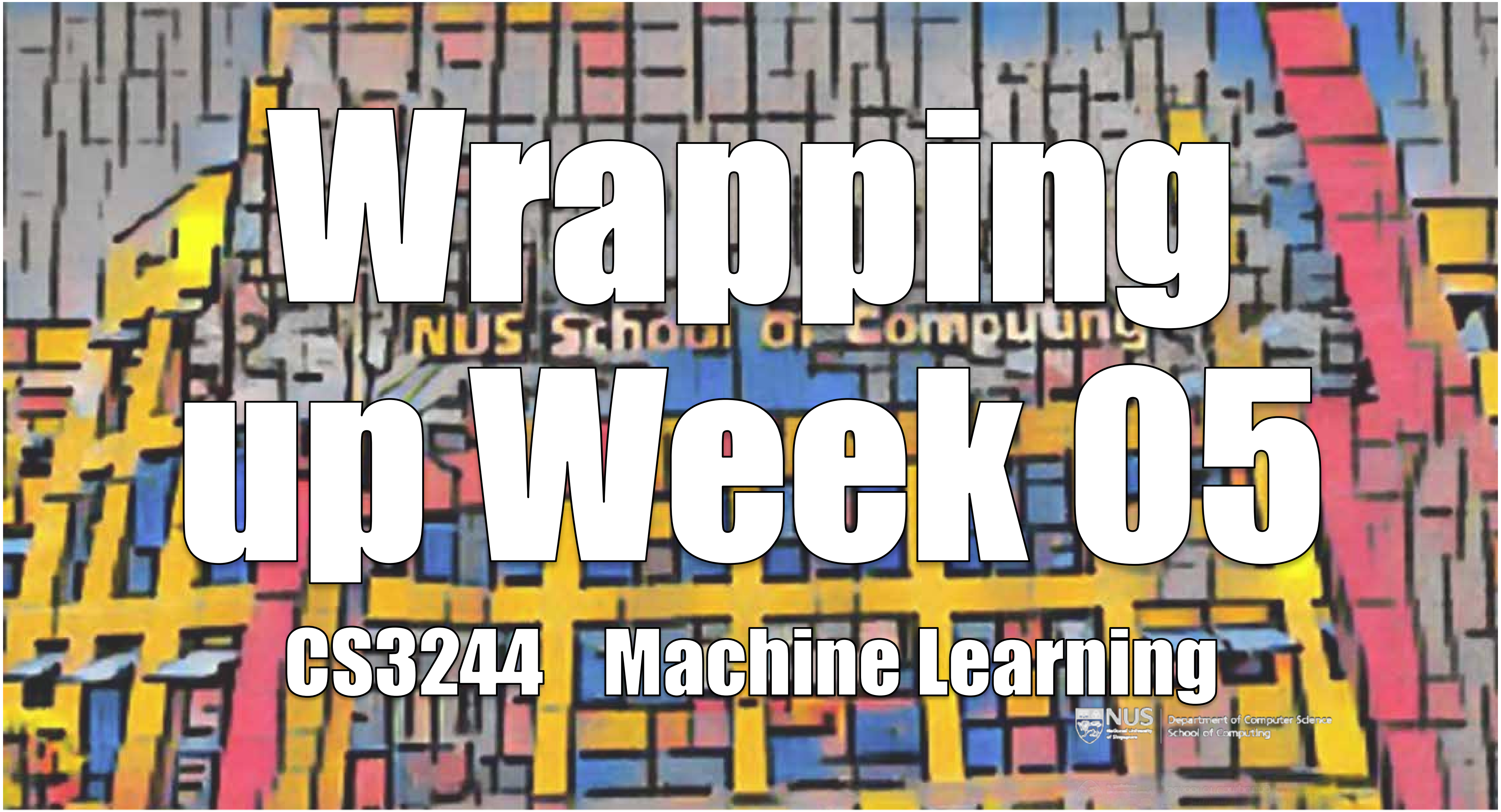
Actually, two noise terms

$$\mathbb{E}_{\mathcal{D}}[(h_{\mathcal{D}}(x) - f(x))^2] =$$

$$\underbrace{\mathbb{E}_{\mathcal{D},x}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right]}_{\text{variance}} + \underbrace{\mathbb{E}_x\left(\bar{h}(x) - f(x)\right)^2}_{\text{bias}} + \underbrace{\mathbb{E}_{\epsilon,x}\left(\epsilon(x)\right)^2}_{\sigma^2}$$

Deterministic Noise

Stochastic Noise


Wrapping up Week 05
NUS School of Computing
CS3244 Machine Learning
NUS
Department of Computer Science
School of Computing

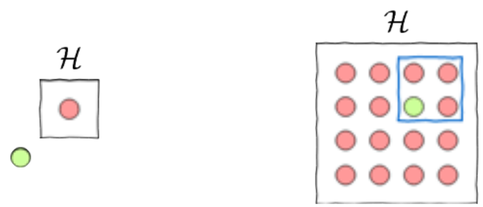
Summary

Bias-Variance Tradeoff

Underline{Bias}

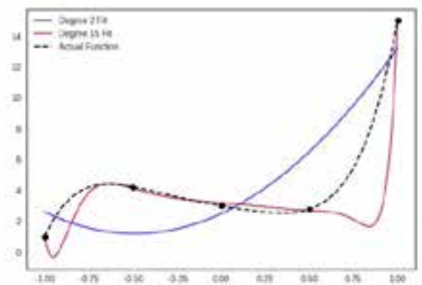
<u>Bias</u>: How well $\mathcal{H}$ can approximate f overall

<u>Variance</u>: How well we can zoom in on a good $h \in \mathcal{H}$



Match the 'model complexity' to the **data resources**, not to the target complexity

Overfitting: Fitting the data more than is warranted



Two causes: stochastic + deterministic noise

Bias $\equiv$ deterministic noise

Noise causes overfitting

Overfitting is the disease

Noise is the cause

Learning is led astray by fitting the noise more than the signal

Two Cures:

1. **Regularization**: Restrain the model
2. **Validation**: Reality check by peeking at (the bottom line)

Outlook for next week

Photo by Louis Reed on Unsplash

Assigned Task (due before next Mon)

Read the post https://www.kaggle.com/alexisbcook/cross-validation (10 mins)

Post a 1–2 sentence answer to the topic in your tutorial group: #tg-xx

How does cross validation relate to variance?

[There's an optional exercise with this course page, using random forest and MAE, you're welcomed to do it if you'd like]