CS7015 (Deep Learning) : Lecture 8

Regularization: Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout

Mitesh M. Khapra

Department of Computer Science and Engineering
Indian Institute of Technology Madras

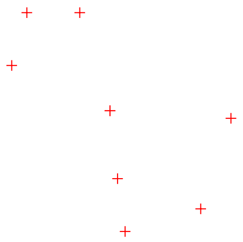
Acknowledgements

- Chapter 7, Deep Learning book
- Ali Ghodsi's Video Lectures on Regularization[a]
- Dropout: A Simple Way to Prevent Neural Networks from Overfitting[b]

[a] Lecture 2.1 and Lecture 2.2
[b] Dropout

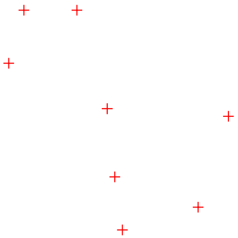
Module 8.1 : Bias and Variance

We will begin with a quick overview of bias, variance and the trade-off between them.



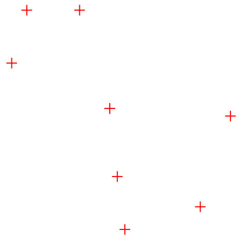
The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points



The points were drawn from a sinusoidal function (the true $f(x)$)

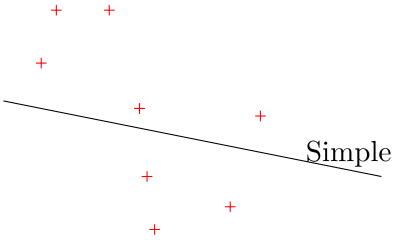
- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :



The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

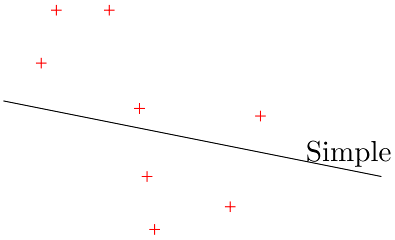
$$\underset{(degree:1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$



The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

$$\underset{(degree:1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$

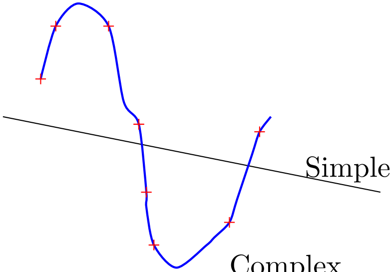


The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

$$\underset{(degree{:}1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$

$$\underset{(degree{:}25)}{Complex} \quad y = \hat{f}(x) = \sum_{i=1}^{25} w_i x^i + w_0$$

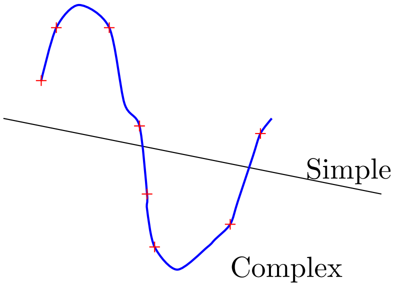


The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

$$\underset{(degree:1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$

$$\underset{(degree:25)}{Complex} \quad y = \hat{f}(x) = \sum_{i=1}^{25} w_i x^i + w_0$$



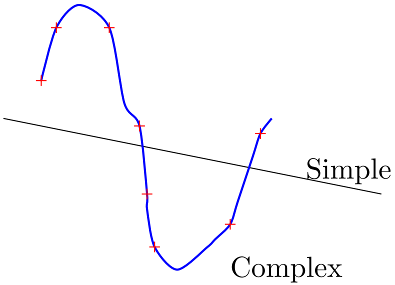
The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

$$\underset{(degree:1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$

$$\underset{(degree:25)}{Complex} \quad y = \hat{f}(x) = \sum_{i=1}^{25} w_i x^i + w_0$$

- Note that in both cases we are making an assumption about how y is related to x. We have no idea about the true relation $f(x)$



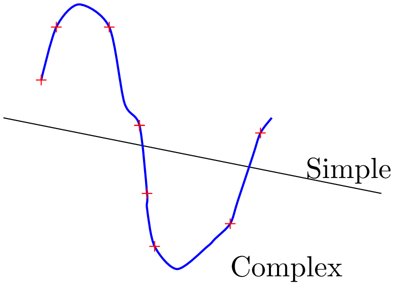
The points were drawn from a sinusoidal function (the true $f(x)$)

- Let us consider the problem of fitting a curve through a given set of points
- We consider two models :

$$\underset{(degree:1)}{Simple} \quad y = \hat{f}(x) = w_1 x + w_0$$

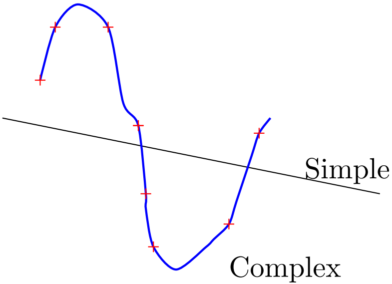
$$\underset{(degree:25)}{Complex} \quad y = \hat{f}(x) = \sum_{i=1}^{25} w_i x^i + w_0$$

- Note that in both cases we are making an assumption about how y is related to x. We have no idea about the true relation $f(x)$
- The training data consists of 100 points



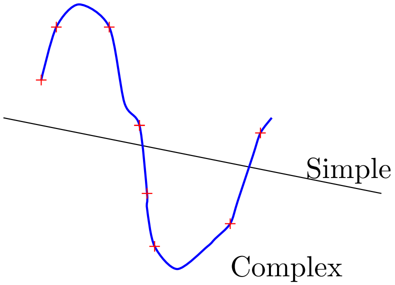
The points were drawn from a sinusoidal function (the true $f(x)$)

- We sample 25 points from the training data and train a simple and a complex model



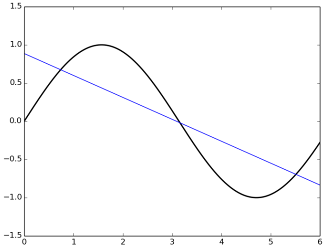
The points were drawn from a sinusoidal function (the true $f(x)$)

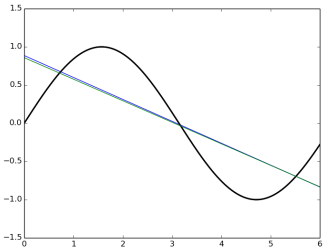
- We sample 25 points from the training data and train a simple and a complex model
- We repeat the process 'k' times to train multiple models (each model sees a different sample of the training data)

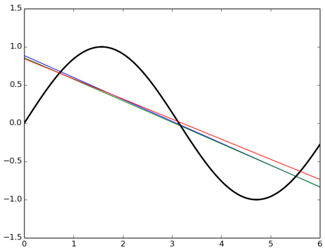


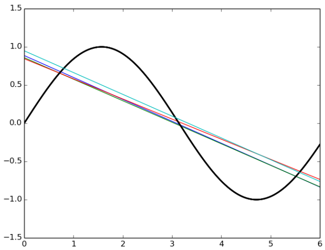
The points were drawn from a sinusoidal function (the true $f(x)$)

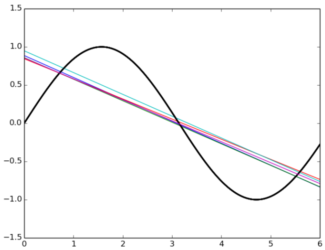
- We sample 25 points from the training data and train a simple and a complex model
- We repeat the process 'k' times to train multiple models (each model sees a different sample of the training data)
- We make a few observations from these plots

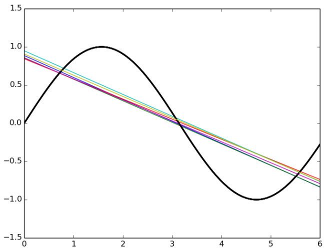

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

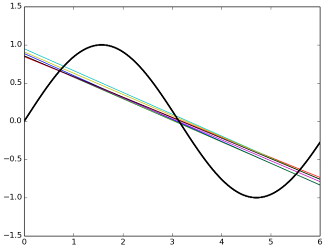

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

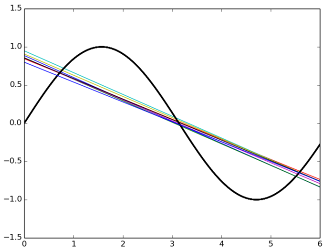

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

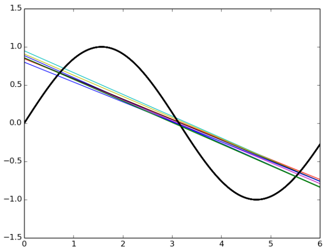

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

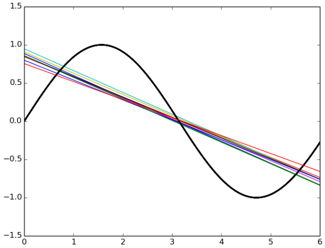

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6



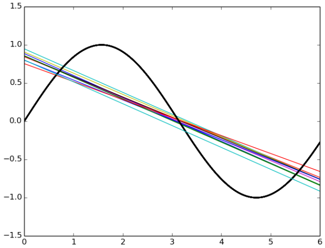
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

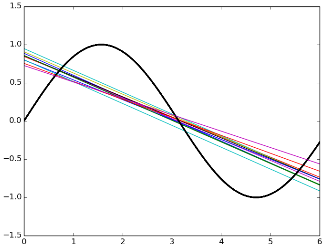

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

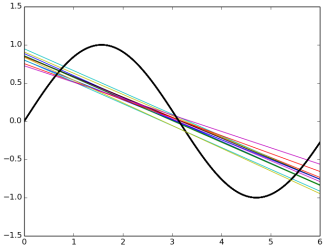

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

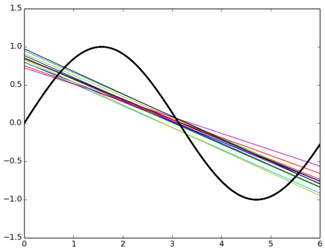


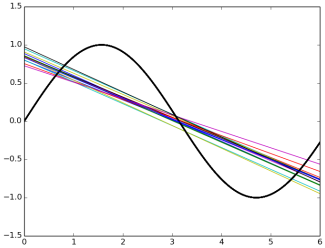
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

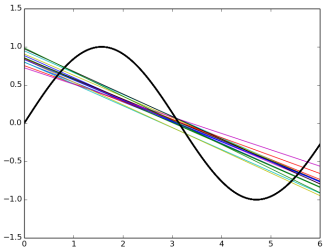

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

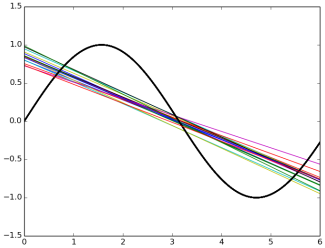

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

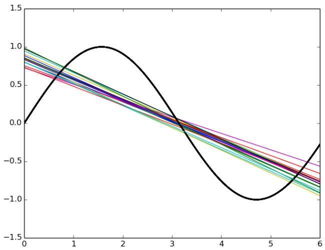

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

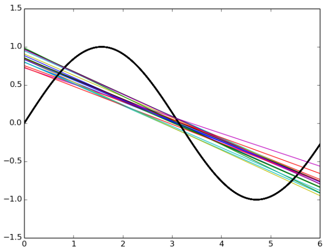

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

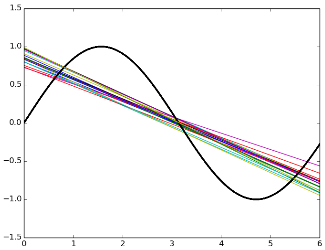

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

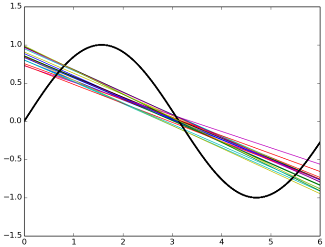

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

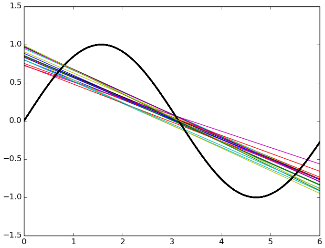

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6

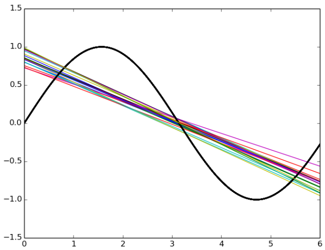

1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5
6



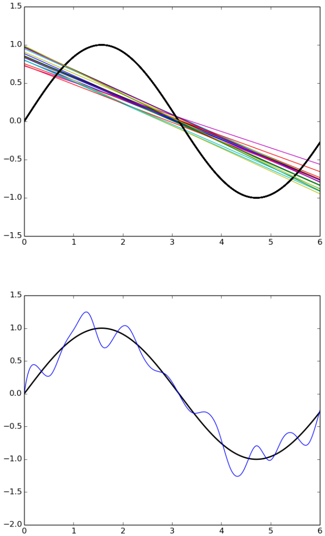
- Simple models trained on different samples of the data do not differ much from each other



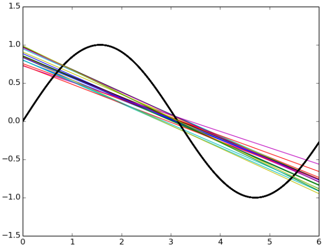
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

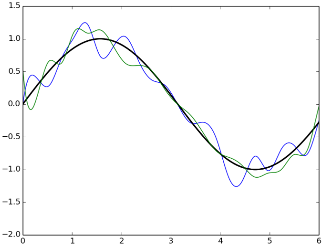


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

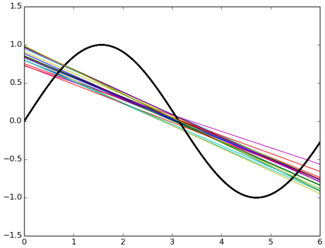


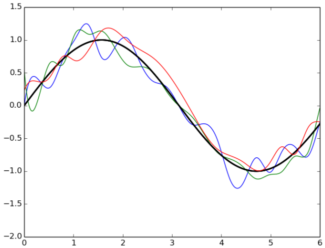
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



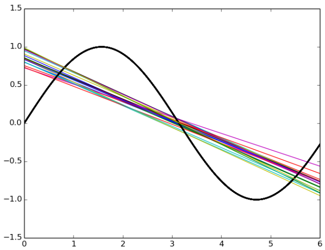


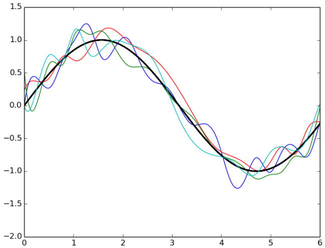
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



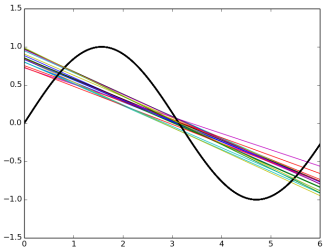


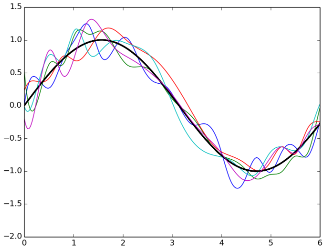
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



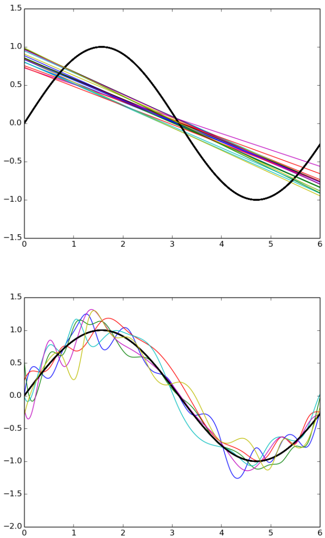


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

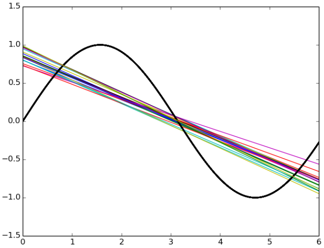


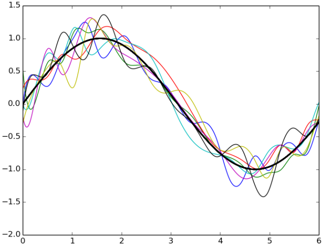


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

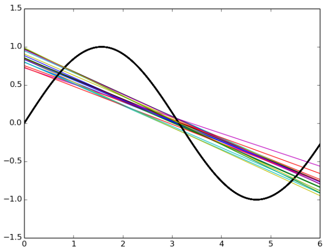


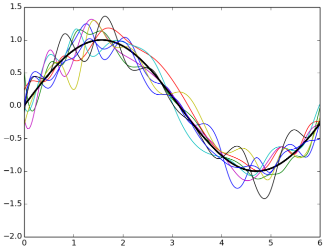
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



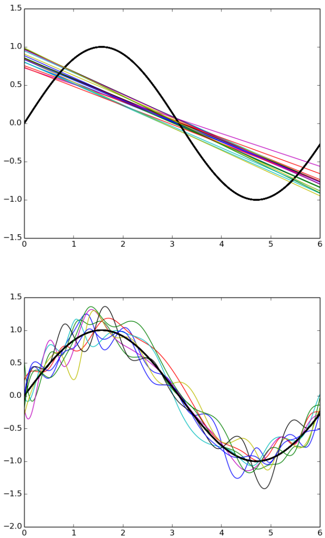


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

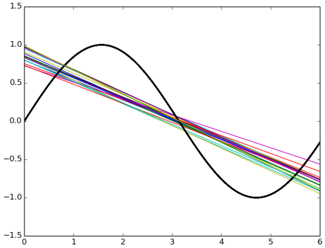


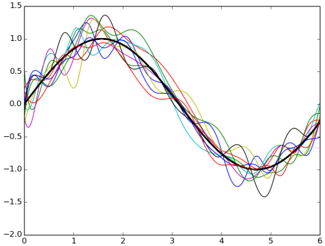


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

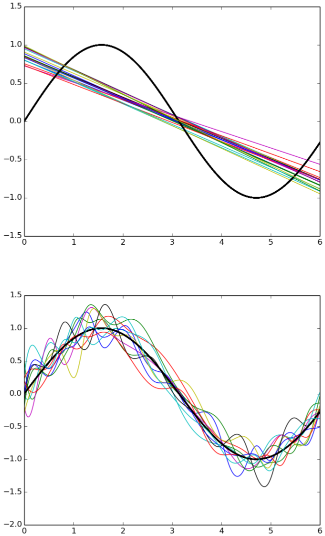


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

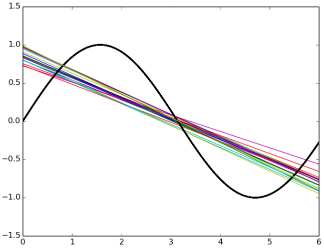


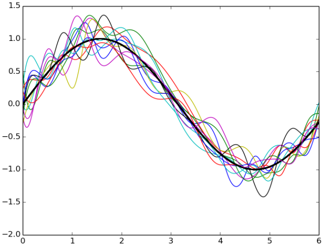


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

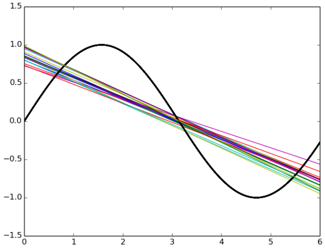


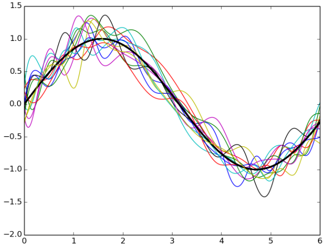
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



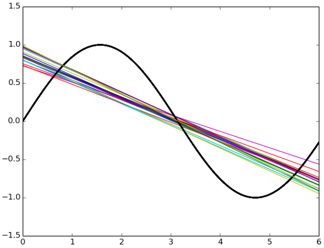


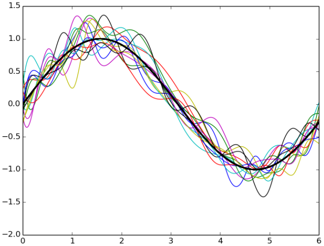
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



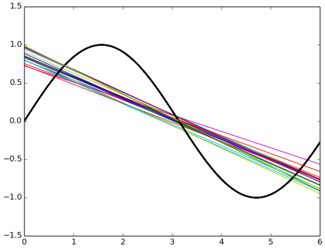


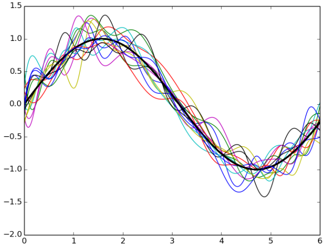
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



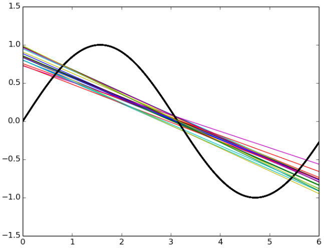


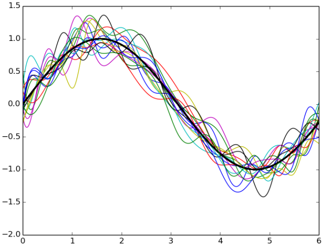
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



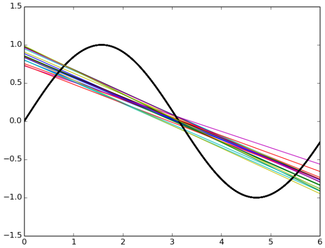


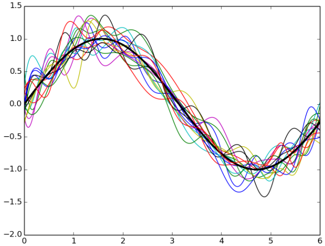
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



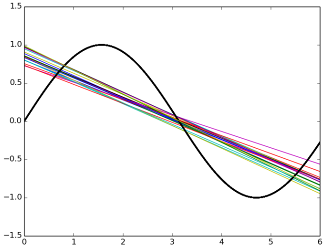


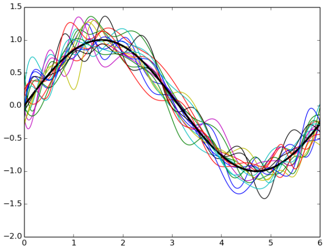
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)



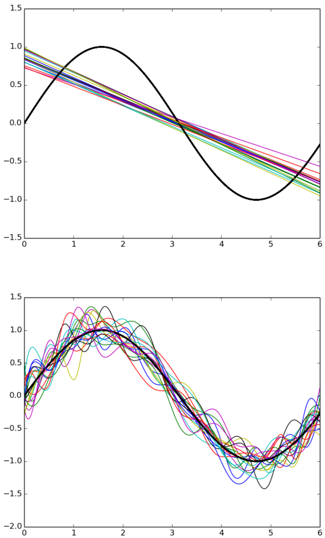


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

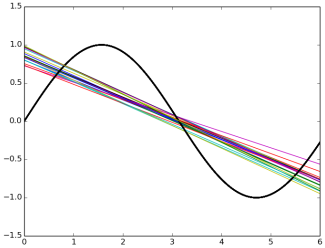


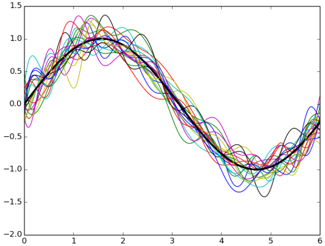


- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

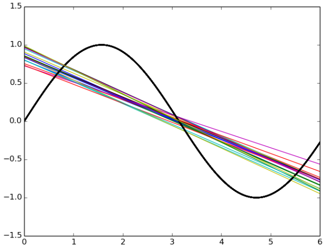


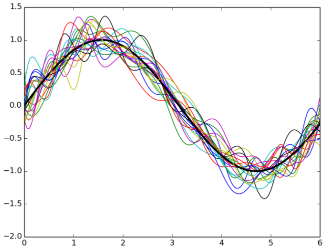
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)





- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)

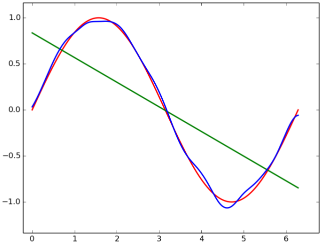




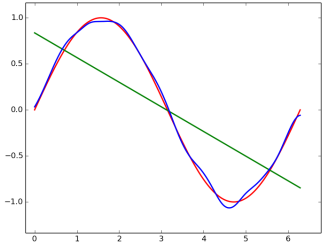
- Simple models trained on different samples of the data do not differ much from each other
- However they are very far from the true sinusoidal curve (under fitting)
- On the other hand, complex models trained on different samples of the data are very different from each other (high variance)

- Let $f(x)$ be the true model (sinusoidal in this case) and $\hat{f}(x)$ be our estimate of the model (simple or complex, in this case) then,

$$\text{Bias } (\hat{f}(x)) = E[\hat{f}(x)] - f(x)$$



Green Line: Average value of $\hat{f}(x)$ for the simple model
Blue Curve: Average value of $\hat{f}(x)$ for the complex model
Red Curve: True model (f(x))

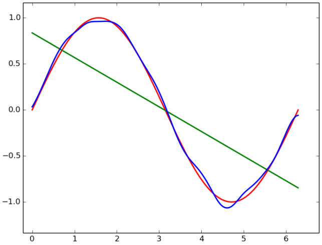


Green Line: Average value of $\hat{f}(x)$ for the simple model
Blue Curve: Average value of $\hat{f}(x)$ for the complex model
Red Curve: True model (f(x))

- Let $f(x)$ be the true model (sinusoidal in this case) and $\hat{f}(x)$ be our estimate of the model (simple or complex, in this case) then,

$$\text{Bias } (\hat{f}(x)) = E[\hat{f}(x)] - f(x)$$

- $E[\hat{f}(x)]$ is the average (or expected) value of the model

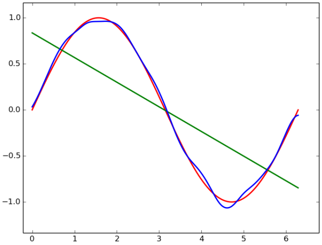


Green Line: Average value of $\hat{f}(x)$ for the simple model
Blue Curve: Average value of $\hat{f}(x)$ for the complex model
Red Curve: True model (f(x))

- Let $f(x)$ be the true model (sinusoidal in this case) and $\hat{f}(x)$ be our estimate of the model (simple or complex, in this case) then,

$$\text{Bias } (\hat{f}(x)) = E[\hat{f}(x)] - f(x)$$

- $E[\hat{f}(x)]$ is the average (or expected) value of the model
- We can see that for the simple model the average value (blue line) is very far from the true value $f(x)$ (sinusoidal function)

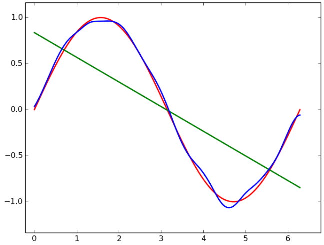


Green Line: Average value of $\hat{f}(x)$ for the simple model
Blue Curve: Average value of $\hat{f}(x)$ for the complex model
Red Curve: True model (f(x))

- Let $f(x)$ be the true model (sinusoidal in this case) and $\hat{f}(x)$ be our estimate of the model (simple or complex, in this case) then,

$$\text{Bias } (\hat{f}(x)) = E[\hat{f}(x)] - f(x)$$

- $E[\hat{f}(x)]$ is the average (or expected) value of the model
- We can see that for the simple model the average value (blue line) is very far from the true value $f(x)$ (sinusoidal function)
- Mathematically, this means that the simple model has a high bias

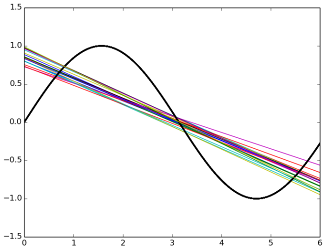


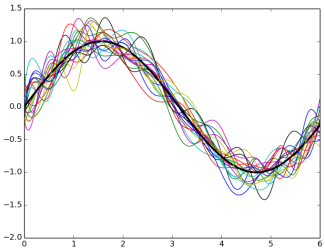
Green Line: Average value of $\hat{f}(x)$ for the simple model
Blue Curve: Average value of $\hat{f}(x)$ for the complex model
Red Curve: True model (f(x))

- Let $f(x)$ be the true model (sinusoidal in this case) and $\hat{f}(x)$ be our estimate of the model (simple or complex, in this case) then,

$$\text{Bias } (\hat{f}(x)) = E[\hat{f}(x)] - f(x)$$

- $E[\hat{f}(x)]$ is the average (or expected) value of the model
- We can see that for the simple model the average value (blue line) is very far from the true value $f(x)$ (sinusoidal function)
- Mathematically, this means that the simple model has a high bias
- On the other hand, the complex model has a low bias

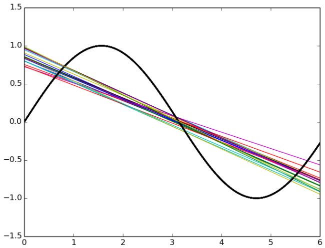


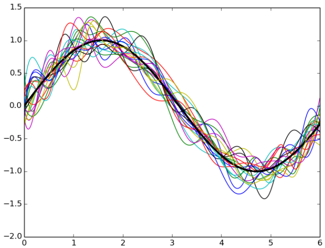


- We now define,

$$\text{Variance } (\hat{f}(x)) = E[(\hat{f}(x) - E[\hat{f}(x)])^2]$$

(Standard definition from statistics)

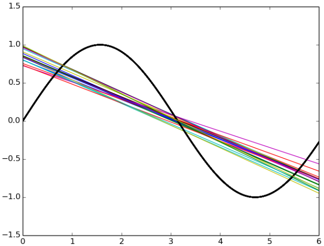


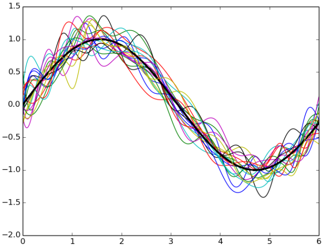


- We now define,

$$\text{Variance } (\hat{f}(x)) = E[(\hat{f}(x) - E[\hat{f}(x)])^2]$$
$$\text{(Standard definition from statistics)}$$

- Roughly speaking it tells us how much the different $\hat{f}(x)$'s (trained on different samples of the data) differ from each other



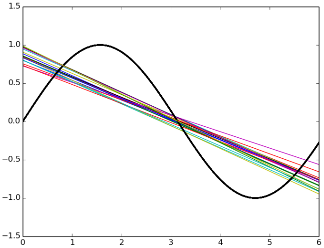


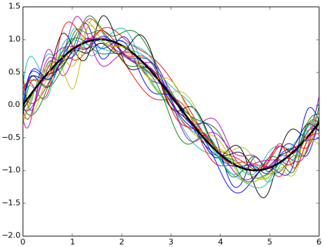
- We now define,

$$\text{Variance } (\hat{f}(x)) = E[(\hat{f}(x) - E[\hat{f}(x)])^2]$$

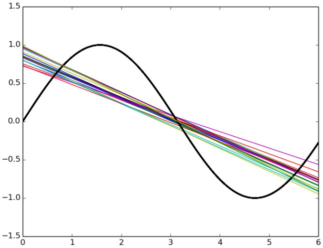
(Standard definition from statistics)

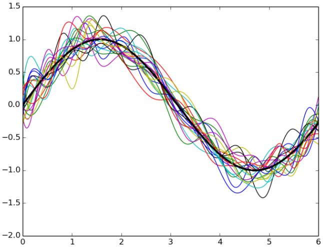
- Roughly speaking it tells us how much the different $\hat{f}(x)$'s (trained on different samples of the data) differ from each other
- It is clear that the simple model has a low variance whereas the complex model has a high variance



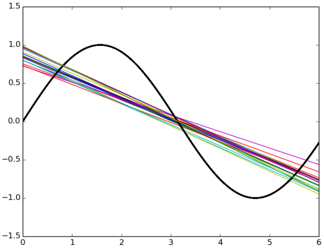


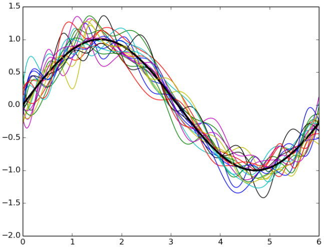
- In summary (informally)



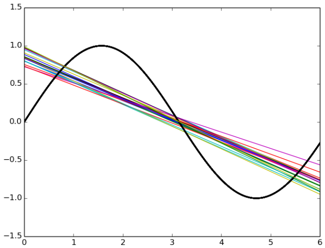


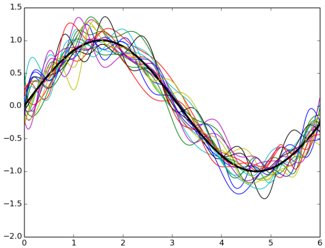
- In summary (informally)
- Simple model: high bias, low variance



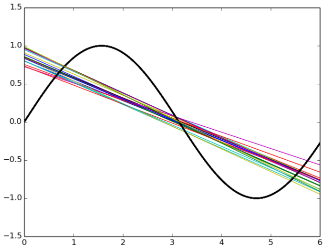


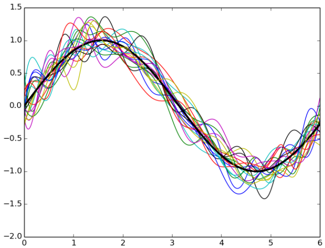
- In summary (informally)
- Simple model: high bias, low variance
- Complex model: low bias, high variance





- In summary (informally)
- Simple model: high bias, low variance
- Complex model: low bias, high variance
- There is always a trade-off between the bias and variance





- In summary (informally)
- Simple model: high bias, low variance
- Complex model: low bias, high variance
- There is always a trade-off between the bias and variance
- Both bias and variance contribute to the mean square error. Let us see how,

Module 8.2 : Train error vs Test error

- Consider a new point (x, y) which was not seen during training

- Consider a new point (x, y) which was not seen during training
- If we use the model $\hat{f}(x)$ to predict the value of y then the mean square error is given by

$$E[(y - \hat{f}(x))^2]$$

(average square error in predicting y for many such unseen points)

- Consider a new point (x, y) which was not seen during training
- If we use the model $\hat{f}(x)$ to predict the value of y then the mean square error is given by

$$E[(y - \hat{f}(x))^2]$$

(average square error in predicting y for many such unseen points)

- We can show that

$$\begin{aligned} E[(y - \hat{f}(x))^2] &= Bias^2 \\ &+ Variance \\ &+ \sigma^2 \text{ (irreducible error)} \end{aligned}$$

- Consider a new point (x, y) which was not seen during training
- If we use the model $\hat{f}(x)$ to predict the value of y then the mean square error is given by

$$E[(y - \hat{f}(x))^2]$$

(average square error in predicting y for many such unseen points)

- We can show that

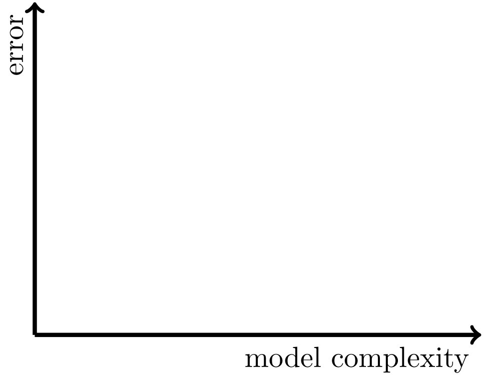
$$\begin{aligned} E[(y - \hat{f}(x))^2] &= Bias^2 \\ &+ Variance \\ &+ \sigma^2 \text{ (irreducible error)} \end{aligned}$$

- See proof here

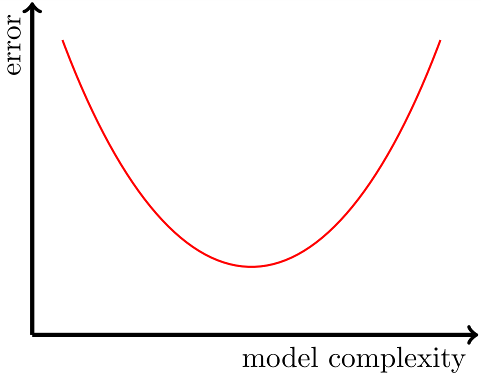
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$

- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training

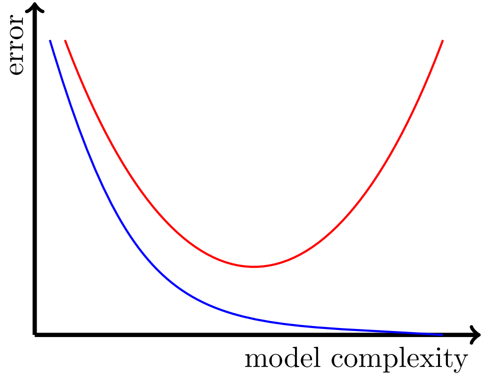
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)



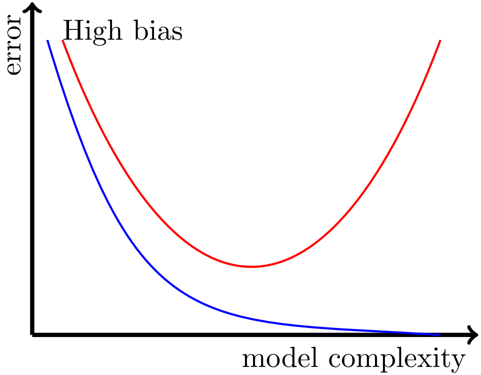
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



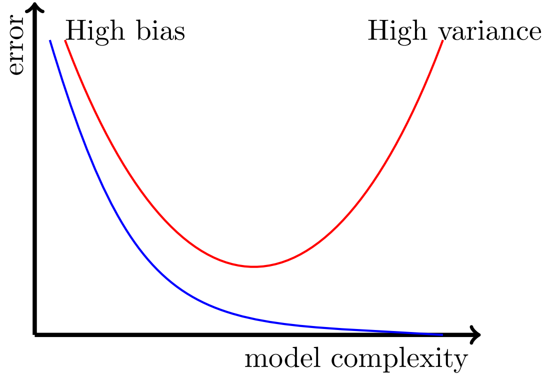
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



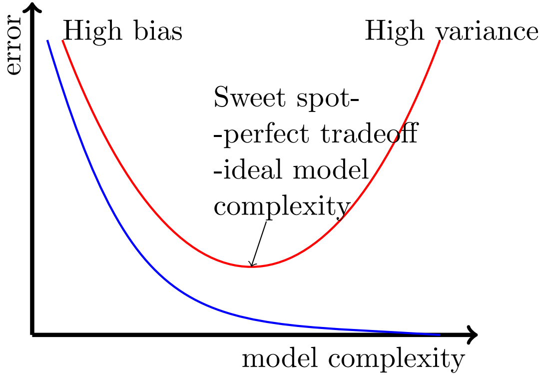
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



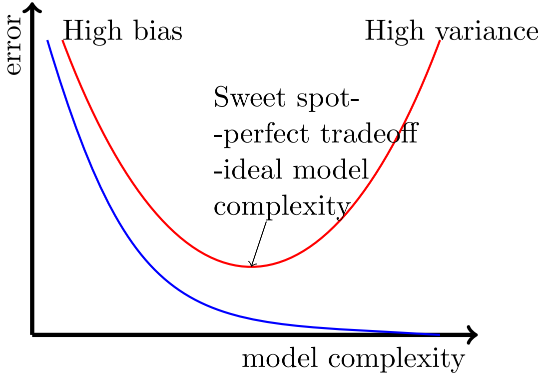
- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^n$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure



$$E[(y - \hat{f}(x))^2] = Bias^2 \\ + Variance \\ + \sigma^2 \text{ (irreducible error)}$$

- The parameters of $\hat{f}(x)$ (all w_i's) are trained using a training set $\{(x_i, y_i)\}_{i=1}^{n}$
- However, at test time we are interested in evaluating the model on a validation (unseen) set which was not used for training
- This gives rise to the following two entities of interest:
  $train_{err}$ (say, mean square error)
  $test_{err}$ (say, mean square error)
- Typically these errors exhibit the trend shown in the adjacent figure

Intuitions developed so far

- Let there be n training points and m test (validation) points

$$train_{err} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$test_{err} = \frac{1}{m}\sum_{i=n+1}^{n+m}(y_i - \hat{f}(x_i))$$

Intuitions developed so far

- Let there be n training points and m test (validation) points

$$train_{err} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$test_{err} = \frac{1}{m}\sum_{i=n+1}^{n+m}(y_i - \hat{f}(x_i))$$

- As the model complexity increases $train_{err}$ becomes overly optimistic and gives us a wrong picture of how close $\hat{f}$ is to f

Intuitions developed so far

- Let there be n training points and m test (validation) points

$$train_{err} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$test_{err} = \frac{1}{m}\sum_{i=n+1}^{n+m}(y_i - \hat{f}(x_i))$$

- As the model complexity increases $train_{err}$ becomes overly optimistic and gives us a wrong picture of how close $\hat{f}$ is to f
- The validation error gives the real picture of how close $\hat{f}$ is to f

Intuitions developed so far

- Let there be n training points and m test (validation) points

$$train_{err} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

$$test_{err} = \frac{1}{m}\sum_{i=n+1}^{n+m}(y_i - \hat{f}(x_i))$$

- As the model complexity increases $train_{err}$ becomes overly optimistic and gives us a wrong picture of how close $\hat{f}$ is to f
- The validation error gives the real picture of how close $\hat{f}$ is to f
- We will concretize this intuition mathematically now and eventually show how to account for the optimism in the training error

- Let $D=\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

$$y_i = f(x_i) + \varepsilon_i$$

- Let D=$\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $$y_i = f(x_i) + \varepsilon_i$$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation

- Let $D=\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $$y_i = f(x_i) + \varepsilon_i$$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation

- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

- Let D=$\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $$y_i = f(x_i) + \varepsilon_i$$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation

- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and of course we do not know f

- Let $D=\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $y_i = f(x_i) + \varepsilon_i$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation

- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and of course we do not know f

- Further we use $\hat{f}$ to approximate f and estimate the parameters using T $\subset$ D such that

  $y_i = \hat{f}(x_i)$

- Let $D=\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $y_i = f(x_i) + \varepsilon_i$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation

- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and of course we do not know f

- Further we use $\hat{f}$ to approximate f and estimate the parameters using T $\subset$ D such that

  $y_i = \hat{f}(x_i)$

- We are interested in knowing $E[(\hat{f}(x_i) - f(x_i))^2]$

- Let D=$\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $y_i = f(x_i) + \varepsilon_i$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation
- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and of course we do not know f
- Further we use $\hat{f}$ to approximate f and estimate the parameters using T $\subset$ D such that

  $y_i = \hat{f}(x_i)$

- We are interested in knowing $E[(\hat{f}(x_i) - f(x_i))^2]$ but we cannot estimate this directly because we do not know f

- Let $D=\{x_i, y_i\}_{i=1}^{m+n}$, then for any point (x, y) we have,

  $y_i = f(x_i) + \varepsilon_i$

- which means that y_i is related to x_i by some true function f but there is also some noise ε in the relation
- For simplicity, we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and of course we do not know f
- Further we use $\hat{f}$ to approximate f and estimate the parameters using T $\subset$ D such that

  $y_i = \hat{f}(x_i)$

- We are interested in knowing $E[(\hat{f}(x_i) - f(x_i))^2]$ but we cannot estimate this directly because we do not know f
- We will see how to estimate this empirically using the observation y_i & prediction $\hat{y}_i$

$$E[(\hat{y}_i - y_i)^2]$$

$$E[(\hat{y}_i - y_i)^2] = E[(\hat{f}(x_i) - f(x_i) - \varepsilon_i)^2] \quad (y_i = f(x_i) + \varepsilon_i)$$

$$E[(\hat{y}_i - y_i)^2] = E[(\hat{f}(x_i) - f(x_i) - \varepsilon_i)^2] \quad (y_i = f(x_i) + \varepsilon_i)$$

$$= E[(\hat{f}(x_i) - f(x_i))^2 - 2\varepsilon_i(\hat{f}(x_i) - f(x_i)) + \varepsilon_i^2]$$

$$E[(\hat{y}_i - y_i)^2] = E[(\hat{f}(x_i) - f(x_i) - \varepsilon_i)^2] \quad \color{red}{(y_i = f(x_i) + \varepsilon_i)}$$

$$= E[(\hat{f}(x_i) - f(x_i))^2 - 2\varepsilon_i(\hat{f}(x_i) - f(x_i)) + \varepsilon_i^2]$$

$$= E[(\hat{f}(x_i) - f(x_i))^2] - 2E[\varepsilon_i(\hat{f}(x_i) - f(x_i))] + E[\varepsilon_i^2]$$

$$E[(\hat{y}_i - y_i)^2] = E[(\hat{f}(x_i) - f(x_i) - \varepsilon_i)^2] \quad (y_i = f(x_i) + \varepsilon_i)$$

$$= E[(\hat{f}(x_i) - f(x_i))^2 - 2\varepsilon_i(\hat{f}(x_i) - f(x_i)) + \varepsilon_i^2]$$

$$= E[(\hat{f}(x_i) - f(x_i))^2] - 2E[\varepsilon_i(\hat{f}(x_i) - f(x_i))] + E[\varepsilon_i^2]$$

$$\therefore E[(\hat{f}(x_i) - f(x_i))^2] \ = \ E[(\hat{y}_i - y_i)^2] \ - \ E[\varepsilon_i^2] \ + \ 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

We will take a small detour to understand how to empirically estimate an Expectation and then return to our derivation

- Suppose we have observed the goals scored(z) in k matches as $z_1 = 2,\ z_2 = 1,\ z_3 = 0,\ \ldots\ z_k = 2$

- Suppose we have observed the goals scored(z) in k matches as $z_1 = 2,\ z_2 = 1,\ z_3 = 0,\ \ldots\ z_k = 2$
- Now we can empirically estimate $E[z]$ i.e. the expected number of goals scored as

$$E[z] = \frac{1}{k}\sum_{i=1}^{k} z_i$$

- Suppose we have observed the goals scored(z) in k matches as $z_1 = 2,\ z_2 = 1,\ z_3 = 0,\ \ldots\ z_k = 2$
- Now we can empirically estimate $E[z]$ i.e. the expected number of goals scored as

$$E[z] = \frac{1}{k}\sum_{i=1}^{k} z_i$$

- Analogy with our derivation: We have a certain number of observations y_i & predictions $\hat{y}_i$ using which we can estimate

$$E[(\hat{y}_i - y_i)^2] =$$

- Suppose we have observed the goals scored(z) in k matches as $z_1 = 2,\ z_2 = 1,\ z_3 = 0,\ \ldots\ z_k = 2$
- Now we can empirically estimate $E[z]$ i.e. the expected number of goals scored as

$$E[z] = \frac{1}{k}\sum_{i=1}^{k} z_i$$

- Analogy with our derivation: We have a certain number of observations y_i & predictions $\hat{y}_i$ using which we can estimate

$$E[(\hat{y}_i - y_i)^2] = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2$$

... returning back to our derivation

$$E[(\hat{f}(x_i) - f(x_i))^2] \;=\; E[(\hat{y}_i - y_i)^2] \;-\; E[\varepsilon_i^2] \;+\; 2E[\;\varepsilon_i(\hat{f}(x_i) - f(x_i))\;]$$

$$E[(\hat{f}(x_i) - f(x_i))^2] \;=\; E[(\hat{y}_i - y_i)^2] \;-\; E[\varepsilon_i^2] \;+\; 2E[\;\varepsilon_i(\hat{f}(x_i) - f(x_i))\;]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\, error}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} -$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\, error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\, estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\, constant}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i,\hat{f}(x_i)-f(x_i))}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$\because$ covariance(X, Y)

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\because \text{covariance}(X, Y) = E[(X - \mu_X)(Y - \mu_Y)]$$

$$E[(\hat{f}(x_i) - f(x_i))^2] \;=\; E[(\hat{y}_i - y_i)^2] \;-\; E[\varepsilon_i^2] \;+\; 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\begin{aligned} \because \text{covariance}(X, Y) &= E[(X - \mu_X)(Y - \mu_Y)] \\ &= E[(X)(Y - \mu_Y)] (\text{if } \mu_X = E[X] = 0) \end{aligned}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\begin{aligned}
\because \text{covariance}(X, Y) &= E[(X - \mu_X)(Y - \mu_Y)] \\
&= E[(X)(Y - \mu_Y)](\text{if } \mu_X = E[X] = 0) \\
&= E[XY] - E[X\mu_Y]
\end{aligned}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] \;=\; E[(\hat{y}_i - y_i)^2] \;-\; E[\varepsilon_i^2] \;+\; 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\begin{aligned} \because \text{covariance}(X, Y) &= E[(X - \mu_X)(Y - \mu_Y)] \\ &= E[(X)(Y - \mu_Y)](\text{if } \mu_X = E[X] = 0) \\ &= E[XY] - E[X\mu_Y] = E[XY] - \mu_X E[X] \end{aligned}$$

$$E[(\hat{f}(x_i) - f(x_i))^2] = E[(\hat{y}_i - y_i)^2] - E[\varepsilon_i^2] + 2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]$$

- We can empirically evaluate R.H.S using training observations or test observations

Case 1: Using test observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\,error} = \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\,estimation\,of\,error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\,constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\begin{aligned} \because \text{covariance}(X, Y) &= E[(X - \mu_X)(Y - \mu_Y)] \\ &= E[(X)(Y - \mu_Y)](\text{if } \mu_X = E[X] = 0) \\ &= E[XY] - E[X\mu_Y] = E[XY] - \mu_X E[X] = E[XY] \end{aligned}$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))]$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] = E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))]$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] = E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] = 0 \cdot E[\hat{f}(x_i) - f(x_i))]$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] = E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] = 0 \cdot E[\hat{f}(x_i) - f(x_i))] = 0$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] = E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] = 0 \cdot E[\hat{f}(x_i) - f(x_i))] = 0$$

$$\therefore \text{true error} = \text{empirical test error} + \text{small constant}$$

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{m}\sum_{i=n+1}^{n+m}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

- None of the test observations participated in the estimation of $\hat{f}(x)$[the parameters of $\hat{f}$(x) were estimated only using training data]

$$\therefore \varepsilon \perp (\hat{f}(x_i) - f(x_i))$$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] = E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] = 0 \cdot E[\hat{f}(x_i) - f(x_i))] = 0$$

$$\therefore \text{true error} = \text{empirical test error} + \text{small constant}$$

- Hence, we should always use a validation set(independent of the training set) to estimate the error

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

Now, $\varepsilon \not\perp \hat{f}(\mathrm{x})$ because ε was used for estimating the parameters of $\hat{f}(x)$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))]$$

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

Now, $\varepsilon \not\perp \hat{f}(\mathrm{x})$ because ε was used for estimating the parameters of $\hat{f}(x)$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))]$$

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

Now, $\varepsilon \not\perp \hat{f}(\mathrm{x})$ because ε was used for estimating the parameters of $\hat{f}(x)$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] \neq E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))]$$

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\ covariance\ (\varepsilon_i, \hat{f}(x_i) - f(x_i))}$$

Now, $\varepsilon \not\perp \hat{f}(\mathrm{x})$ because ε was used for estimating the parameters of $\hat{f}(x)$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] \neq E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] \neq\ 0$$

Hence, the empirical train error is smaller than the true error and does not give a true picture of the error

Case 2: Using training observations

$$\underbrace{E[(\hat{f}(x_i) - f(x_i))^2]}_{true\ error} = \underbrace{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}_{empirical\ estimation\ of\ error} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2}_{small\ constant} + \underbrace{2E[\ \varepsilon_i(\hat{f}(x_i) - f(x_i))\ ]}_{=\,covariance\,(\varepsilon_i,\hat{f}(x_i)-f(x_i))}$$

Now, $\varepsilon \not\perp \hat{f}(\mathrm{x})$ because ε was used for estimating the parameters of $\hat{f}(x)$

$$\therefore E[\varepsilon_i \cdot (\hat{f}(x_i) - f(x_i))] \neq E[\varepsilon_i] \cdot E[\hat{f}(x_i) - f(x_i))] \neq\ 0$$

Hence, the empirical train error is smaller than the true error and does not give a true picture of the error

But how is this related to model complexity? Let us see

Module 8.3 : True error and Model complexity

Using Stein's Lemma (and some trickery) we can show that

$$\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i(\hat{f}(x_i) - f(x_i)) = \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$$

Using Stein's Lemma (and some trickery) we can show that

$$\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i(\hat{f}(x_i) - f(x_i)) = \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$$

- When will $\frac{\partial \hat{f}(x_i)}{\partial y_i}$ be high? When a small change in the observation causes a large change in the estimation($\hat{f}$)

Using Stein's Lemma (and some trickery) we can show that

$$\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i(\hat{f}(x_i) - f(x_i)) = \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$$

- When will $\frac{\partial \hat{f}(x_i)}{\partial y_i}$ be high? When a small change in the observation causes a large change in the estimation($\hat{f}$)
- Can you link this to model complexity?

Using Stein's Lemma (and some trickery) we can show that

$$\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i(\hat{f}(x_i) - f(x_i)) = \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$$

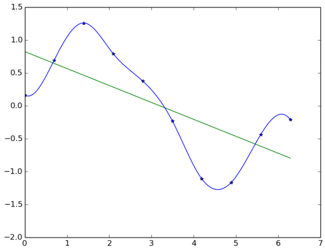
- When will $\frac{\partial \hat{f}(x_i)}{\partial y_i}$ be high? When a small change in the observation causes a large change in the estimation($\hat{f}$)
- Can you link this to model complexity?
- Yes, indeed a complex model will be more sensitive to changes in observations whereas a simple model will be less sensitive to changes in observations

Using Stein's Lemma (and some trickery) we can show that

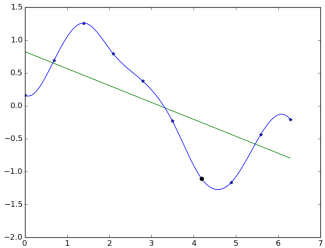
$$\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i(\hat{f}(x_i) - f(x_i)) = \frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$$

- When will $\frac{\partial \hat{f}(x_i)}{\partial y_i}$ be high? When a small change in the observation causes a large change in the estimation($\hat{f}$)
- Can you link this to model complexity?
- Yes, indeed a complex model will be more sensitive to changes in observations whereas a simple model will be less sensitive to changes in observations
- Hence, we can say that
  true error = empirical train error + small constant + Ω(model complexity)

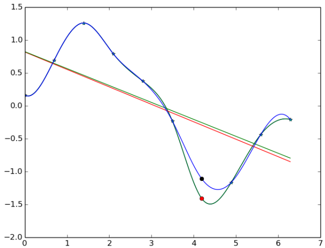
- Let us verify that indeed a complex model is more sensitive to minor changes in the data



- Let us verify that indeed a complex model is more sensitive to minor changes in the data
- We have fitted a simple and complex model for some given data



- Let us verify that indeed a complex model is more sensitive to minor changes in the data
- We have fitted a simple and complex model for some given data
- We now change one of these data points



- Let us verify that indeed a complex model is more sensitive to minor changes in the data
- We have fitted a simple and complex model for some given data
- We now change one of these data points
- The simple model does not change much as compared to the complex model

- Hence while training, instead of minimizing the training error $\mathscr{L}_{train}(\theta)$ we should minimize

$$\min_{w.r.t\ \theta} \mathscr{L}_{train}(\theta) + \Omega(\theta) = \mathscr{L}(\theta)$$

- Hence while training, instead of minimizing the training error $\mathscr{L}_{train}(\theta)$ we should minimize

$$\min_{w.r.t\ \theta} \mathscr{L}_{train}(\theta) + \Omega(\theta) = \mathscr{L}(\theta)$$

- Where $\Omega(\theta)$ would be high for complex models and small for simple models

- Hence while training, instead of minimizing the training error $\mathscr{L}_{train}(\theta)$ we should minimize

$$\min_{w.r.t\ \theta} \mathscr{L}_{train}(\theta) + \Omega(\theta) = \mathscr{L}(\theta)$$

- Where $\Omega(\theta)$ would be high for complex models and small for simple models

- $\Omega(\theta)$ acts as an approximate for $\frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$

- Hence while training, instead of minimizing the training error $\mathscr{L}_{train}(\theta)$ we should minimize

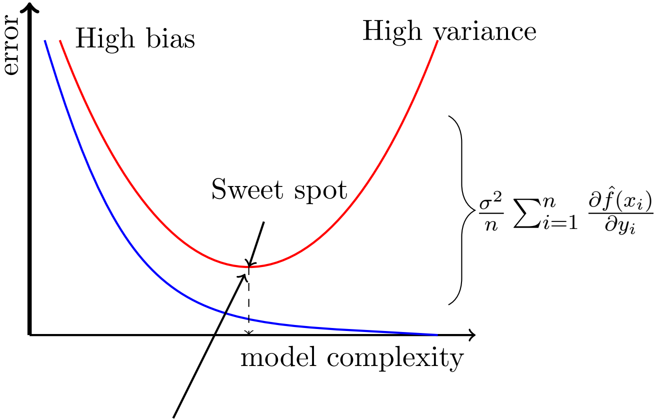
$$\min_{w.r.t\ \theta} \mathscr{L}_{train}(\theta) + \Omega(\theta) = \mathscr{L}(\theta)$$

- Where $\Omega(\theta)$ would be high for complex models and small for simple models
- $\Omega(\theta)$ acts as an approximate for $\frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$
- This is the basis for all regularization methods

- Hence while training, instead of minimizing the training error $\mathscr{L}_{train}(\theta)$ we should minimize

$$\min_{w.r.t\ \theta} \mathscr{L}_{train}(\theta) + \Omega(\theta) = \mathscr{L}(\theta)$$

- Where $\Omega(\theta)$ would be high for complex models and small for simple models
- $\Omega(\theta)$ acts as an approximate for $\frac{\sigma^2}{n}\sum_{i=1}^{n}\frac{\partial \hat{f}(x_i)}{\partial y_i}$
- This is the basis for all regularization methods
- We can show that L_1 regularization, L_2 regularization, early stopping and injecting noise in input are all instances of this form of regularization.



$\Omega(\theta)$ should ensure that model has reasonable complexity

- Why do we care about this bias variance tradeoff and model complexity?

- Why do we care about this bias variance tradeoff and model complexity?

- Deep Neural networks are highly complex models.

- Why do we care about this bias variance tradeoff and model complexity?

- Deep Neural networks are highly complex models.
- Many parameters, many non-linearities.

- Why do we care about this bias variance tradeoff and model complexity?

- Deep Neural networks are highly complex models.
- Many parameters, many non-linearities.
- It is easy for them to overfit and drive training error to 0.

- Why do we care about this bias variance tradeoff and model complexity?

- Deep Neural networks are highly complex models.
- Many parameters, many non-linearities.
- It is easy for them to overfit and drive training error to 0.
- Hence we need some form of regularization.

Different forms of regularization

- L_2 regularization

Different forms of regularization

- L_2 regularization
- Dataset augmentation

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying
- Adding Noise to the inputs

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying
- Adding Noise to the inputs
- Adding Noise to the outputs

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying
- Adding Noise to the inputs
- Adding Noise to the outputs
- Early stopping

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying
- Adding Noise to the inputs
- Adding Noise to the outputs
- Early stopping
- Ensemble methods

Different forms of regularization

- L_2 regularization
- Dataset augmentation
- Parameter Sharing and tying
- Adding Noise to the inputs
- Adding Noise to the outputs
- Early stopping
- Ensemble methods
- Dropout