Review of Lecture 10

• Multilayer perceptrons
  Logical combinations of perceptrons

• Neural networks

\[ x_j^{(l)} = \theta \left( \sum_{i=0}^{d^{(l-1)}} w_{ij}^{(l)} x_i^{(l-1)} \right) \]

where \( \theta(s) = \tanh(s) \)

• Backpropagation

\[ \Delta w_{ij}^{(l)} = -\eta x_i^{(l-1)} \delta_j^{(l)} \]

where

\[ \delta_i^{(l-1)} = (1 - (x_i^{(l-1)})^2) \sum_{j=1}^{d^{(l)}} w_{ij}^{(l)} \delta_j^{(l)} \]
Learning From Data

Yaser S. Abu-Mostafa
California Institute of Technology

Lecture 11: Overfitting
Outline

• What is overfitting?

• The role of noise

• Deterministic noise

• Dealing with overfitting
Illustration of overfitting

Simple target function

5 data points - noisy

4th-order polynomial fit

\[ E_{\text{in}} = 0, \ E_{\text{out}} \text{ is huge} \]
Overfitting versus bad generalization

Neural network fitting noisy data

Overfitting: \( E_{\text{in}} \downarrow \quad E_{\text{out}} \uparrow \)
The culprit

**Overfitting:** “fitting the data more than is warranted”

**Culprit:** fitting the noise - **harmful**
Case study

10th-order target + noise

50th-order target
Two fits for each target

Noisy low-order target

<table>
<thead>
<tr>
<th></th>
<th>2nd Order</th>
<th>10th Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{in}$</td>
<td>0.050</td>
<td>0.034</td>
</tr>
<tr>
<td>$E_{out}$</td>
<td>0.127</td>
<td>9.00</td>
</tr>
</tbody>
</table>

Noiseless high-order target

<table>
<thead>
<tr>
<th></th>
<th>2nd Order</th>
<th>10th Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{in}$</td>
<td>0.029</td>
<td>$10^{-5}$</td>
</tr>
<tr>
<td>$E_{out}$</td>
<td>0.120</td>
<td>7680</td>
</tr>
</tbody>
</table>
An irony of two learners

Two learners $O$ and $R$

They know the target is 10th order

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

Learning a 10th-order target
We have seen this case

Remember learning curves?

$$\mathcal{H}_2$$

$$\mathcal{H}_{10}$$

Expected Error

Number of Data Points, $N$

$$E_{\text{out}}$$

$$E_{\text{in}}$$

$$E_{\text{out}}$$

$$E_{\text{in}}$$

© Creator: Yaser Abu-Mostafa - LFD Lecture 11
Even without noise

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

They know there is no noise

Is there really no noise?

Learning a 50th-order target
A detailed experiment

Impact of noise level and target complexity

\[ y = f(x) + \varepsilon(x) = \sum_{q=0}^{Q_f} \alpha_q x^q + \varepsilon(x) \]

- noise level: \( \sigma^2 \)
- target complexity: \( Q_f \)
- data set size: \( N \)
The overfit measure

We fit the data set $(x_1, y_1), \cdots, (x_N, y_N)$ using our two models:

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

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

Compare out-of-sample errors of $g_2 \in \mathcal{H}_2$ and $g_{10} \in \mathcal{H}_{10}$

overfit measure: $E_{\text{out}}(g_{10}) - E_{\text{out}}(g_2)$
The results

Impact of $\sigma^2$  
Impact of $Q_f$
Impact of “noise”

Stochastic noise

Deterministic noise

number of data points $\uparrow$ Overfitting $\downarrow$

stochastic noise $\uparrow$ Overfitting $\uparrow$

deterministic noise $\uparrow$ Overfitting $\uparrow$
Outline

- What is overfitting?
- The role of noise
- Deterministic noise
- Dealing with overfitting
Definition of deterministic noise

The part of $f$ that $\mathcal{H}$ cannot capture: $f(x) - h^*(x)$

Why “noise”? 

Main differences with stochastic noise:

1. depends on $\mathcal{H}$
2. fixed for a given $x$
Impact on overfitting

Deterministic noise and $Q_f$

Finite $N$: $\mathcal{H}$ tries to fit the noise

how much overfit
Noise and bias-variance

Recall the decomposition:

\[
\mathbb{E}_D \left[ (g^{(D)}(x) - f(x))^2 \right] = \mathbb{E}_D \left[ (g^{(D)}(x) - \bar{g}(x))^2 \right] + \underbrace{\mathbb{E}_D \left[ (\bar{g}(x) - f(x))^2 \right]}_{\text{bias}(x)}
\]

What if \( f \) is a noisy target?

\[
y = f(x) + \epsilon(x) \quad \mathbb{E} \left[ \epsilon(x) \right] = 0
\]
A noise term

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

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

\[ = \mathbb{E}_{D, \epsilon} \left[ \left( g^{(D)}(x) - \bar{g}(x) \right)^2 + \left( \bar{g}(x) - f(x) \right)^2 + \left( \epsilon(x) \right)^2 \right] + \text{cross terms} \]
Actually, two noise terms

\[ E_{D,x} \left[ \left( g^{(D)}(x) - \bar{g}(x) \right)^2 \right] + E_x \left[ \left( \bar{g}(x) - f(x) \right)^2 \right] + E_{\epsilon,x} \left[ \left( \epsilon(x) \right)^2 \right] \]

\[ \text{var} \quad \text{bias} \quad \sigma^2 \]

deterministic noise  stochastic noise
Outline

- What is overfitting?
- The role of noise
- Deterministic noise
- Dealing with overfitting
Two cures

Regularization: Putting the brakes

Validation: Checking the bottom line
Putting the brakes

- Data
- Target
- Fit

free fit  
restrained fit