CS 4824/ECE 4424: Deep Neural Networks I

Acknowledgement:
Many of these slides are derived from Tom Mitchell, Pascal Poupart, Pieter Abbeel, Eric Eaton, Carlos Guestrin, William Cohen, and Andrew Moore.
Deep Neural Networks

- **DNN**: neural network with many hidden layers

- **Advantage**: highly expressive

- **Challenges**:
  - How to train a deep neural network?
  - How to avoid overfitting?
Expressiveness

- Neural networks with one hidden layer of sigmoid/tanh units can approximate arbitrarily closely neural networks with several layers of sigmoid/hyperbolic units.

- However, as we increase the number of layers, the number of units needed may decrease exponentially (with the number of layers).
Example – Parity Function

- Odd or even
  \[ \begin{cases} 
  1 & \text{if odd} \\
  -1 & \text{if even} 
  \end{cases} \]

- Possible odd combinations

\[
\begin{array}{cccc}
X_1 & X_2 & X_3 & X_4 \\
\end{array}
\]
Example – Parity Function

- Single layer of hidden nodes

\[ w = 1 \]

\[ w = -1 \]

or

\[ = \begin{cases} 
1 & \text{if odd} \\
-1 & \text{if even}
\end{cases} \]

2^{n-1} odd subsets

\[ n \text{ inputs} \]
Example – Parity Function

- $2n - 2$ layers of hidden nodes

Diagram:

$X_1 \xrightarrow{\text{and}} \xrightarrow{\text{or}} \xrightarrow{\text{and}} \xrightarrow{\text{or}} \xrightarrow{\text{and}} \xrightarrow{\text{or}}$

- 2 odd subsets

$w = 1$

$w = -1$

Output:

$= \begin{cases} 1 & \text{if odd} \\ -1 & \text{if even} \end{cases}$
The power of depth (practice)

- Challenge: how to train deepNNs?
Speech

- **2006** (Hinton and coworkers): first effective algorithm for deep NN
  - layer-wise training of Stacked Restricted Boltzmann Machines (SRBM)s

- **2009**: Breakthrough in acoustic modeling
  - replace Gaussian Mixture Models by SRBMs
  - Improved speech recognition at Google, Microsoft, IBM

- **2013**: recurrent neural nets (LSTM)
  - Microsoft error rate: **5.9%** (Oct 17, 2016) same as human performance

- ...
Image Classification

- ImageNet Large Scale Visual Recognition Challenge

<table>
<thead>
<tr>
<th>Features + SVMs</th>
<th>Deep Convolutional Neural Nets</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEC (2010)</td>
<td>28.2</td>
</tr>
<tr>
<td>XRCE (2011)</td>
<td>25.8</td>
</tr>
<tr>
<td>AlexNet (2012)</td>
<td>16.4</td>
</tr>
<tr>
<td>VGG (2013)</td>
<td>11.7</td>
</tr>
<tr>
<td>GoogleLeNet (2014)</td>
<td>7.3</td>
</tr>
<tr>
<td>GoogleLeNet-v4 (2016)</td>
<td>3.57</td>
</tr>
<tr>
<td>Human</td>
<td>5.1</td>
</tr>
</tbody>
</table>

Note: The diagram illustrates the classification error (%) for different models and highlights the significant improvement in error rates with the introduction of deep convolutional neural networks.
Challenges in Deep Neural Networks

- How to train a deep neural network?
- How to avoid overfitting?