Deep Learning: Lecture 0

Keith L. Downing

The Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
keithd@idi.ntnu.no

November 17, 2020
The Big Choice

1. **Crafted by Human Engineers**: Hard work for knowledge engineers, but the AI’s decision making is normally straightforward to understand.

2. **Formed by the AI System**: Easy to setup (e.g. via SciKit Learn) but VERY difficult to interpret results, particularly with non-symbolic systems (e.g. neural nets).
Outline

1. A Brief History of Neural Networks
2. Linear Separability
3. Brief Overview of Backpropagation
4. A Few Example Applications
The Early History of Neural Networks

- McCulloch + Pitts (1943) - neuron model similar to logic gates: no weights and no learning, but special excitatory and inhibitory connections.
- Rosenblatt (1958) - The **perceptron**, a 3-layered network. Today, we call his output layer a perceptron, since connections between other two layers were not adaptive.
- Widrow + Hoff (1960) - **Adalines + Delta Rule** for training them, where error signal is based on the weighted sum of inputs, not the output of an activation function.
- Minsky + Papert (1969) - Proved that non-linearly-separable functions (e.g. XOR) could not be represented by a two-layered neural network (regardless of the type of neuron). Since a) Most hard data sets are not linearly separable, and b) Delta rule fails for nets with more than 2 layers → Nets for hard data sets cannot be trained!
- **Near death of neural net research** (1970-1985)
The Ressurection of Neural Networks

- **Hibernation (1995-2005)** - Trappings of local minima and failure of deep nets (due to attenuated backpropagation signals, i.e. gradients) became glaring weaknesses that prevented scaling up.
- **The Deep Learning Revolution (2006-present)** - Unsupervised pre-training (later found unnecessary) + many small (but significant) changes/extensions to backpropagation + major hardware improvements + BIG DATA → Learning in nets with 100+ layers !!
The Perceptron

The diagram illustrates a perceptron model. It consists of input neurons, a hidden layer, and an output neuron. The input neurons are labeled as $x_{1d}$, $x_{2d}$, ..., $x_{nd}$, connected to the hidden layer through weights $w_{i1}$, $w_{i2}$, ..., $w_{in}$. The hidden layer computes the sum of the weighted inputs, represented as $\text{sum}_{id}$. The output neuron is labeled as $f_T$, and the output is $o_{id}$. The error signal is $E_{id}$.
The Delta Rule

\[ \Delta w = \eta \delta X \]

- \( T \) = target output value
- \( \delta \) = error

\[ \delta = T - Y \]
XOR: The (Near) Death of Neural Networks

\[(0.5)x + (0.5)y \geq 1\]
\[y \geq 2 - x\]
\[y \geq -x\]

\[(0.5)x + (0.5)y \geq 0\]

\[(??)x + (??)y \geq ??\]
Linear Separability of Data

\[ w_x x + w_y y \geq t_z \]

\[ \text{net}_z \]

\[ t_z = 1 \]
Adding a Hidden Layer

Not linearly separable $\rightarrow$ Need hidden layer with non-linear act func.
Backpropagation

Training/Test Cases: \{(d_1, r_1) (d_2, r_2) (d_3, r_3)\ldots\}

\[
E = r_3 - r^* \\
dE/dW
\]

N times, with learning
1 time, without learning

Neural Net
NETtalk (Sejnowski + Rosenberg, 1986)

- IBM’s DECTalk: **several man years** of work → Reading Machine.
- NETtalk: **10 hours** of backprop training on 1000-word text (T1000).
- 95% accuracy on T1000; 78% accuracy on novel text.
- Improvement during training sounds like a child learning to read.
- Concept layer is key: 79 different (overlapping) clouds of active neurons gradually form, with each mapping to one of the 79 phonemes.
Endless Applications of Neural Networks

- Stock and commodity price predictions
- Electrical load predictions for the power industry
- Detection of disease from MRI images
- Facial recognition
- Colorization of old black-white movies.
- Natural language interpretation, generation and translation.
- Production of captions for images.
- Control of self-driving vehicles.
- Generation of art, poetry and music
- Automated journalism: given data, write article.
Big Data - *Data is the new oil*

GPUs - greatly speed up the complex calculations of backpropagation.

Convolution nets - based on mammalian visual processing.

LSTMs - slick implementation of recurrence adds critical memory of varying durations.

Dropout - deactivation of random subsets of neurons improves generalization.

Rectified Linear Units (ReLU) - very simple activation function reduces the vanishing-gradient problem→ backprop works in very deep networks.
The Universe of Deep Learning

- **General DL**: Feed Fwd + Backprop over many layers
- **Sequence Models**: Recurrent Nets, LSTM, GRU
- **Image Models**: Convolution Nets (CNNs)
- **Others**: Deep Reinforcement Learning, Adversarial Nets, Unsupervised Nets

* Nuts and Bolts of Applying Deep Learning (Andrew Ng, 2016, YouTube)