LAB COURSE IN DEEP LEARNING

Fall 2016
IMPORTANT ADMINISTRIVIA

• 11-785 – LTI course, **12 credits**, lab course

• http://deeplearning.cs.cmu.edu
What is Learning

• The human perspective:
  • Acquisition of knowledge through experience
    – Underlying causes/influences/patterns
      • for data/phenomena
    – Not the same as memory

• What is deep learning
  – Comprehending the inner structure of observed data
  – Cross-linking new and known concepts to make non-obvious inferences
  – As opposed to surface learning..
    • Learning about the immediately observed data..
What is Learning

• The computational perspective:
  • Acquisition of knowledge through experience
    – Exposure to data

• What is deep learning
  – Learning multi-level representations from data
  – Learning layered models of inputs.
Deep Structures

• In any directed network of computational elements with input source nodes and output sink nodes, “depth” is the length of the longest path from a source to a sink

• Left: Depth = 2.  Right: Depth = 3
Deep Structures

- *Layered* deep structure

![Diagram of a deep neural network showing layers with input, output channels, and pooling sizes.](image)

- “Deep” $\rightarrow$ Depth $> 2$
Deep Structures

• “Learning Deep Architectures for AI”
  – By Yoshua Bengio
Connectionist Machines

• Neural networks are *connectionist* machines
  – As opposed to Von Neumann Machines

• The machine has many processing units
  – The program is the connections between these units
    • Connections may also define memory
A little history: Associationism

- Lightning is generally followed by thunder
  - Ergo – “hey here’s a bolt of lightning, we’re going to hear thunder”
  - Ergo – “We just heard thunder; did someone get hit by lightning”?

- Association!
A little history: **Associationism**

- Collection of ideas stating a basic philosophy:
  - “Pairs of thoughts become associated based on the organism’s past experience”
  - Learning is a mental process that forms associations between temporally related phenomena

- 360 BC: Aristotle
  - "Hence, too, it is that we hunt through the mental train, excogitating from the present or some other, and from similar or contrary or coadjacent. Through this process reminiscence takes place. For the movements are, in these cases, sometimes at the same time, sometimes parts of the same whole, so that the subsequent movement is already more than half accomplished."

- In English: *we memorize and rationalize through association*
Aristotle and Associationism

- Proposed four laws of association from examination of the processes of remembrance and recall:
  - *The law of contiguity*. Things or events that occur close to each other in space or time tend to get linked together.
  - *The law of frequency*. The more often two things or events are linked, the more powerful that association.
  - *The law of similarity*. If two things are similar, the thought of one will tend to trigger the thought of the other.
  - *The law of contrast*. Seeing or recalling something may also trigger the recollection of something opposite.
A little history: Associationism

• More recent associationists (upto 1800s): John Locke, David Hume, David Hartley, James Mill, John Stuart Mill, Alexander Bain, Ivan Pavlov
  – Associationist theory of mental processes: there is only one mental process: the ability to associate ideas
  – Associationist theory of learning: cause and effect, contiguity, resemblance
  – Behaviorism (early 20th century): Behavior is learned from repeated associations of actions with feedback
  – Etc.
Dawn of Connectionism

David Hartley’s *Observations on man* (1749)

- We receive input through vibrations and those are transferred to the brain
- Memories could also be small vibrations (called *vibratiuncles*) in the same regions
- Our brain represents compound or connected ideas by connecting our memories with our current senses
- Current science did not know about neurons
Observation: The Brain

- Mid 1800s: The brain is a mass of interconnected neurons
Enter *Connectionism*

- Alexander Bain, philosopher, mathematician, logician, linguist, professor
- 1873: The information is in the *connections*
Enter: Connectionism

Alexander Bain (*The senses and the intellect* (1855), *The emotions and the will* (1859), *The mind and body* (1873))

- Idea 1: The “nerve currents” from a memory of an event are the same but reduce from the “original shock”

- Idea 2: “for every act of memory, ... there is a specific grouping, or co-ordination of sensations ... by virtue of specific growths in cell junctions”
Bain’s Idea 1: Neural Groupings

- Neurons excite and stimulate each other
- Different combinations of inputs can result in different outputs
Bain’s Idea 1: Neural Groupings

• Different intensities of activation of A lead to the differences in when X and Y are activated
Bain’s Idea 2: Making Memories

• “when two impressions concur, or closely succeed one another, the nerve currents find some bridge or place of continuity, better or worse, according to the abundance of nerve matter available for the transition.”

• Predicts “Hebbian” learning (half a century before Hebb!)
Bain’s Doubts

• “The fundamental cause of the trouble is that in the modern world the stupid are cocksure while the intelligent are full of doubt.”
  – Bertrand Russell

• In 1873, Bain postulated that there must be one million neurons and 5 billion connections relating to 200,000 “acquisitions”

• In 1883, Bain was concerned that he hadn’t taken into account the number of “partially formed associations” and the number of neurons responsible for recall/learning

• By the end of his life (1903), recanted all his ideas!
Connectionism lives on..

• The human brain is a connectionist machine

• Neurons connect to other neurons. The processing/capacity of the brain is a function of these connections

• Connectionist machines emulate this structure
Modelling the brain

- What are the units?
- A neuron:
  - Signals come in through the dendrites into the Soma
  - A signal goes out via the axon to other neurons
    - Only one axon per neuron
  - Factoid that may only interest me: Neurons do not undergo cell division
McCullough and Pitts

• The Doctor and the Hobo..
  – Warren McCulloch: Neurophysician
  – Walter Pitts: Homeless wannabe logician who arrived at his door
The McCulloch and Pitts model

- A mathematical model of a neuron
  - Threshold Logic
Synaptic Model

• *Excitatory synapse*: Transmits weighted input to the neuron

• *Inhibitory synapse*: Any signal from an inhibitory synapse forces output to zero
  – The activity of any inhibitory synapse absolutely prevents excitation of the neuron at that time.
    • Regardless of other inputs
  – This prevents learning from going on indefinitely
Figure 1. Diagrams of McCulloch and Pitts nets. In order to send an output pulse, each neuron must receive two excitatory inputs and no inhibitory inputs. Lines ending in a dot represent excitatory connections; lines ending in a hoop represent inhibitory connections.
Complex Perceput & Inhibition in action

Figure 2. Net explaining the heat illusion. Neuron 3 (heat sensation) fires if and only if it receives two inputs, represented by the lines terminating on its body. This happens when either neuron 1 (heat reception) fires or neuron 2 (cold reception) fires once and then immediately stops firing. When neuron 2 fires twice in a row, the intermediate (unnumbered) neurons excite neuron 4 rather than neuron 3, generating a sensation of cold.
Criticisms

• A misconception spread nets can compute anything that Turing Machines can compute

• They didn’t prove any results themselves

• They claimed that their nets should be able to compute a small class of function

• Also if tape is provided their nets can compute a richer class of functions.

• Additionally they will be equivalent to Turing machines
• So how does the brain *learn*??
Donald Hebb

- Born in 1904
- Initially studied to become a novelist, then became a teacher, later became a farmer and then travelled as a laborer
- Finally became a psychologist inspired by Sigmund Freud
- One of the first psychologists to work on neural basis for describing behavior
Hebb’s Synapse

“When an axon of cell A is near enough to excite cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A’s efficiency as one of the cells firing B is increased.”

Cells that fire together, wire together!
Synaptic knobs

When one cell repeatedly fires another, Axon on first cell develops synaptic knobs or enlarges existing ones and increase contact area with soma of second cell.
Learning

• “Strengthen” connection if any input-output pair co-fire
  – But only if slight delay between input and output
  – To distinguish between causation and co-occurrence
Hebbian Learning

- Mathematically,
  \[ \Delta w_{ij} = \eta \times x_i \times x_j \]

where,
- \( w_{ij} \rightarrow \) the weight of the connection from neuron i to neuron j
- \( x_i, x_j \rightarrow \) the binary excitation levels of neuron i and j
- \( \eta \rightarrow \) learning rate
Hebbian Learning

• Good: Provides a basic mechanism for learning
  – Explains slow and fast learning
  – Provides a mechanism that explains human development

• Deals only with increase in strength of connections, but not decrease in synaptic strength

• Considers only local excitations and correlations. Does not consider the network as a whole while learning

• Learning rule is unstable – Any dominant signal can cause the weights to increase rapidly and is unbounded.
A better model

- Frank Rosenblatt
  - Psychologist, Logician
  - Inventor of the solution to everything, aka the Perceptron (1958)
Rosenblatt’s perceptron

- Original perceptron model
  - Groups of sensors (S) on retina combine onto cells in association area A1
  - Groups of A1 cells combine into Association cells A2
  - Signals from A2 cells combine into response cells R
  - All connections may be excitatory or inhibitory
Rosenblatt’s perceptron

- Even included feedback between A and R cells
  - Ensures mutually exclusive outputs
Simplified mathematical model

- Number of inputs combine linearly
  - Threshold logic: Fire if combined input exceeds threshold

\[ Y = \begin{cases} 
1 & \text{if } \sum_i w_i x_i + b > 0 \\
0 & \text{else}
\end{cases} \]
Simplified mathematical model

• A mathematical model
  – Originally assumed could represent any Boolean circuit
  – Rosenblatt, 1958: “the embryo of an electronic computer that [the Navy] expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence”
Perceptron

- Boolean Gates
- But...
Perceptron

No solution for XOR!
Not universal!

• Minsky and Papert, 1968
A single neuron is not enough

• Individual elements are weak computational elements
  – Marvin Minsky and Seymour Papert, 1969, *Perceptrons: An Introduction to Computational Geometry*

• *Networked* elements are required
Multi-layer Perceptron

- XOR

\[ X \oplus Y \]
Multi-layer perceptrons are universal

- A multi-layer perceptron is a universal Boolean function
  - A universal approximator even in the general case
    - Hornik, Stinchcombe and White, 1989
Revisiting the perceptron: What is a perceptron?

• A correlation filter
  – Fire if correlation between input and weights exceeds a threshold

• Feature detector
  – Detect if a specific pattern occurs in input
Networks of perceptrons

• Individual features may represent *local* patterns in data
• Complex patterns: combinations of local patterns
• Options:
  – A *large* number of perceptrons to learn every possible complex pattern (potentially exponential number of patterns)  -- OR
  – A much smaller heirarchical network that *builds* complex patterns from local patterns (much much much more efficient)
A Learning Problem

• Many layers of inputs
  – Output = $f_1(f_2(f_3(\ldots f_N(X; \theta_N);\ldots); \theta_3); \theta_2);\theta_1)$
  – Learning all parameters $\theta_1, \theta_2, \ldots, \theta_N$ is an optimization nightmare..
  – Simple Hebbian learning and variants do not work directly
A Learning Problem

• Solution: Backpropagation
  – Werbos, 1975
  – Propagate errors and gradients backwards through the network

• Problem:
  – Unreliable for large networks
  – Highly dependent on initialization..

• Cue... a cartoon view of the history of networks..
The story of a great man..
More to it than this

• Is memory really separate from computation
  – Or can computation “remember” ??
    • John Hopfield
  – Is “remembering computation” different from generation?
    • Hinton
How about the eye?

- Neocognitron
  - Hubel and Wiesel 1959 (simple and complex cells in visual cortex)
  - Fukushima (computational model) 1980
- Convolutional neural network
  - Homma, Atlas, Marks, 1988, LeCunn 90s
Interestingly...

- Patterns learned by individual layers of a convolutional network correlate well with activation patterns of individual layers of the visual cortex!
  - Agarwal and Gallant, 2014, Others..
What *can* we learn?

- Learn to play a game from scratch!
  - Without external information
- Learn about the environment
- Learn about language. Learn about representations!
This Course..

• A lab and reading-based course on deep networks

• From the webpage:
  – In this course students will learn about this resurgent subject. The course presents the subject through a series of seminars, which will explore it from its early beginnings, and work themselves to some of the state of the art. The seminars will cover the basics of deep learning and the underlying theory, as well as the breadth of application areas to which it has been applied, as well as the latest issues on learning from very large amounts of data..
How the course is run

• **Standard format:**
  – Each class consists of an introductory lecture (10-20 mins) by instructor/TAs, followed by two paper presentations by students
  – Except for guest lectures

• All students are required to present 2 papers in class.
• We will have 2 presentations per class
• Each presentation will be 30 minutes long
  – 20 minutes presentation, 10 minutes for questions/discussion

• **Everyone is expected to read the papers before the class**
  – Or at least the abstract and intro..
  – Presenters must read all of the papers, obviously 😊
How the course is run

• **Presenters:**
  – Please make slides. We will post these on the website
  – Present the paper thoroughly
  – Backread referenced papers for clarification
  – Attempt to be clear and tutorial
    • This is not a simple recitation of the paper; you have to understand and explain
  – Where required/possible, run simulations etc. for illustration
Lab course

• For 11-785:
  – Several lab exercises
    • The first will be put up next week
    • Lab reports due for each exercise
  
  – One project
    • “Researchy” problem

• http://deeplearning.cs.cmu.edu/labs
Grading

• Presentation
• Reports
• Attendance and participation
• Labs
What we will cover

• Those who cannot remember the past.. (George Santayana)
  – Bain, McCulloch, Rosenblatt, Turing
  – Werbos,
  – Hopfield..

• Types of networks
  – Feedforward
  – Self organizing
  – Convolutive
  – Recurrent structures
  – Generative models
What we will cover

• Applications
  – Image analysis
  – Feature learning
  – Memory
  – Language
  – Reinforcement learning
  – Large data

• Structure discovery
  – Embeddings

• Implementations
  – Distributed mechanisms
  – GPU
Many Labs

• Explorations of feedforward nets
  – Backprop
  – Simple classification and visualization
  – Deep vs shallow

• Real data: Convergence, Initialization and regularization
  – Learning rate,
  – Autoencoding
  – Denoising, dropout
  – Regularization
Many Labs

• Generative models
  – RBM, DBM vs NN, DNN

• Convolutive networks

• Recurrent networks
  – RNN
  – LSTM
  – Uni- and bi-directional

• Tasks: Simulated, image, speech, text
Projects

• Exploratory projects
  – Teams of 2
• May lead to publication
• Push: Please finish by mid November
  – Objective: Submit to ICLR/IJCNN/ICML
    • Deadlines Nov-Feb
• Sign up by end of next week if you can
Projects

• Inverting the network: Exploring null spaces
  – Or how to fool a network
• $L_1$ alternatives to dropout and other hacks
• Spatially coherent networks
  – Or how to mimic spatial localization in the brain
• Pruning networks
  – How to reduce the size of a pre-trained net
Projects

• Deep dictionaries
  – Can Nnets be dictionaries for sparse coding
    • Reversing the network

• Shrinking networks
  – How to zap a net into a tiny processor

• Text to images
  – Create a comic from a story

• Exploration of embeddings

• Static recurrence
  – Recurrent structures for static regression
Administrivia

• Instructor: Me!
  – bhiksha@cs.cmu.edu
  – GHC6705
  – 8-9826
  – Office hours: TBD
  – But you can approach me anytime I’m free

• TAs:
  – Haohan Wang (haohanw@cs.cmu.edu)
    • Office hours: TBD
  – Haoqi Fan (haoqif@andrew.cmu.edu)
    • Office hours: TBD
Webpage

- Hope to have a proper discussion board
- → → → Haohan

- For now, we use blackboard
Readings

• Next Class: September 7th: Haohan and Haoqi will present a tutorial on Theano
  – MLP, Convolutional networks, LSTMs

• September 12th: Backpropagation and its limitations
  – **Backprop will find the local (or global) optimum**: On the problem of local minima in backpropagation, IEEE tran. Pattern Analysis and Machine Intelligence, Vol 14(1), 76-86, 1992, Gori and Tesi
  – **Backprop fails to find the obvious answer**: Backpropagation fails where perceptrons succeed, IEEE Trans on circuits and systems. Vol. 36:5, May 1989, Brady, Raghavan, Slawny
Week of Sep 12th

• September 14th
• Speeding up training
  – Rprop, acceleration, Nestorov’s method
Rough Schedule

• Week 2: Basics
  – Learning, speeding up learning

• Week 3:
  – What does a network represent
  – Alternate uses of networks: Network as memory, networks for structure recovery

• Week 4 & 5:
  – Alternate structures: Convolutive networks, Recurrent formalisms
Further Readings

• 14\textsuperscript{th} Sep: Self Organized Maps, Hopfield Nets

• We will share a Google doc in the next couple of days

• Please sign up

• Remember \textit{everyone} presents

• Next up: Learning rules:
  – Hebbian learning, Widrow Hoff rule, Delta rule, Back propagation, Rprop come next in the series of topics
Reports!

• A report is due from every student on the paper(s) they presented, at the end of the semester
Some History


Some History

- McClullogh and Pitts, 1943 – Threshold Logic
- Turing, 1948 – “Intelligent Machines”
- Farley and Clark 1954 – Hebbian Network
  - Several others followed up
- Rosenblatt 1958 – Perceptron
  - XOR
- Minsky and Papert, 1969 – Limitations
- Werbos, 1975 – back propagation
  - Other algorithms followed