READINGS IN DEEP LEARNING

Fall 2014
IMPORTANT ADMINISTRIVIA

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

• deeplearning.cs.cmu.edu/

• All papers and slides to be linked here
• “Readings in Deep Learning – a student perspective”..
Book

• “Readings in Deep Learning – a student perspective”.
  – Authored by students at CMU
What is Learning

• The human perspective:
  • Acquisition of knowledge through experience
    – *Associations* and *Structural Inferences*
    – Underlying causes/influences/patterns of/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
  – *Associations* and *Structural Inferences*

• 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 with annotations](image)

- "Deep" → 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
Why connectionist machines

- 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
Modelling the brain

- A mathematical model
  - Threshold Logic: McCulloch and Pitts, 1943
  - The perceptron: Rosenblatt, 1958

- Number of inputs combine linearly
  - Threshold logic: Fire if combined input exceeds threshold
  - Perceptron: Output is continuous function of combined input
Learning

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

• “Strengthen” connection if any input-output pair co-fire
  – Hebbian learning rule
    • [Hebb, D.O.](1949). *The Organization of Behavior*
  – ADALINE (adaptive linear element): Widrow-Hoff learning rule
    • Widrow and Hoff, 1960
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
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 hierarchical network that *builds* complex patterns from local patterns (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 / W-H learning and their 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 Nnetworks..
The story of a great man..
This Course..

• A reading-based course on deep networks
• From the unupdated 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.
Administrivia

• **Instructor:** Me!
  – [email: bhiksha@cs.cmu.edu](mailto:bhiksha@cs.cmu.edu)
  – GHC6705
  – 8-9826
  – **Office hours:** Mondays, 3.30-5.00
  – But you can approach me anytime I’m free

• **TA:** TBD
  – **Office hours:** TBD
Webpage

• deeplearning.cs.cmu.edu/

• Will set up a blog for discussion
How the course is run

• The course will be delivered entirely by you!
  – Except for guest lectures

• All students are required to present at least 2 papers in class.
• We will have 2 presentations per class
• Each presentation will be 40 minutes long
  – 30 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 the entire paper, 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

• Six lab exercises
  – The first will be put up next week
  – We may have fewer exercises if time is a constraint

– Lab reports due for each exercise.
Grading

• Presentation
• Reports
• Attendance and participation
• Lab reports
Suggested Readings

• We will cover
  – History and basics of neural networks
  – Deep learning theory and applications
  – Large-scale learning – learning large networks from a large amount of data

• Today: Amos Ng presents Wilkes and Wade’s paper describing Bain’s work
Suggested Readings

• Next 3 weeks: Basics

• September 3rd:
  – Discuss one of the first mathematical models of a neuron
  – The Threshold Logic Model

• September 8th:
  – Hebbian learning rules for neurons
    • “The organization of behavior” by Hebb
  – The Widrow Hoff learning rule
    • Widrow and Lehr
Suggested Readings

• Next 2 weeks:
  • Limitations of Neurons
    • Minsky and Papert (XOR). [Multilayer networks]
  • Delta rule and Back propagation
  • Hopfield networks, content-addressable memory, self-organizing maps
  • Boltzmann machines
Suggested Readings

• Volunteers required for the week of the 8th
• Volunteers required for the next 3 weeks by Monday

• Reading list will be on webpage
  – We will put up pdfs where permitted/possible
  – You may choose an alternate paper that covers the same topics, with our permission
Reports!

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

• This course will become a book
  – “Readings in Deep Learning – a student perspective”
• Make your reports into book chapters!
  – You may collaborate – up to 2 persons per chapter
• Make them thorough
  – Write more than just what was in your assigned paper(s)
  – Make it tutorial

• Students who are not enrolled are also permitted to contribute chapters
Readings..

- We will
  - Select a subset of chapters (based on readability of the chapter, as well as to have balanced coverage)
  - Iterate with you in the spring semester to polish your chapters
    - Please be willing to rewrite it a couple of times if needed
- The final volume will be sent to a publisher
  - We may already have a publisher
Some History

• 430BC .. 1800s: Associationist school
  – Plato, Aristotle ... John Locke, David Hume, David Hartley, James Mill, John Stuart Mill, Alexander Bain and Ivan Pavlov


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
Some History

- Rumelhart, McLellan and Jacobs: “Parallel Distributed Processing” – 1986
- Hopfield
- Boltzmann Machines
  - RBMs
- ??
- ??
Over to Amos

• “Bain on Neural Networks”...
• Alexander Bain (1818-1903)
  – Linguist, mathematician, philosopher
  – One of the earliest people to propose connectionist architecture
  – Anticipated much of modern ideas