Lecture 1: ECE 239AS, Neural Networks & Deep Learning

General info:

EE 239AS, Neural Networks & Deep Learning
Winter quarter, UCLA AY 2017-18.

Instructor:

Prof. Jonathan Kao
Office: 56-147H

TAs:

Cheng Zheng

Tianwei Xing

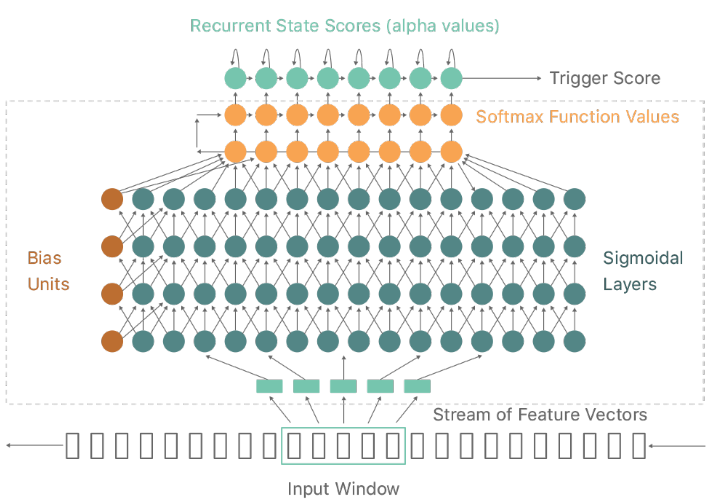
Some high-level thoughts about this class:

- First time this class is being taught at UCLA (and my first time teaching this material).
- Deep learning is a very relevant topic for many different applications today, and we hope this class will be useful for people going to a diversity of areas.
- In many ways, we're living in a Renaissance, and that's exciting.
- Hopefully the class will be fun. I really mean that.
- Ask questions!

Why a class on deep learning?

You've probably interacted deep learning technology somewhere today.

- Siri, Alexa, Cortana

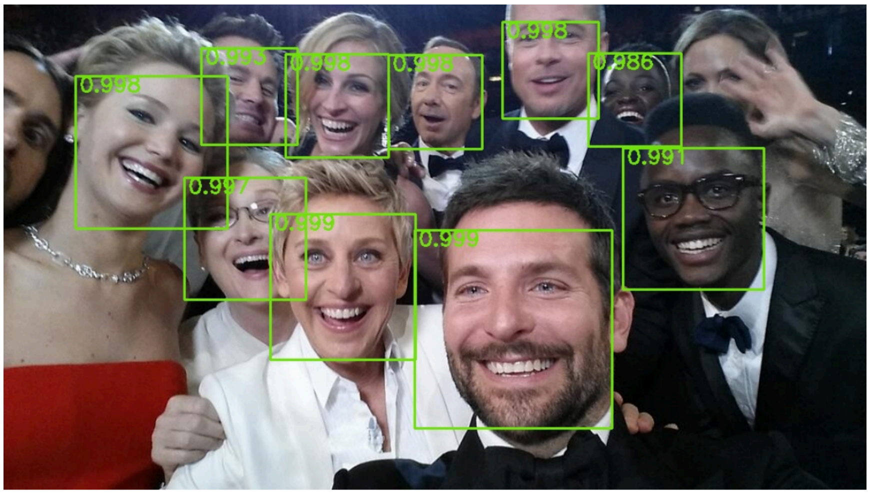


https://machinelearning.apple.com/2017/10/01/hey-siri.html

Why a class on deep learning?

You've probably interacted deep learning technology somewhere today.

- Face detection (e.g., Facebook)



Deep Dense Face Detector, https://arxiv.org/pdf/1502.02766.pdf

Why a class on deep learning?

You've probably interacted deep learning technology somewhere today.

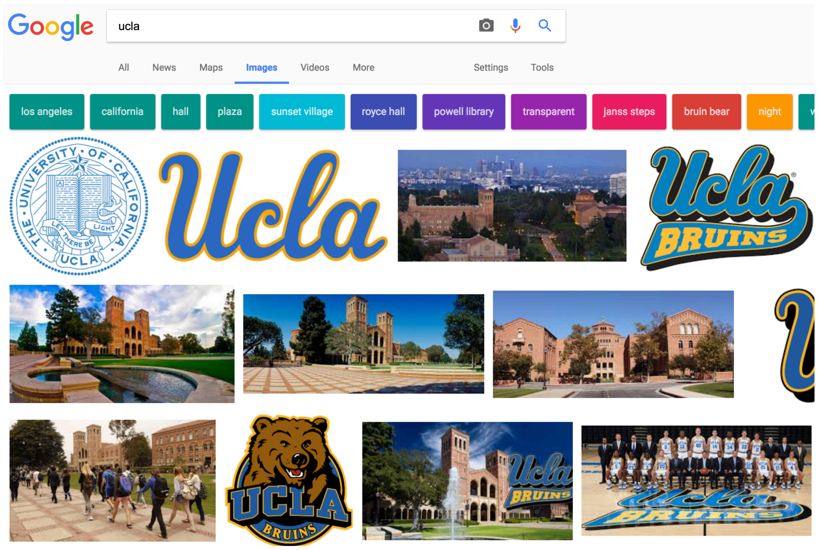
- Image detection (by Prof. Fei-Fei Li)

https://www.youtube.com/watch?v=40riCqvRoMs

Why a class on deep learning?

You've probably used deep learning technology somewhere today.

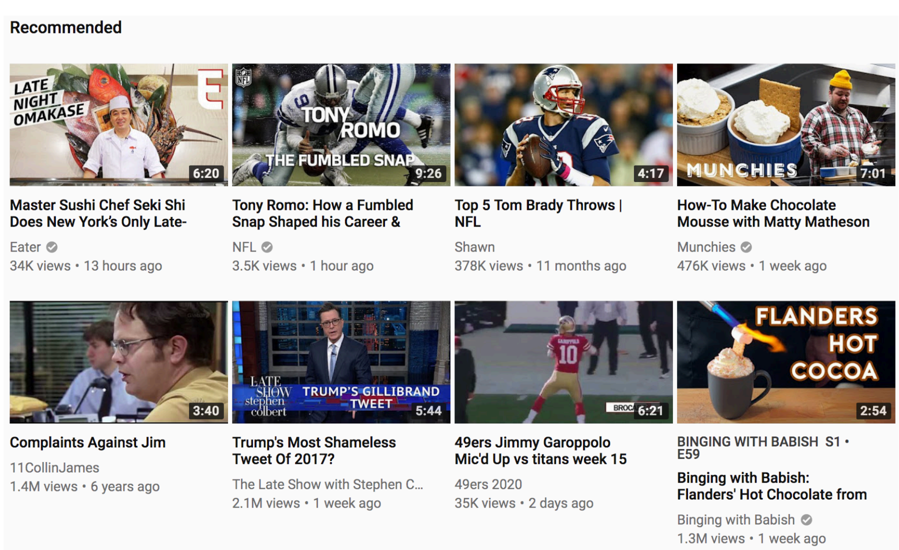
- Improving Google image search



Why a class on deep learning?

You've probably used deep learning technology somewhere today.

- Video recommendations on YouTube.



Why a class on deep learning?

You've probably used deep learning technology somewhere today.

- Netflix vectorflow (https://medium.com/@NetflixTechBlog/introducing-vectorflow-fe10d7f126b8)
- Gmail priority inbox and spam (e.g., https://gizmodo.com/gmail-now-uses-artificial-neural-networks-to-sniff-out-1716975952)
- Yelp's deep learning for classifying images (and likely more: https://engineeringblog.yelp.com/2015/10/how-we-use-deep-learning-to-classify-business-photos-at-yelp.html)
- … and many more.

Why a class on deep learning?

Its applications are diverse.

Why a class on deep learning?

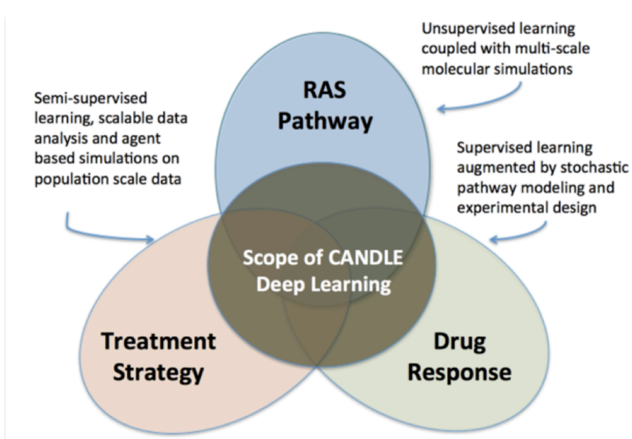
Its applications are diverse.

https://www.youtube.com/watch?v=4HCE1P-m1l8

Why a class on deep learning?

Deep learning in important applications.

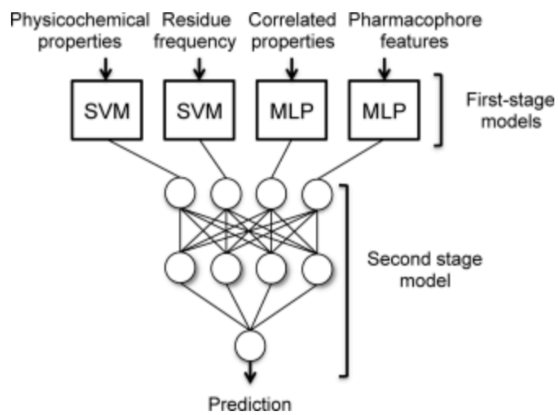
- Fraud detection (e.g., https://www.technologyreview.com/s/545631/how-paypal-boosts-security-with-artificial-intelligence/)
- Many medical endeavors, including:
  - Cancer moonshot: http://candle.cels.anl.gov/



Why a class on deep learning?

Deep learning in important applications.

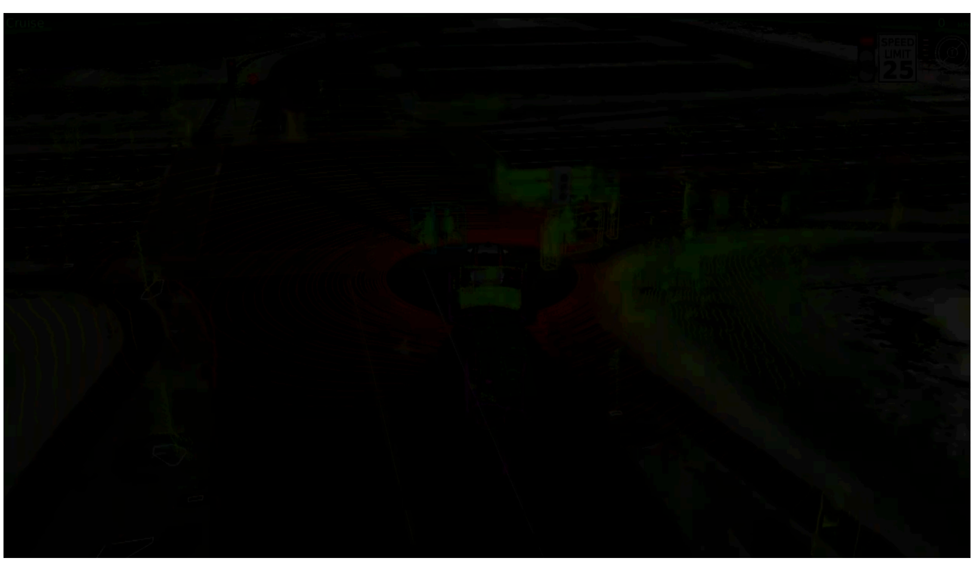
- Fraud detection (e.g., https://www.technologyreview.com/s/545631/how-paypal-boosts-security-with-artificial-intelligence/)
- Many medical endeavors, including:
  - Cancer moonshot: http://candle.cels.anl.gov/
  - Drug discovery (e.g., http://onlinelibrary.wiley.com/doi/10.1002/minf.201501008/full)



Why a class on deep learning?

New technologies.

- Self-driving vehicles.



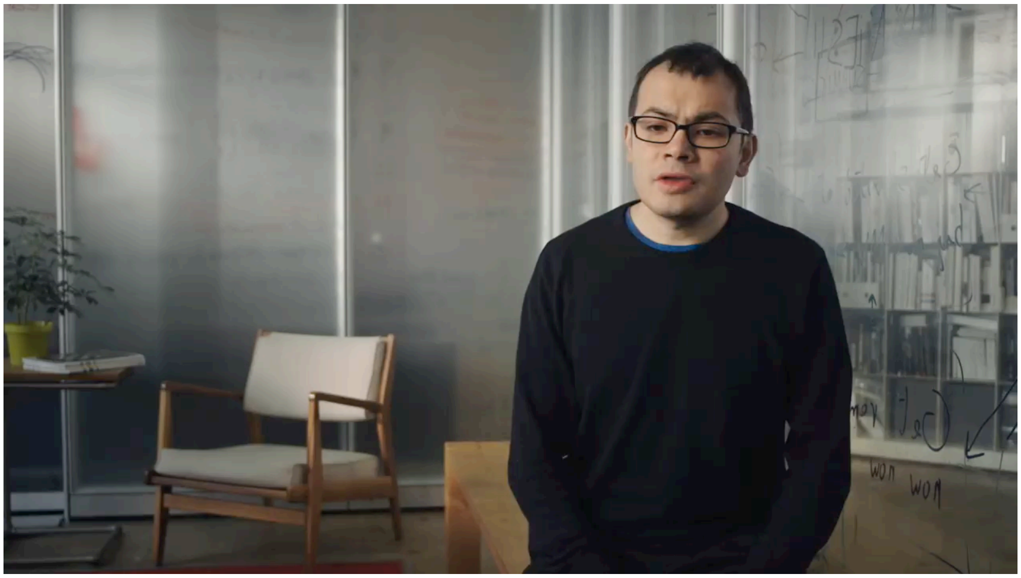
Chris Urmson TED talk, https://www.youtube.com/watch?v=tiwVMrTLUWg

Prof J.C. Kao, UCLA ECE

Why a class on deep learning?

Things of the future becoming real today.

- The game of Go.

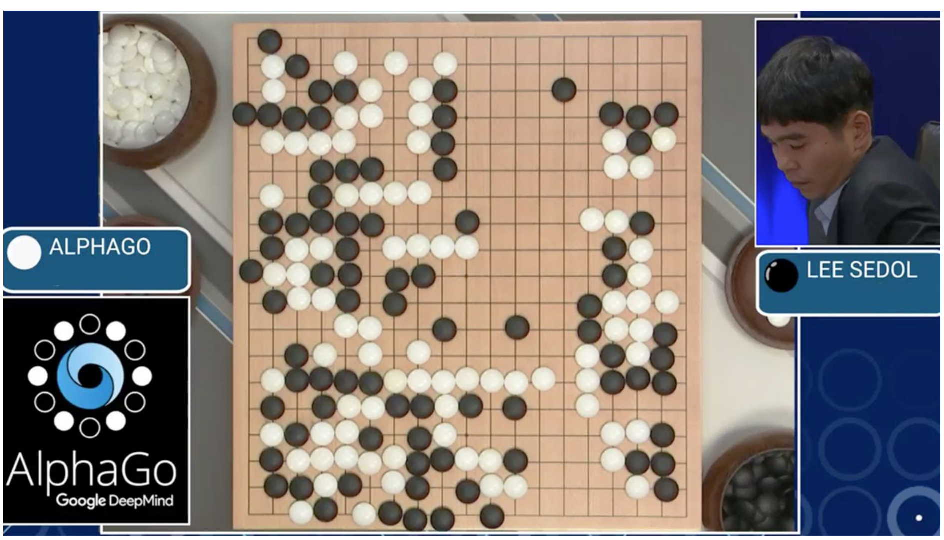


https://www.youtube.com/watch?v=SUbqykXVx0A

Why a class on deep learning?

Things of the future becoming real today.

- Game of Go solved. (AlphaGo def Lee Sedol 4-1, 2016.)



Game 3 of AlphaGo vs Lee Sedol, https://www.youtube.com/watch?v=_OV0HIj8Fb8

Why a class on deep learning?

Things of the future becoming real today.

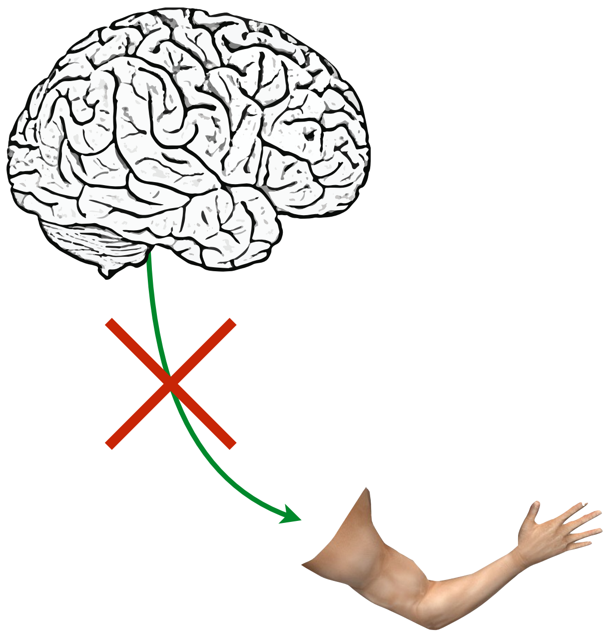
- AI’s not even requiring human expert data anymore.

Why a class on deep learning?

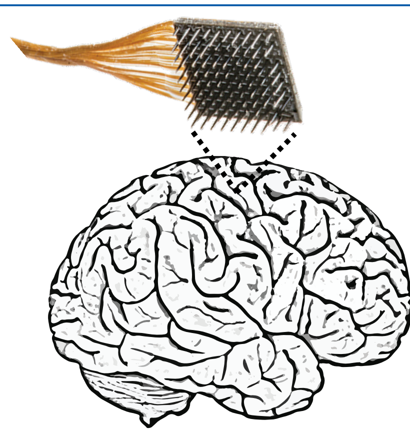
Something closer to my area of expertise:

- Brain-machine interfaces.

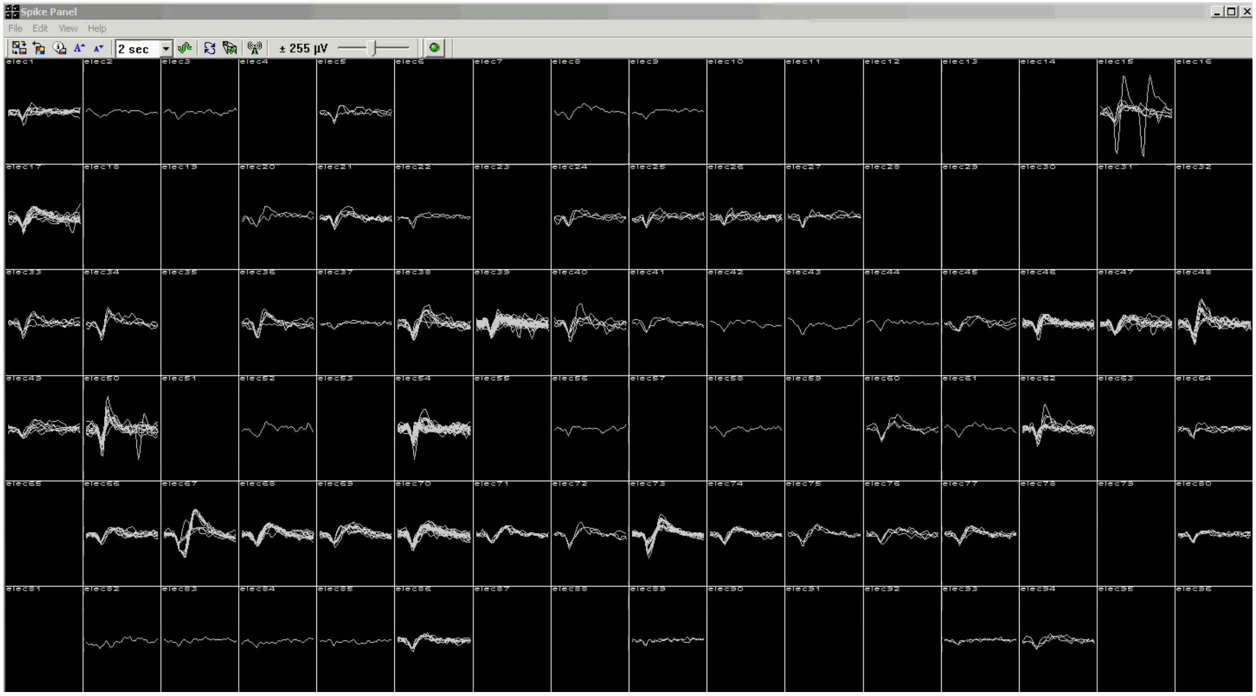
Why a class on deep learning?



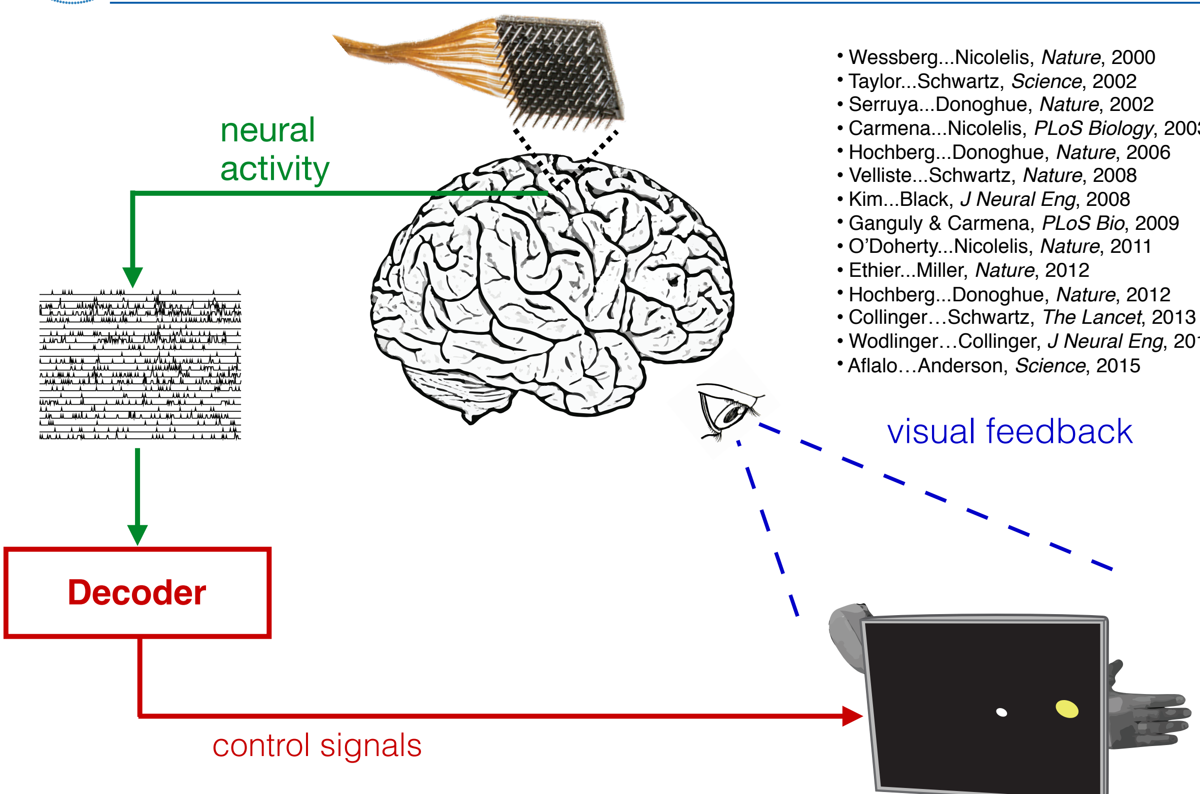
Why a class on deep learning?



Why a class on deep learning?



Why a class on deep learning?



- Wessberg...Nicolelis, *Nature*, 2000
- Taylor...Schwartz, *Science*, 2002
- Serruya...Donoghue, *Nature*, 2002
- Carmena...Nicolelis, *PLoS Biology*, 2003
- Hochberg...Donoghue, *Nature*, 2006
- Velliste...Schwartz, *Nature*, 2008
- Kim...Black, *J Neural Eng*, 2008
- Ganguly & Carmena, *PLoS Bio*, 2009
- O'Doherty...Nicolelis, *Nature*, 2011
- Ethier...Miller, *Nature*, 2012
- Hochberg...Donoghue, *Nature*, 2012
- Collinger…Schwartz, *The Lancet*, 2013
- Wodlinger…Collinger, *J Neural Eng*, 2015
- Aflalo…Anderson, *Science*, 2015

Why a class on deep learning?

Free-paced typing using the OPTI-II keyboard

“How did you encourage your sons to practice music?”

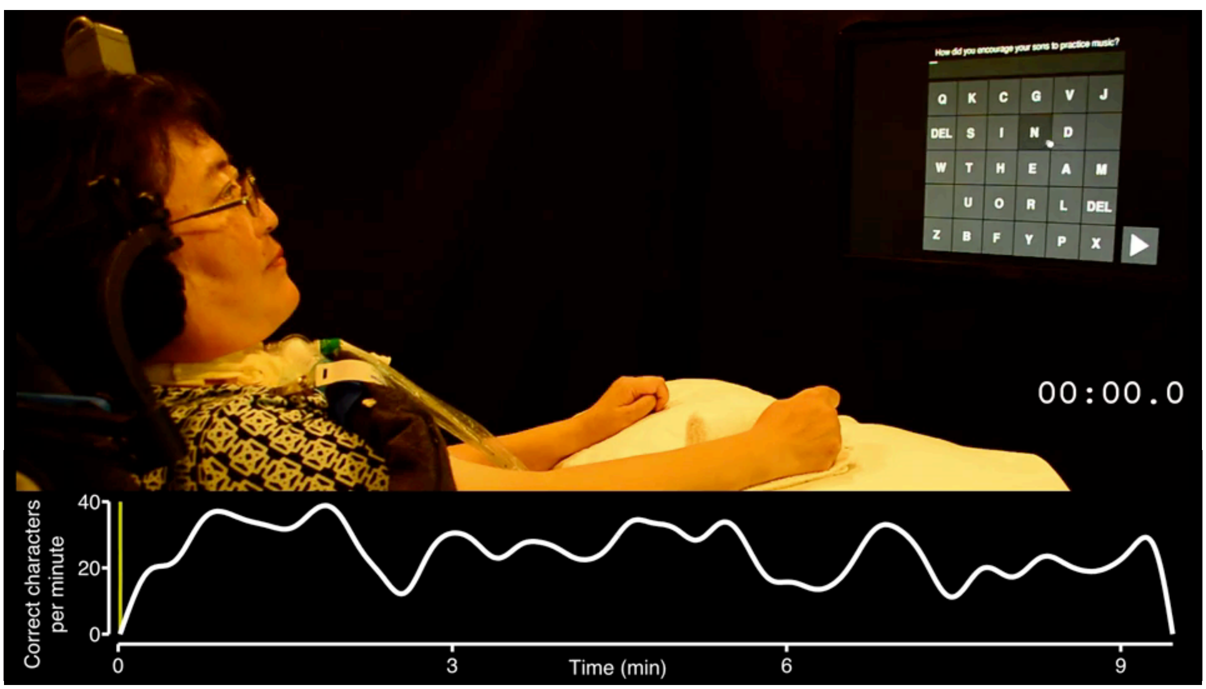
Algorithm: ReFIT-KF + HMM

(Kao, Nuyujukian*, et al., IEEE TBME 2016)*

Pandarinath, Nuyujukian*, et al., eLife 2017*

BrainGate2 Pilot Clinical Trial at Stanford University
Caution: Investigational Device.
Limited by Federal Law to Investigational Use.

Why a class on deep learning?



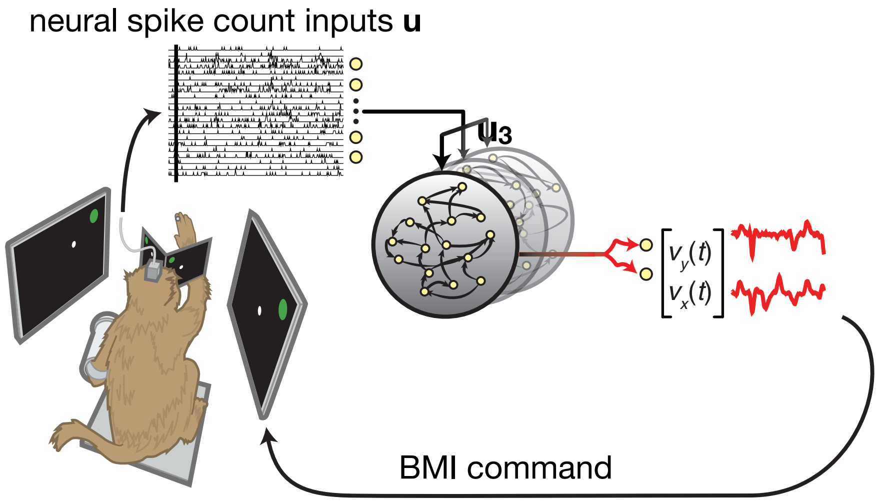
Algorithm: ReFIT-KF + HMM

(Kao, Nuyujukian*, et al., IEEE TBME 2016)*

Pandarinath, Nuyujukian*, et al., in revision*

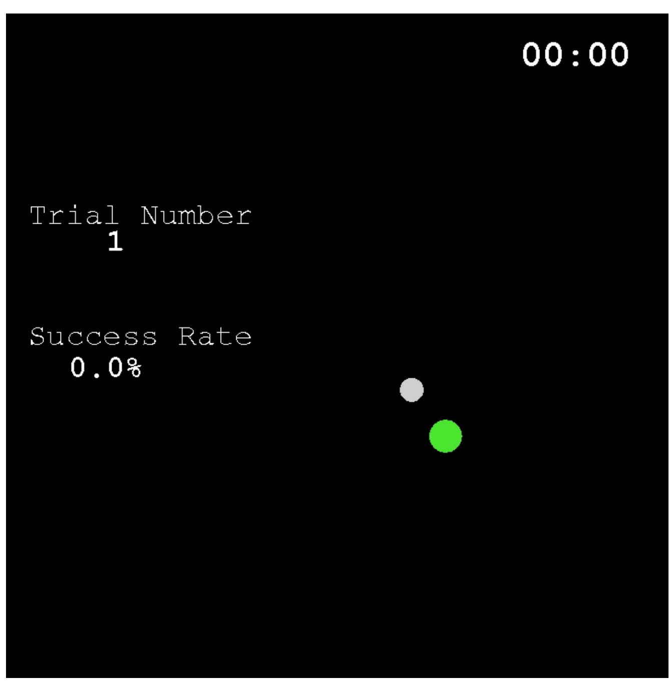
BrainGate2 Pilot Clinical Trial at Stanford University
Caution: Investigational Device.
Limited by Federal Law to Investigational Use.

Why a class on deep learning?



Sussillo*, Stavisky*, Kao* (co-first author), Ryu, Shenoy, Nat Comm 2016

Why a class on deep learning?



ECE C143A/C243A: Spring quarter 2018, where we discuss techniques to analyze neural signals from the brain and build brain-machine interfaces.

Sussillo*, Stavisky*, Kao* (co-first author), Ryu, Shenoy, Nat Comm 2016

Deep Learning is becoming ubiquitous

- We see that deep learning is enabling AI's to become an everyday part of life.
- It has resulted in key breakthroughs in many areas; in many different fields, many groups are looking towards deep learning to achieve better performance.
- Deep learning may be useful to research in **your** area / future work.
- There's lots of potential for increasing technology development.
- A brief history of Neural Networks upcoming…

A brief history of neural networks

- Neural networks have been around for a long time, but have only become relevant as of late.
- A few questions than come up:
  - Why did research into neural networks "stall" for so long?
  - What revived neural network research today?
  - What are the main differences between initial research into neural networks and this renaissance of neural networks?
- Much of this history is taken from the review by Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, 2015.

Earliest neural networks

- Neural networks date back to McCulloch and Pitts, 1943, and were inspired by nervous system activity. These networks did not learn.

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

We shall make the following physical assumptions for our calculus.

1. The activity of the neuron is an "all-or-none" process.
2. A certain fixed number of synapses must be excited within the period of latent addition in order to excite a neuron at any time, and this number is independent of previous activity and position on the neuron.
3. The only significant delay within the nervous system is synaptic delay.
4. The activity of any inhibitory synapse absolutely prevents excitation of the neuron at that time.
5. The structure of the net does not change with time.

Earliest neural networks

- Neural networks date back to McCulloch and Pitts, 1943, and were inspired by nervous system activity. These networks did not learn.

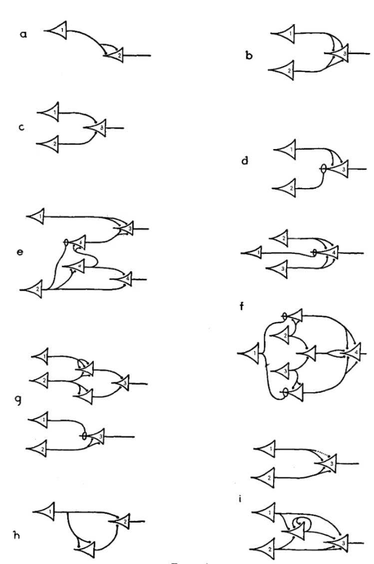
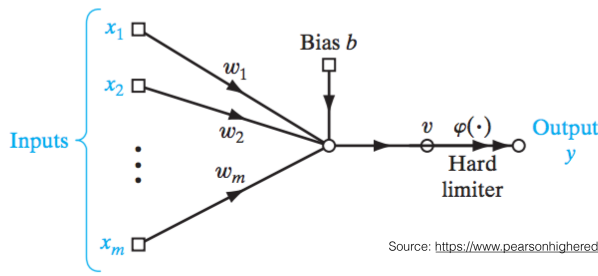


Figure 1

Rosenblatt's perceptron

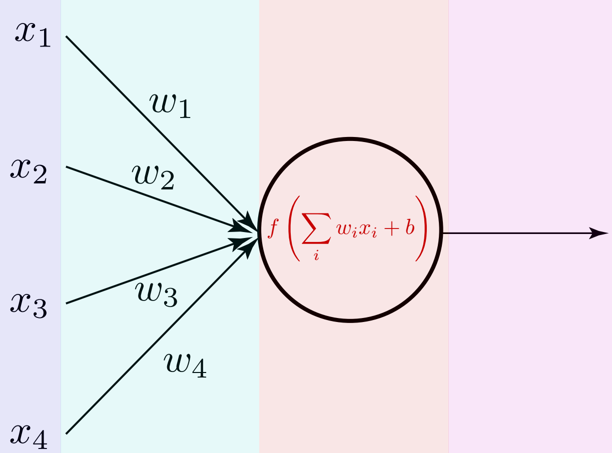
- In 1958, Rosenblatt (a psychologist) proposed the perceptron.
- This perceptron, also inspired by the nervous system, had a learning rule to train it to perform classification.



Source: https://www.pearsonhighered.com/assets/samplechapter/0/1/3/1/0131471392.pdf

Model of a neuron

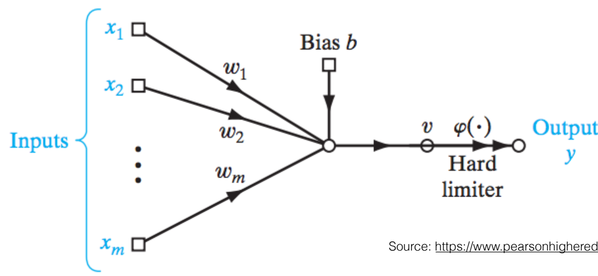
Neuroscience	Axonal activity of prior neurons	Synaptic weights	Axon hillock (integration and threshold)	Axonal output



Neural network	Outputs of prior artificial neurons	Artificial weights	Artificial neuron compute (sum and nonlinearity)	Artificial neuron output

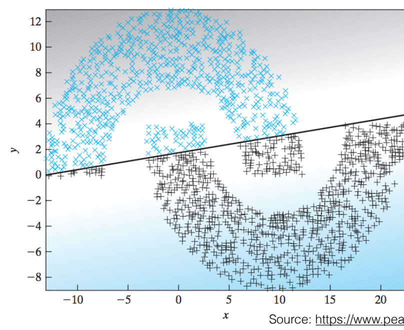
Rosenblatt's perceptron

- In 1958, Rosenblatt (a psychologist) proposed the perceptron.
- This NN, also inspired by the nervous system, had a learning rule to train it to perform classification.



Source: https://www.pearsonhighered.com/assets/samplechapter/0/1/3/1/0131471392.pdf

- It is a linear classifier.



Source: https://www.pearsonhighered.com/assets/samplechapter/0/1/3/1/0131471392.pdf

From one layer to multiple layers

- A criticism of Rosenblatt's perceptron was that it was a single layer and thus could only do linear classification. Concretely, it could not solve the XOR problem, and to do so would require multiple layers for which Rosenblatt's learning rule did not work (Minsky & Papert, 1969)
- An important step to training neural networks with multiple layers is backpropagation. Backpropagation was introduced for solving other problems.
- Paul Werbos *suggested* this could be used to train NN's in 1974.
- We'll cover these topics (in technical depth) in later lectures.

Research into NN's largely quietens

- A book by Minsky & Papert, on the limitations of the perceptron learning rule (1969) discouraged researchers from further studying neural networks (Schmidhuber, 2015).

- Paul Werbos recollects that time:

> *"Minsky's book was best known for arguing that (1)* ***we need to use MLPs even to represent simple nonlinear functions such as the XOR mapping****; and (2)* ***no one on earth had found a viable way to train MLPs good enough to learn such simple functions****. Minsky's book convinced most of the world that neural networks were a discredited dead-end – the worst kind of heresy. Widrow has stressed that this pessimism, which squashed the early "perceptron" school of AI, should not really be blamed on Minsky. Minsky was merely summarizing the experience of hundreds of sincere researchers who had tried to find good ways to train MLPs, to no avail. …*
>
> *…* ***But the pessimism at that time became terminal.*** *In the early 1970s, I did in fact visit Minsky at MIT. I proposed that we do a joint paper showing that MLPs can in fact overcome the earlier problems … But Minsky was not interested. In fact, no one at MIT or Harvard or any place I could find was interested at the time."*
>
> - Paul Werbos in *Backwards Differentiation in AD and Neural Nets: Past Links and New Opportunities*

- Another factor was limited computational power.

Some research lingers

- While neural network research lingered in the 1980's and 90's, some work remained.

- In particular, some networks started to use biological inspiration from the visual system.

- Hubel & Wiesel, in 1962, published work where they recorded from cat V1 and learned some key properties of V1 that would be in the inspiration for some future neural network architectures.

RECEPTIVE FIELDS, BINOCULAR INTERACTION AND FUNCTIONAL ARCHITECTURE IN THE CAT'S VISUAL CORTEX

BY D. H. HUBEL AND T. N. WIESEL

From the Neurophysiology Laboratory, Department of Pharmacology Harvard Medical School, Boston, Massachusetts, U.S.A.

Hubel and Wiesel’s findings

- Hubel & Wiesel, in 1962, recorded from cat V1 and learned some key properties of V1 that would be in the inspiration for some future neural network architectures.
- V1 cells have local receptive fields, and could detect features like edges, corners, end points, etc.
- V1 has simple cells and complex cells.
- Simple cells can be described by a linear model followed by rectification.
- Complex cells are invariant to the positions of features.

Fukushima’s Neocognitron

- Fukushima’s Neocognitron (1982) incorporated these insights from the visual system into a new architecture, based off of experimental results from Hubel and Wiesel.

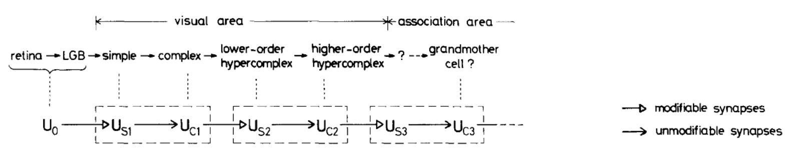


Fig. 1. Correspondence between the hierarchy model by Hubel and Wiesel, and the neural network of the neocognitron

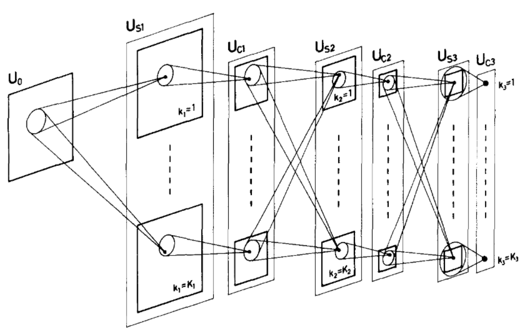
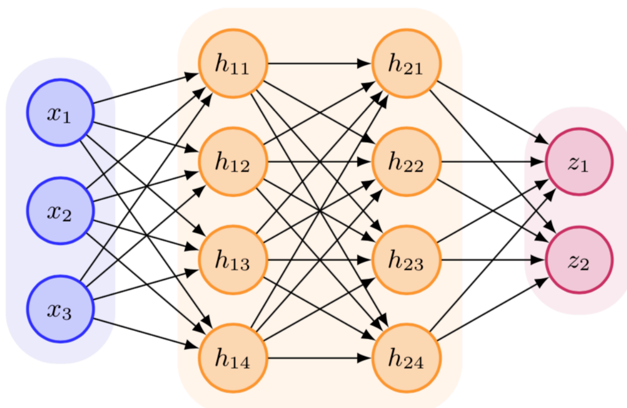


Fig. 2. Schematic diagram illustrating the interconnections between layers in the neocognitron

Backpropagation

- Rumelhart, Hinton & Williams (Nature, 1986) used backpropagation to train neural networks. Though it had been done before, this popularized neural networks.

- This provided a way to train these multi-layer perceptrons (which we will call feedforward neural networks).



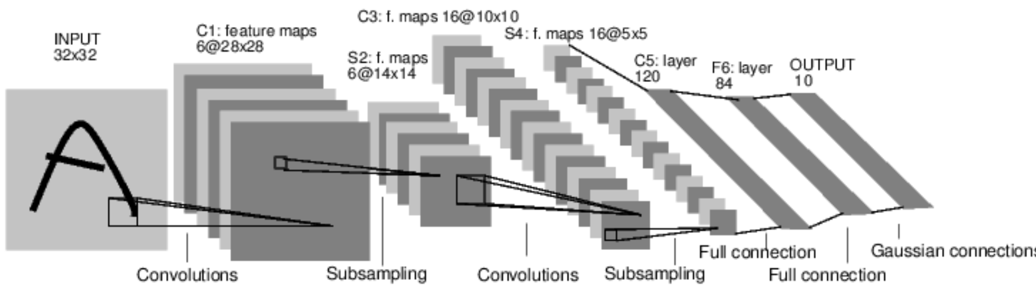
Other developments in NN's in the 80/90's

- 1989: A universal approximator theory for MLP's.
- 1989: LeCun and colleagues at Bell Labs use NN's to recognize handwritten zipcodes (MNIST).

Demo from 1983: https://www.youtube.com/watch?v=FwFduRA_L6Q

Convolutional neural networks

- In 1998, Yann LeCun introduced LeNet, which is the modern convolutional neural network.
- It is similarly inspired by visual cortex experiments.



Does CNN find features used by visual cortex?

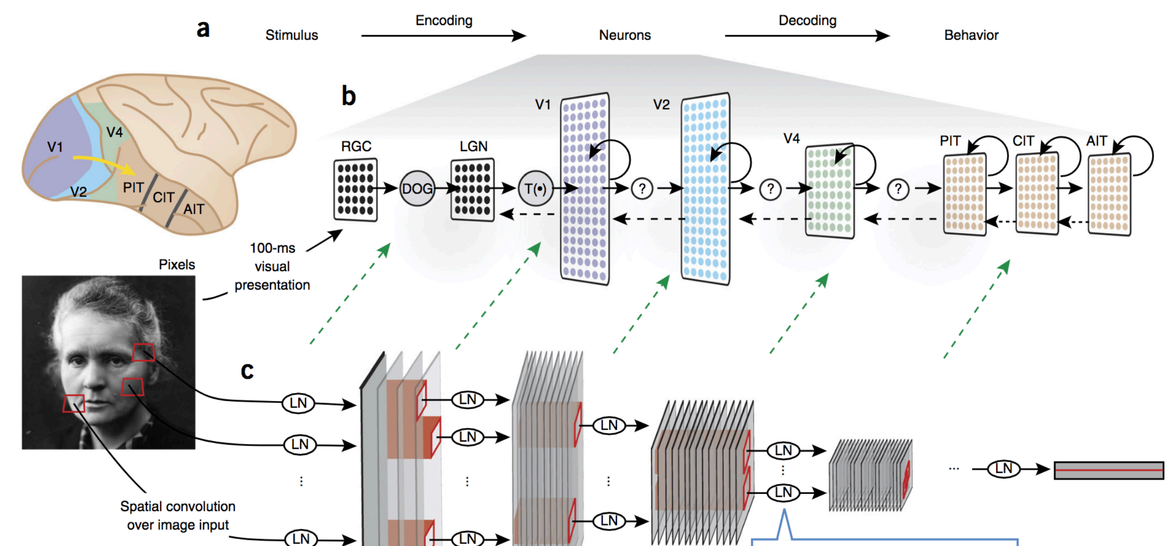


Figure 1 HCNNs as models of sensory cortex. (**a**) The basic framework in which sensory cortex is studied is one of encoding—the process by which stimuli are transformed into patterns of neural activity—and decoding, the process by which neural activity generates behavior. HCNNs have been used to make models of the encoding step; that is, they describe

Yamins & DiCarlo, 2016

What happened?

- If CNNs and backpropagation existed in 1998, why didn't it develop widespread use?
- One key obstacle is termed the *Fundamental Problem of Deep Learning*.
  - Backprop was good for training shallow neural networks, but not *deep* neural networks.
  - As networks are deeper, the back-propagated derivative becomes more inaccurate, and will either explode or vanish.
  - This results in the neural networks being very difficult to train.
  - As a result, research in neural networks stagnated as many believed that these networks could not be adequately trained.

Tackling computer vision

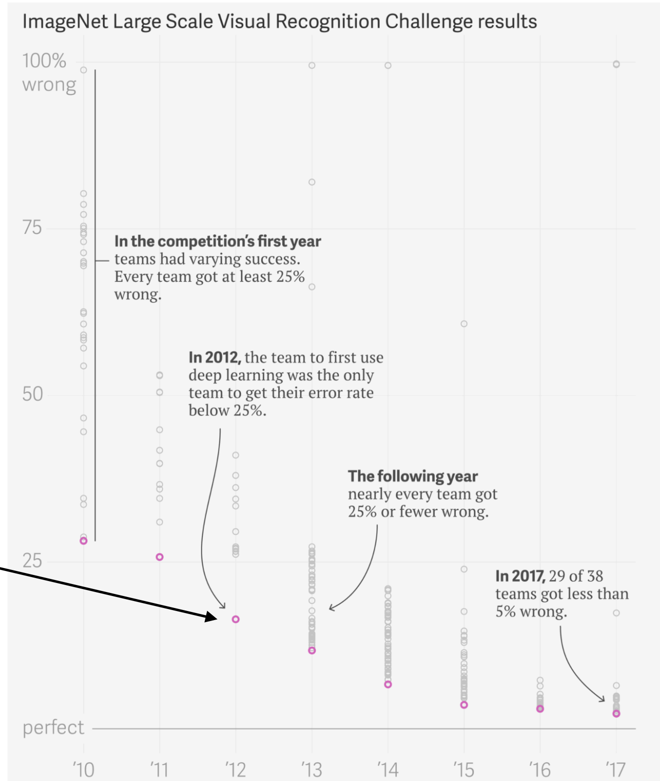
- IMAGENET

Overall

- Total number of non-empty synsets: 21841
- Total number of images: 14,197,122
- Number of images with bounding box annotations: 1,034,908

http://image-net.org/about-stats

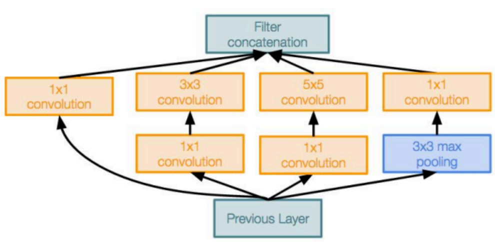
A big splash



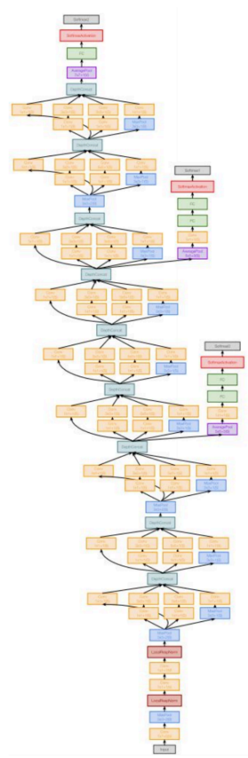
AlexNet

A big splash

- GoogLe Net example.



Inception module



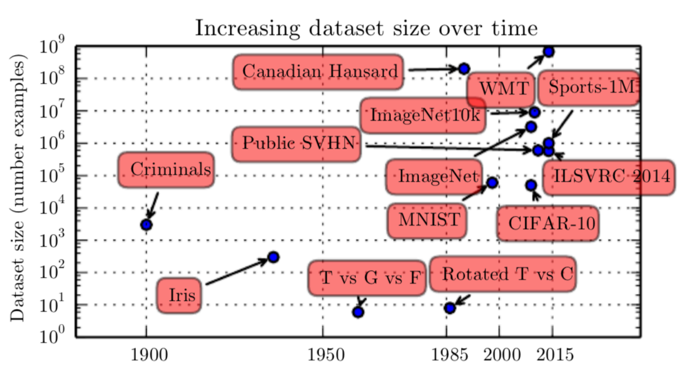
Data and technology

- Part of what has driven deep learning today is the amount of data we have and the computational power we have to process it.
- This includes larger models as computer infrastructure improves.

Fortunately, the amount of skill required [to get good performance from a deep learning algorithm] reduces as the amount of training data increases.
(Goodfellow, p. 18)

Larger dataset sizes

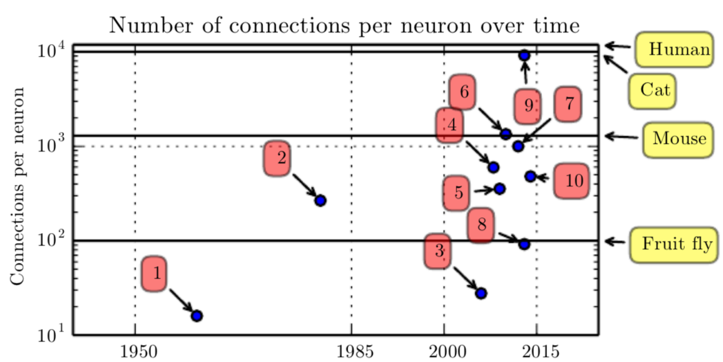
Fortunately, the amount of skill required [to get good performance from a deep learning algorithm] reduces as the amount of training data increases.
(Goodfellow, p. 18)

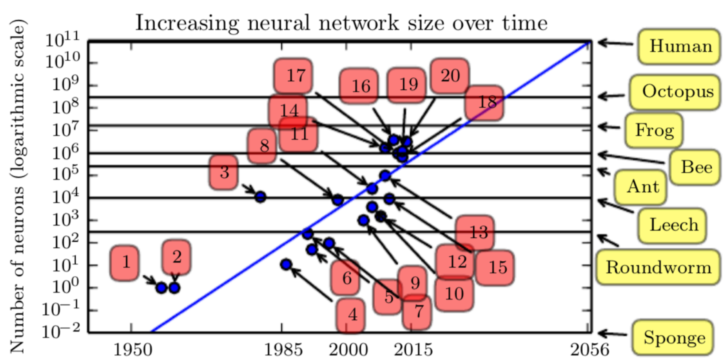


Deep Learning

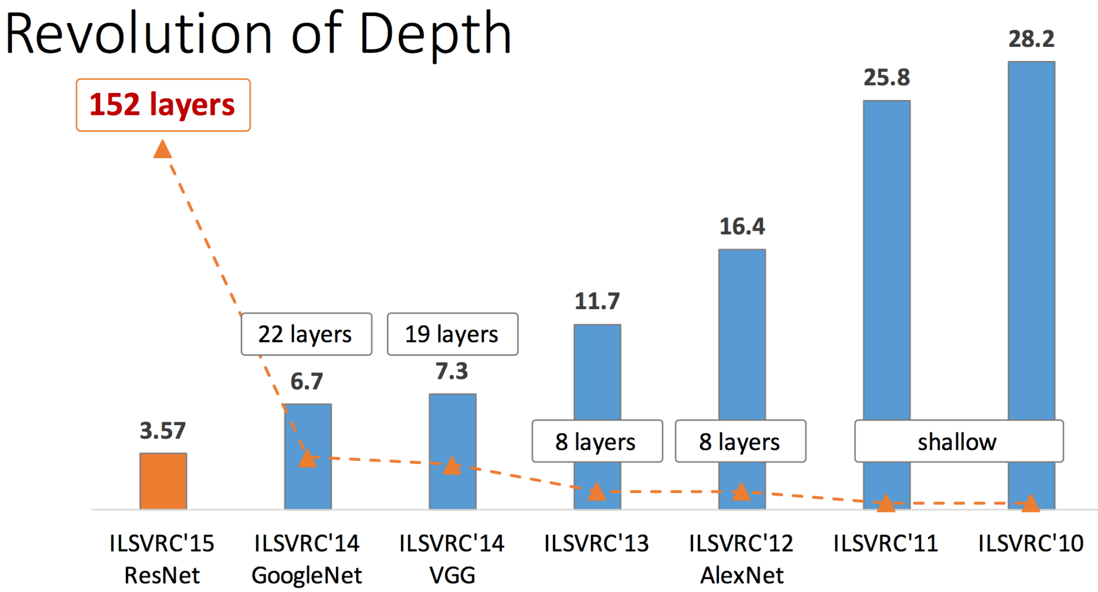
- Hardware development, with some GPUs designed specifically for deep learning (rather than video games) have accelerated the training of bigger models.

Larger networks





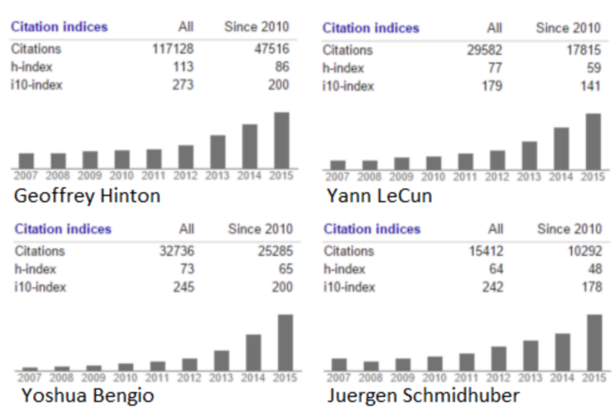
The convolutional neural network revolution



http://kaiminghe.com/icml16tutorial/icml2016_tutorial_deep_residual_networks_kaiminghe.pdf

Deep Learning

In short, deep learning has become more and more widespread.



http://www.andreykurenkov.com/writing/a-brief-history-of-neural-nets-and-deep-learning-part-4/

Deep Learning

- An interesting thought from Goodfellow: Deep learning persisted, largely with the hope that from the brain, a single deep learning algorithm can solve many different tasks (p. 15 of Deep Learning book).
- Most deep learning today is not interested in neuroscience.

  I am a computational neuroscientist / neural engineer.

  And my research group's interests in deep learning is primarily in neuroscience and neural engineering applications.
- With that, let's get to the details of this class.

A word on PTE's

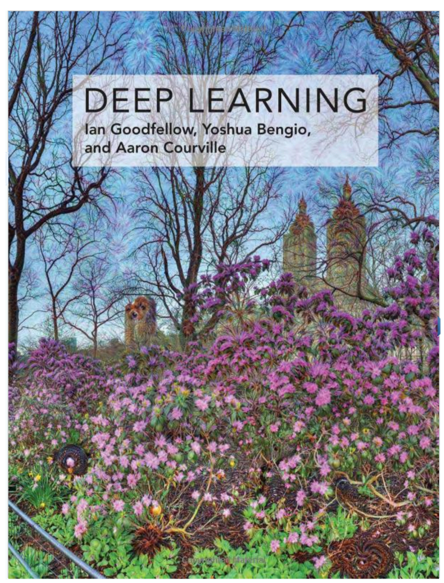
- I do truly want this class to be open for all.
- However, after much thought, this first pass of ECE 239AS will have enrollment limited.
- We have tried to keep enrollment at about 160.
- We are no longer giving out any PTE's, and will implement normal waitlist rules.
- I am fine with people auditing, and because of the scenario we are in, I also publicly post the notes we will go over in the class so you may follow along.

Breaks.

- My personal preference: 1 hr 50 minutes is a really long time to lecture, and to learn.
- I plan to give a 5-10 minute break in between lectures.
- I will show a timer up here on the screen.

A (TENTATIVE) outline of this class

Date (2018)	Lecture	Content
08 Jan	1	Overview to deep learning
10 Jan	2	Introduction to machine learning I
15 Jan	-	MLK Holiday
		HW #1 released, due 22 Jan
17 Jan	3	Introduction to machine learning II
22 Jan	4	Supervised classification & gradient descent principles
		HW #2 released, due 29 Jan
24 Jan	5	Fully connected neural networks
29 Jan	6	Backpropagation
		HW #3 release, due 05 Feb
31 Jan	7	Regularizations for training neural networks
05 Feb	8	Optimization for training neural networks
		HW #4 released, due 12 Feb
07 Feb	9	Convolutional Neural Networks I
12 Feb	10	Convolutional Neural Networks II
		HW #5 released, due 19 Feb
14 Feb	11	Recurrent neural networks I
19 Feb	-	Presidents' Day Holiday
21 Feb	M	Midterm, in class
		Project released, due 16 Mar
26 Feb	12	Recurrent neural networks II
28 Feb	13	Variational autoencoders I
05 Mar	14	Variational autoencoders II
07 Mar	15	Other topics in deep learning
12 Mar	16	Other topics in deep learning II
14 Mar	17	Overview



http://www.deeplearningbook.org/

On the pace of this class

- I like to err on the side of going slower, since it's important for everyone to understand the material to gain mastery.
- If you have a background in machine learning, the start of the class may feel a bit slower. I ask for your patience here.
- Please ask questions.
  - Especially your basic questions!
- The lectures are being given for the first time, and thus our estimate of how long they will take, etc. may not be as accurate as in a more mature class.
- There is a SLIM chance, that if material is taking too long, the midterm date could be postponed to a different lecture. However, I do plan to keep to the schedule. (I want to make sure CNNs are tested on the midterm.)

Class resources

- The class will be done entirely in Python and we will provide resources for setting up Python 3, as well as packages needed to be installed for each assignment.
- "Formal" class lecture notes, that most of these lectures are based on, are posted online at the class website.
- Lecture notes "in class" are informal and not publicly posted. I ask that you please not publicly post them, though we will distribute them on CCLE.

Class resources

- Annotations will *probably* not be available, unless someone figures out how to record them in Keynote.
- We are making recordings of the class, but these are for research purposes and won't be posted. We may not have the videos processed for this quarter.
- Class website: https://seas.ucla.edu/~kao/nndl

Some notes on the class

- Piazza should be used for questions. We will send out info.
- Beyond this, e-mail questions to:

  ece239as.w18@gmail.com

  We do not respond to class material related e-mails sent to our individual accounts. This is to be efficient on all of our time's and to avoid any single instructor of the course staff from becoming inundated.
- Structure:
  - 40% HW
  - 40% midterm (in class)
  - 19% final project.
  - 1% participation (course feedback at end of class)
- Thanks to resources: Deep Learning (textbook by Goodfellow, Courville, Bengio) as well as CS 231n at Stanford University (thanks to Justin Johnson and Serena Yeung for permission to use code from that class).

Some notes on the class

- 5 planned assignments.
- Assignments will contain both written components as well as coding components.
- Code in the HW will walk you through Jupyter Notebooks, and largely it'll be clear if you're doing things correctly or not.
  - Note, we will release solutions for written HW exams, but we will **not** release solution code for HW's.
  - These assignments are planned to be used in future years.
- HW will be submitted via Gradescope; they are due at 11:59PM Pacific Time on the stated due date.
- Any late assignment will receive a grade of **zero**.
- Life happens. We are giving **two late days to every student**, with the intent that these are to be used only in extenuating circumstances.

Some notes on the class

- This class will focus on implementation of neural networks and on algorithms to train them.
- It will **not** cover theory of deep learning.
- It will **not** be a theoretical class in general.
  - We aim to be rigorous in our use of notation and math.
  - But we won't be doing proofs.
  - We're more focused on application and equipping you with tools that may be helpful in your research or future industry positions.
- This is not a statement on the importance of theory. It merely reflects the priorities for this class alone.

On academic integrity

Academic integrity

UCLA embraces the core values of integrity, excellence, accountability, respect, and service through the True Bruin program

`http://www.truebruin.ucla.edu`

I take academic integrity very seriously; students caught cheating or violating these principles will face disciplinary action. Please refer to the UCLA student conduct code:

`http://www.deanofstudents.ucla.edu/portals/16/documents/UCLA%20Student%20Conduct%20Code%209-29-14%20final.pdf`

In this class, unacceptable behavior includes plagiarizing the work of others, plagiarizing code, and copying another person's exam. In accordance with UCLA policy, any instance of suspected academic dishonesty will be immediately reported to the Dean of Students Office and zero credit will be given for any work determined to be dishonest.

What I assume you know

Pre-requisites

This class requires a solid understanding in probability and linear algebra. It also requires coding experience. The class will be taught entirely in `Python`. If you have only had exposure to `MATLAB` there may be some ramp up time to familiarize yourself with `Python`.

Pre-requisite topics I will **assume** you know.

- Probability: independence, conditional probability, Bayes rule, multivariate Gaussian distribution, marginalization, expectation, variance
- Linear algebra: basic matrix operations, span, rank, range and null space, eigenvalue decomposition, singular value decomposition, pseudoinverse

About me

- Still within my first year at UCLA, my second time teaching a class at UCLA.
- Did my graduate work with Prof. Krishna Shenoy at Stanford University.
  - Did a lot of neurophysiological experiments.
  - Primarily designed brain-machine interfaces.
  - My expertise is in applying machine learning to neural systems, and neural engineering.
- Postdoc in the same lab.
  - Focus was on deep learning and its applications to neuroscience.

About Cheng & Tianwei

Questions?