Data Science Initiative for Africa (DSI-A) - Deep Learning Lecture 1

Course Introduction, History of Deep Learning, Neural Nets, and Perceptrons

Santiago Romero-Brufau
Harvard T.H. Chan School of Public Health

Instructor Team

Course Oversight University of Kawzulu-Natal:	**Prof. Henry Mwambi** Co-Principal Investigator UKZN Associate Professor, School of Mathematics, Statistics, and Computer Science Visit Prof. Mwambi's website for more information about his work. 
Lead Facilitator Harvard T.H. Chan School of Public Health:	**Dr. Santiago Romero-Brufau** Adjunct Assistant Professor, Department of Biostatistics, Harvard Chan Assistant Professor of Otorhinolaryngology and Healthcare Systems Engineering, Mayo Clinic Visit Dr. Romero-Brufau's website for more information about his work. 
Co-lead Facilitator Heidelberg Institute of Global Health:	**Dr. Sandra Barteit** Research Group Leader for Digital Global Health Heidelberg Institute of Global Health Visit Dr. Barteit's website for more information about her work. 
Facilitator:	**Dr. Mandlenkosi Gwetu** Senior Lecturer and Academic Leader for Computer Science University of KawZulu-Natal Visit Dr. Gwetu's website for more information about his work. 

Teaching Assistants

Teaching Assistant:	**Dr. Gabriel Kallah-Dagadu** WASHU Takwimu Postdoctoral Fellow University of KwaZulu-Natal 
Teaching Assistant:	**Dr. Mohanad Mohammed** WASHU Takwimu Postdoctoral Fellow University of KwaZulu-Natal 

Daily schedule

9am-10:30am: First lecture session

10:30-11am: tea break

11am-12:30pm: Second lecture session

12:30-13:30: lunch break

13:30-14:30: First lab session (or group work)

14:30-14:45: tea break

14:45-16:00: Second lab session (or group work)

16:00-16:30: office hours

Pearl of wisdom

Adam Grant @AdamMGrant · Jan 30

The first principle of psychological safety:

The harder you make it to voice problems, the harder it becomes to solve them.

The issues people are most afraid to raise are the most likely to become thorns in your side. It’s impossible to fix what you don't know is broken.

45 732 2,589 238.4K

General suggestions box

https://forms.gle/dcxysGxb9su4CCEP8

Anonymous (or you can leave your name, too).

Issues, ideas for lecture topics, ideas for guest speakers, cool news or papers, etc.

Course structure

First week

Morning: lecture

Afternoon: labs

Second week

Morning: lectures

Afternoon: group projects

Course philosophy

Lectures are mainly focused on knowledge, logic and concepts that will be useful over your career

Socratic method whenever possible (in-class interaction)

Course Content Overview

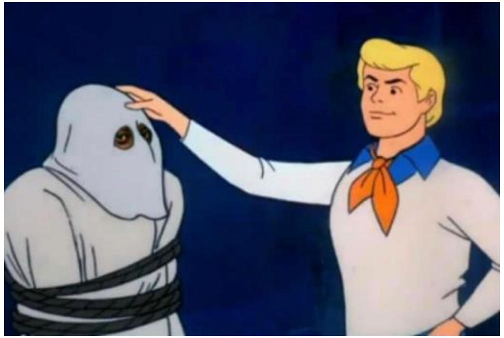
Course content:

- Computational and mathematical foundations of deep learning
- Deep learning workflow
- Bias/variance trade-off
- Feedforward networks (Multilayer perceptrons - MLPs)
- Convolutional neural networks (CNNs)
- Recursive/Recurrent neural networks (RNNs)
- Transformers / attention

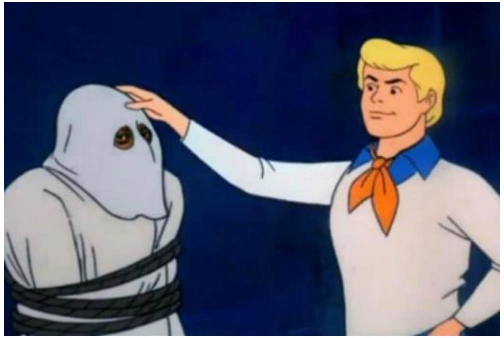
“Any sufficiently advanced technology is indistinguishable from magic.”

Arthur C. Clarke

Two ghosts, and a cool field



The Math
(the conceptual ideas, really)



The Coding

The Cool and Beautiful Things One Can Do with Deep Learning

Readings

- Most of the course content will come from these 2 books:
  - Deep Learning with Python. François Chollet. Manning Publications, 2017. Available at https://www.manning.com/books/deep-learning-with-python
  - Deep Learning. Ian Goodfellow, Yoshua Bengio, Aaron Courville. MIT Press, 2016. Book available at http://www.deeplearningbook.org
- Great explanations in StatQuest Youtube channel
- Also, journal articles: **from this list**

There's so many good courses online about deep learning…

what can this course add?

Things this course can add
(versus just looking things up online)

1) Feedback on your work
   a) PSets, final project, and labs (!)

1) A schedule, and an in-person group
   a) Recommendation: work with classmates

1) Interactivity
   a) Recommendation: ask questions! Don’t fall behind

Deep Learning

What is *Deep Learning*?

- Computers can solve problems that are intellectually difficult for humans
  - Ex: multiplication with large numbers and decimals, chess, etc.
  - Problems that can be described by a list of formal, mathematical rules

- Humans can solve problems that are intuitive, but difficult to describe formally and thus difficult for computers to solve
  - Ex: handwriting recognition, speech recognition, etc.

What is *Deep Learning*?

- Deep learning is a solution to allow computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined through its relation to simpler concepts

- The hierarchy of concepts builds on itself, producing a deep graph with many layers, leading to the concept of ***deep learning***

Artificial Intelligence

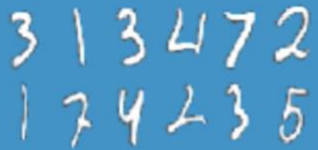
Any technique that enables computers to mimic human behavior

Machine Learning

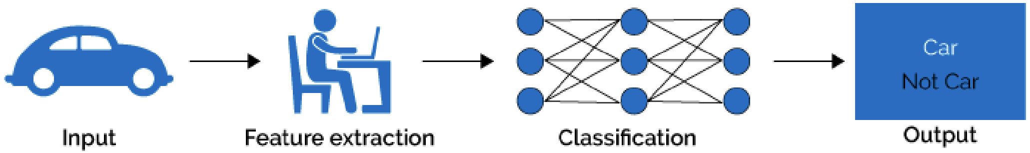
Ability to learn without explicitly being programmed

Deep Learning

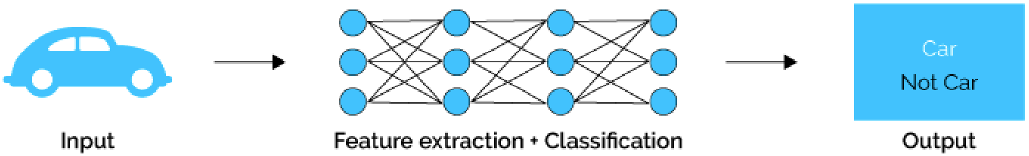
Extract patterns from data using neural networks



Machine Learning

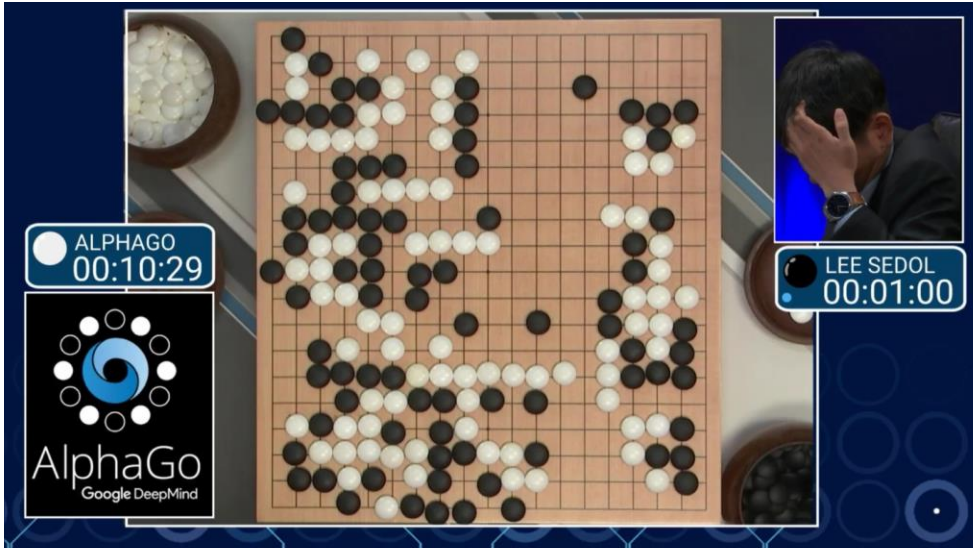


Deep Learning



Deep Learning Successes

Games

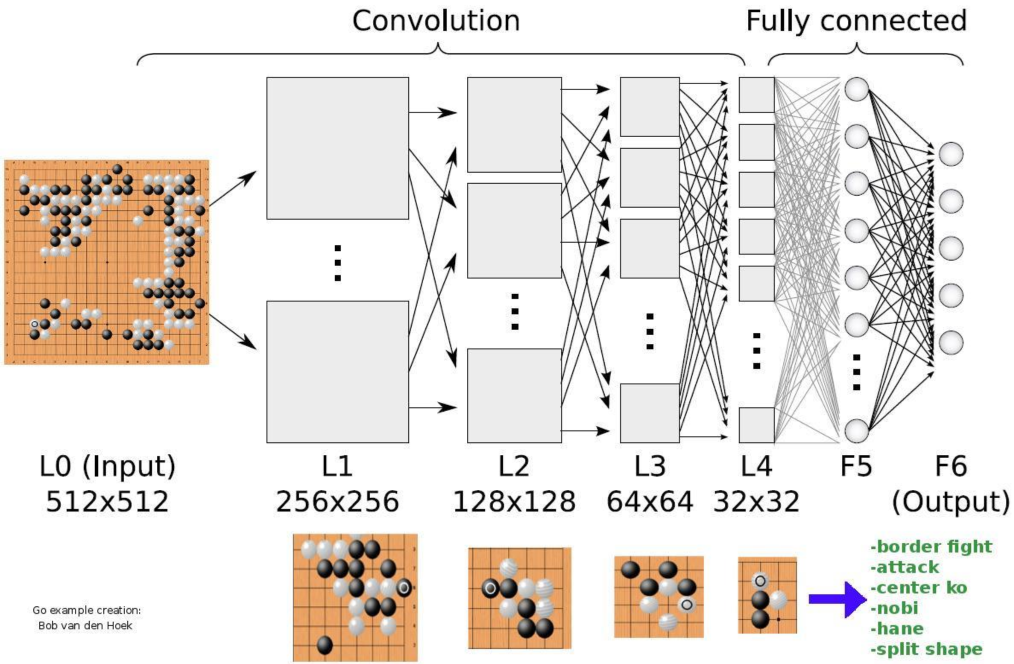


Deep learning successes

ChatGPT!

Deep Learning Successes

Games



Deep Learning Successes

Art

Deep Learning Successes

Art

(a) ***Input**: a casual face photo*

(b) ***Outputs**: new headshots with the styles transferred from the examples. The insets show the examples.*

Deep Learning Successes

[The AI-Art Gold Rush]

AI-generated "faceless portraits" by Ahmed Elgammal and AICAN. Photo: Artrendex Inc./The Atlantic

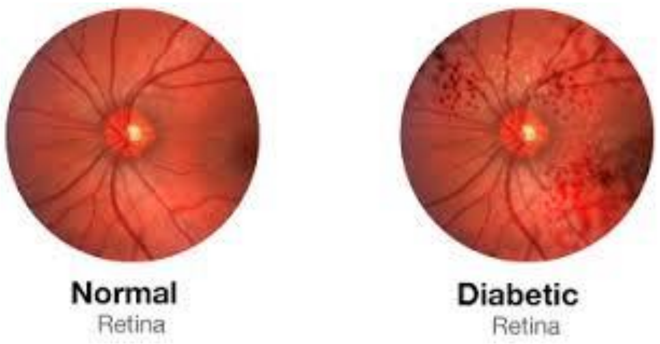
Deep Learning Successes(?)

DeepFake

- Papers and datasets
- One YouTube video
- Obama DeepFake video
  - Warning - explicit language

Deep Learning Successes

Medicine



JAMA | Original Investigation | INNOVATIONS IN HEALTH CARE DELIVERY

Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs

Varun Gulshan, PhD; Lily Peng, MD, PhD; Marc Coram, PhD; Martin C. Stumpe, PhD; Derek Wu, BS; Arunachalam Narayanaswamy, PhD; Subhashini Venugopalan, MS; Kasumi Widner, MS; Tom Madams, MEng; Jorge Cuadros, OD, PhD; Ramasamy Kim, OD, DNB; Rajiv Raman, MS, DNB; Philip C. Nelson, BS; Jessica L. Mega, MD, MPH; Dale R. Webster, PhD

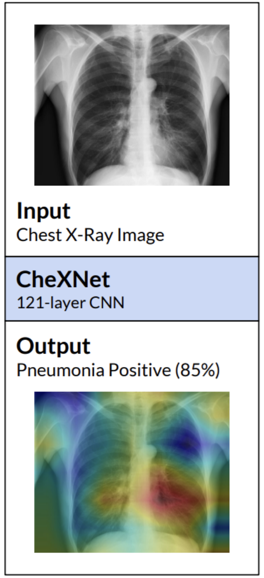
Deep Learning Successes

CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar [*1] Jeremy Irvin [*1] Kaylie Zhu [1] Brandon Yang [1] Hershel Mehta [1]
Tony Duan [1] Daisy Ding [1] Aarti Bagul [1] Robyn L. Ball [2] Curtis Langlotz [3] Katie Shpanskaya [3]
Matthew P. Lungren [3] Andrew Y. Ng [1]

Abstract

We develop an algorithm that can detect pneumonia from chest X-rays at a level exceeding practicing radiologists. Our algorithm, CheXNet, is a 121-layer convolutional neural network trained on ChestX-ray14, currently the largest publicly available chest X-ray dataset, containing over 100,000 frontal-view X-ray images with 14 diseases. Four practicing academic radiologists annotate a test set, on which we compare the performance of CheXNet to that of radiologists. We find that CheXNet exceeds average radiologist performance on the F1 metric. We extend CheXNet to detect all 14 diseases in ChestX-ray14 and achieve state of the art results on all 14 diseases.



1. Introduction

More than 1 million adults are hospitalized with pneumonia and around 50,000 die from the disease every year in the US alone (CDC, 2017). Chest X-rays

Deep Learning Successes

Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks

Pranav Rajpurkar* PRANAVSR@CS.STANFORD.EDU
Awni Y. Hannun* AWNI@CS.STANFORD.EDU
Masoumeh Haghpanahi MHAGHPANAHI@IRHYTHMTECH.COM
Codie Bourn CBOURN@IRHYTHMTECH.COM
Andrew Y. Ng ANG@CS.STANFORD.EDU

Abstract

We develop an algorithm which exceeds the performance of board certified cardiologists in detecting a wide range of heart arrhythmias from electrocardiograms recorded with a single-lead wearable monitor. We build a dataset with more than 500 times the number of unique patients than previously studied corpora. On this dataset, we train a 34-layer convolutional neural network which maps a sequence of ECG samples to a sequence of rhythm classes. Committees of board-certified cardiologists annotate a gold standard test set on which we compare the performance of our model to that of 6 other individual cardiologists. We exceed the average cardiologist performance in both recall (sensitivity) and precision (positive predictive value).

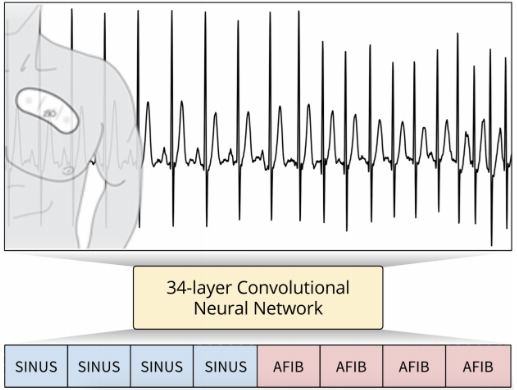
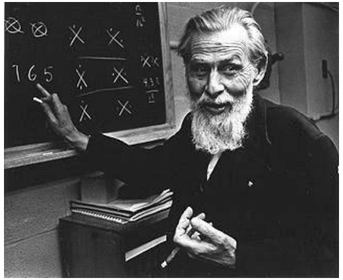


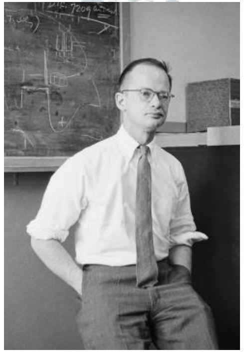
Figure 1. Our trained convolutional neural network correctly detecting the sinus rhythm (SINUS) and Atrial Fibrillation (AFIB) from this ECG recorded with a single-lead wearable heart monitor.

History of Neural Nets and Deep Learning

Neural Nets and Deep Learning Not New

- Date back to the 1940s
- Walter Pitts and Warren McCulloch
  - First notion of an artificial neuron
  - Designed to mimic the way a neuron was thought to work
  - [1943 paper]
- Frank Rosenblatt
  - “Perceptron” algorithm 1950s
  - Could recognize letters and numbers





AI Winter

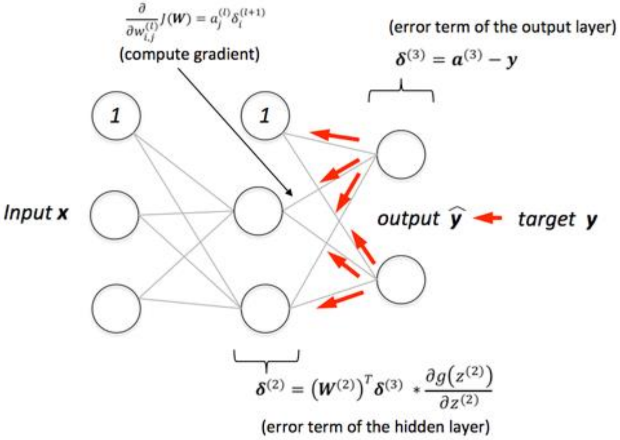
- Many cycles of boom and bust
- Repeated promises of “true AI” that were unfulfilled and followed by “AI winters” - the first in 1969
- Marvin Minsky and Seymor Papert write book about shortcomings of perceptrons and effectively kill all research on neural nets

Return of the Neural Net

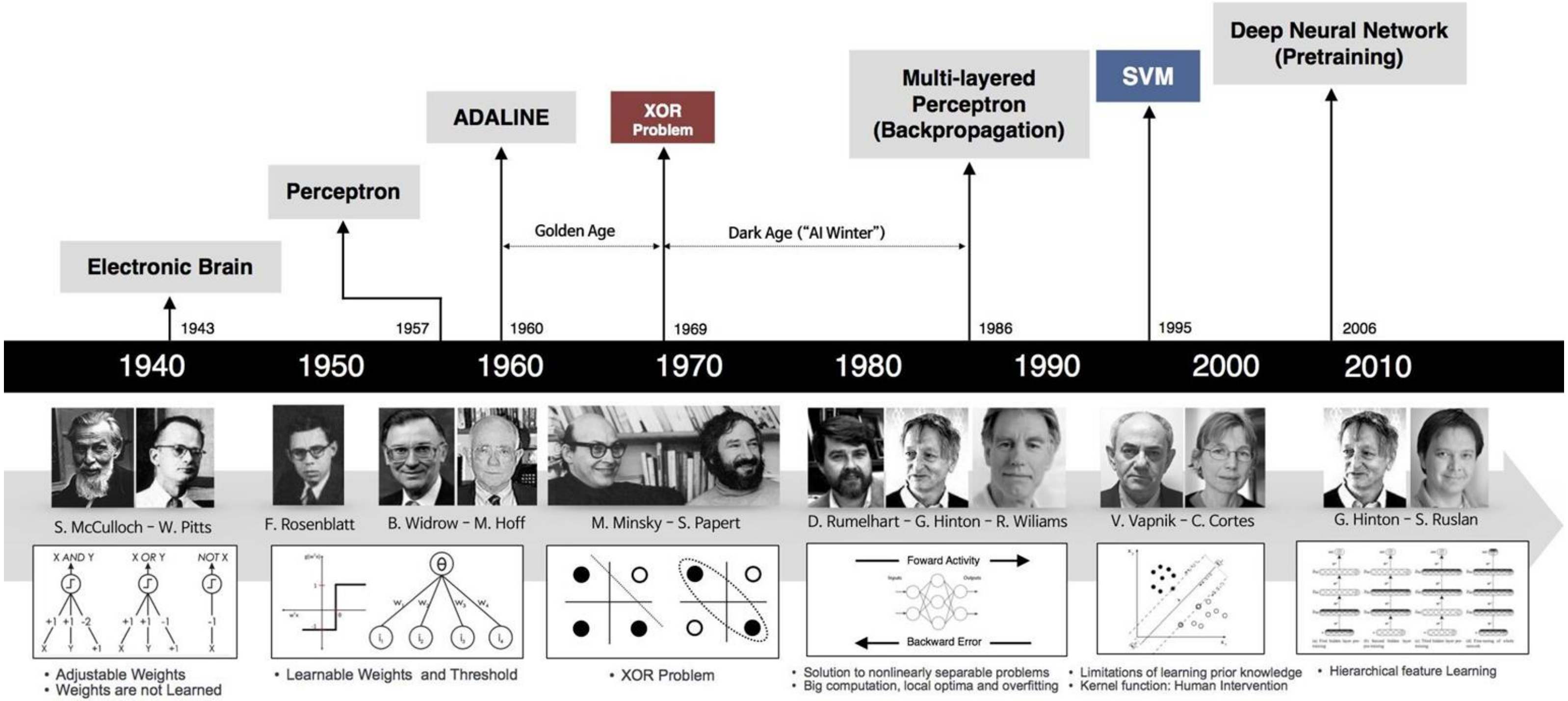
- Geoff Hinton, David Rumelhart and Ronald Williams discover back-propagation (1980s)
  - Allows neural nets to move past the limitations of perceptrons
  - Lead to convolutional neural nets (CNNs) and handwritten digits recognition
  - Problem: didn't scale ⟶ another 10-15 year AI winter

- Rebranding as "Deep Learning" (2006)
  - Unsupervised pretraining and deep belief networks
  - Could create "deeper" neural nets ⟶ "deep" learning

- Great AI Awakening (where we are now!)
  - Alexnet (2012)
  - Availability of GPUs (and TPUs) and larger data sets
  - Neural nets start surpassing humans



Deep Learning Timeline



Deep Learning Timeline

THE RISE OF AI

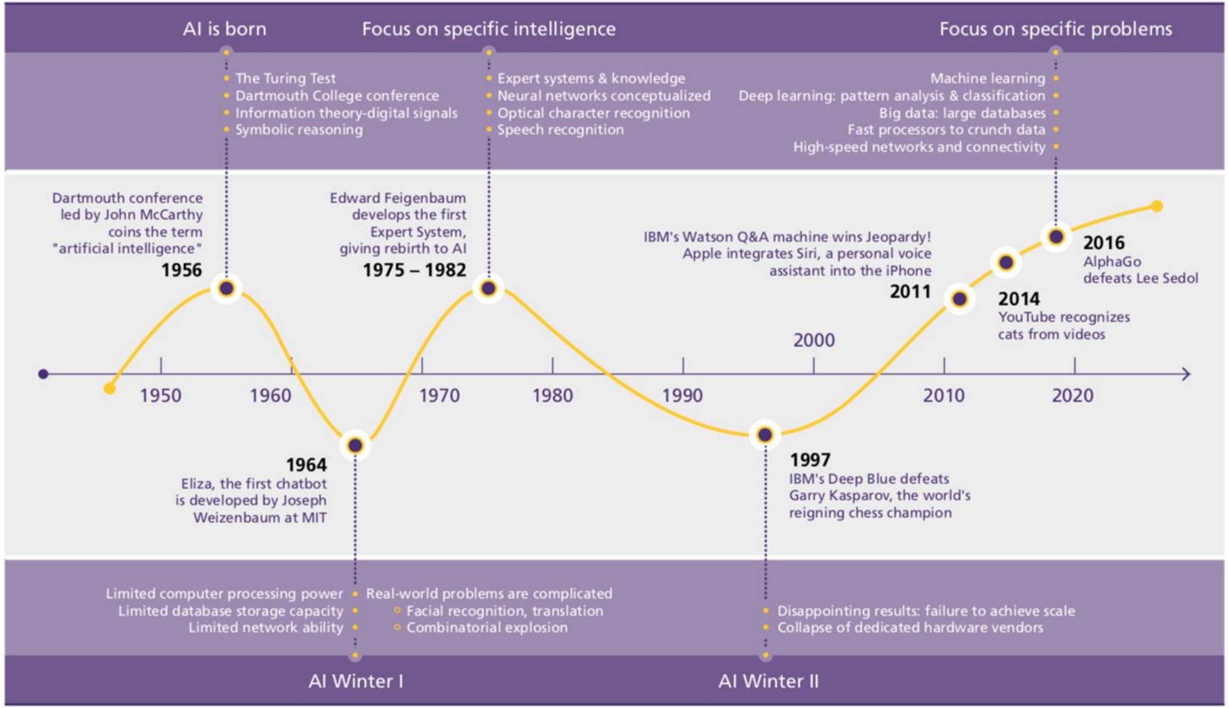
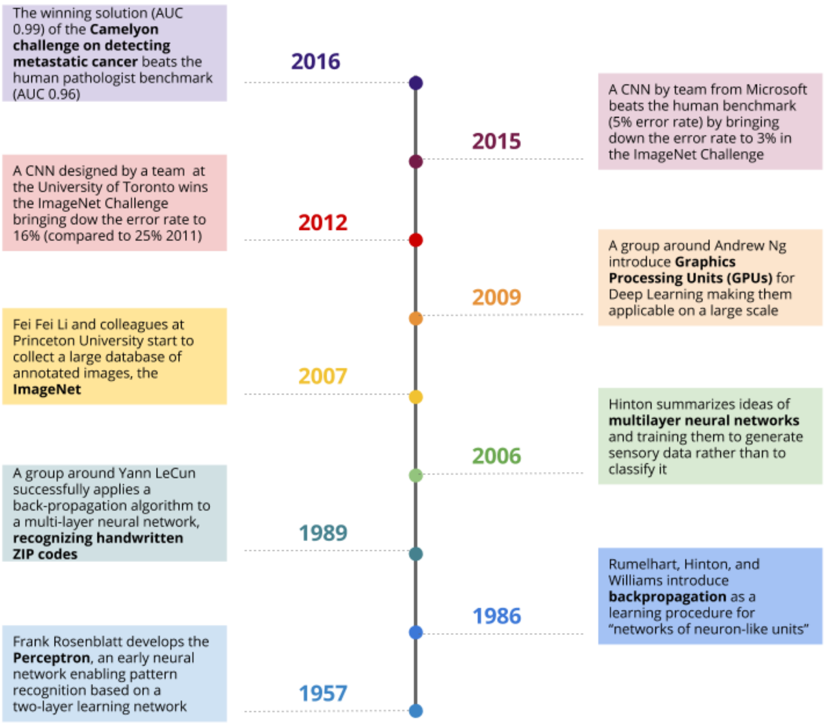


Figure 1: An AI timeline; **Source:** Lavenda, D./Marsden, P.

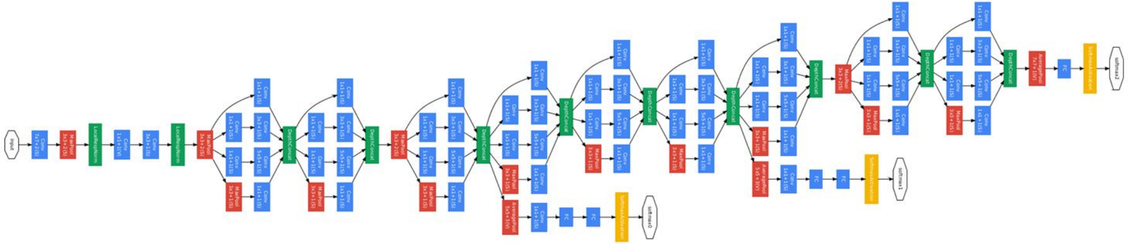
source dhl via @mikequindazzi

Deep Learning Timeline



AI winters are probably over

- We now have large, high-quality, labeled data sets
- GPUs and TPUs abound
  - Allows for deeper models and an increase in accuracy
- Improved functions needed for learning
  - ReLU
  - tanh
- Improved architectures
  - Resnets
  - Inception modules
- New regularization techniques
  - Dropout
  - Batch normalization
- Robust optimizers
- Software platforms
  - Tensorflow
  - Theano



How to stay current

- Advances in deep learning, and AI in general, are happening every day - it isn’t possible to keep track of everything, but below are some good sources to check out
- Read papers on arXiv
- Subscribe to Medium
- Google AI Blog
- Keras Blog
- OpenAI Blog
- Twitter
    - Follow deep learning gods like Ian Goodfellow, Yann Lecun, Fei-Fei Li, Francois Chollet and our own HSPH professor Andrew Beam
- Talking machines podcast

Takeaways

- Deep learning is real and probably here to stay
- Could potentially impact many fields → understand concepts so you have deep learning "insurance"
- Long history and connections to other models and fields
- Prereqs: Data (lots) + GPUs/TPUs (more = better)
- Deep learning models are like legos, but you need to know what blocks you have and how they fit together
- Need to have a sense of sensible default parameter values to get started
- "Babysitting" the learning process is a skill

Neural Nets

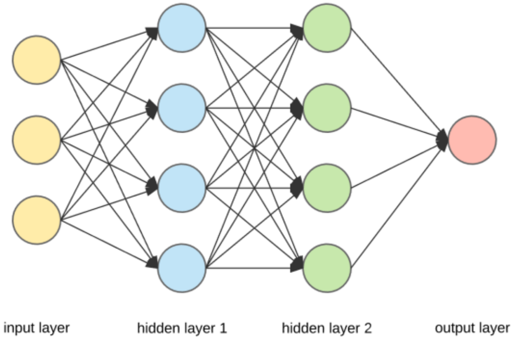
What is a neural net?

A neural net is composed of 3 things:

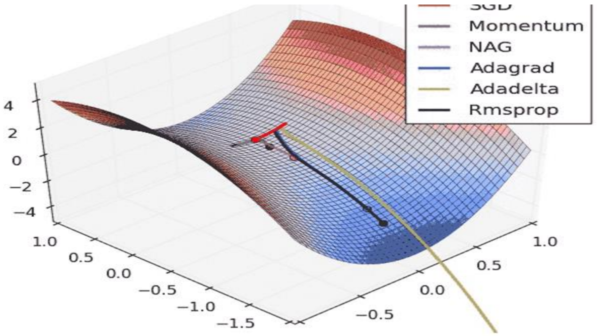
The network structure

The loss function

The optimizer



$$-y_i * \log(p_i) - (1 - y_i) * \log(1 - p_i)$$



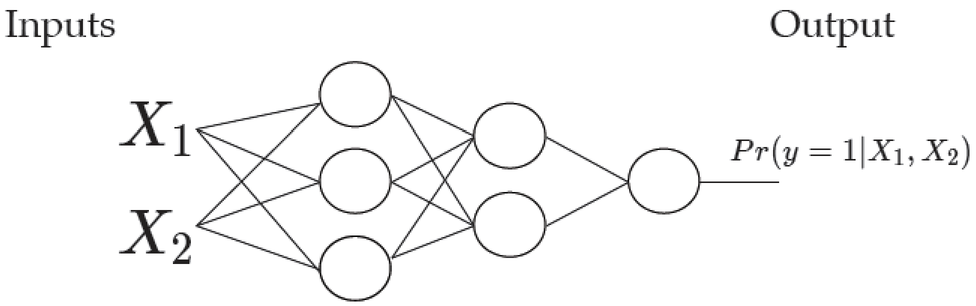
SGD optimization on saddle point

https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-95ae5d39529f

https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6

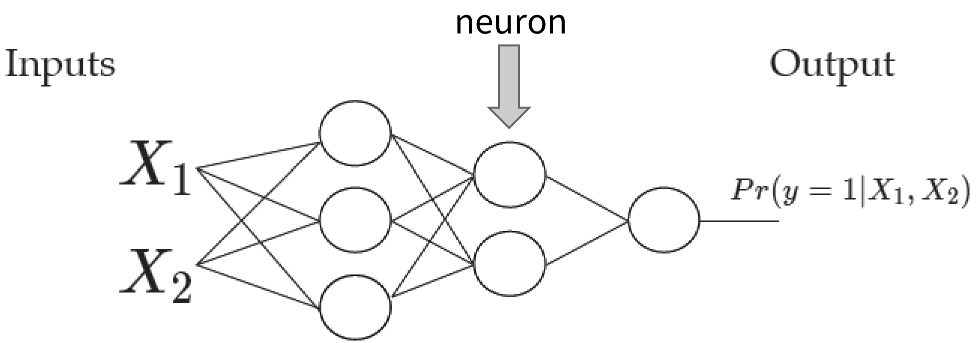
Neural Net Structure

- A neural net is a modular way to build a classifier



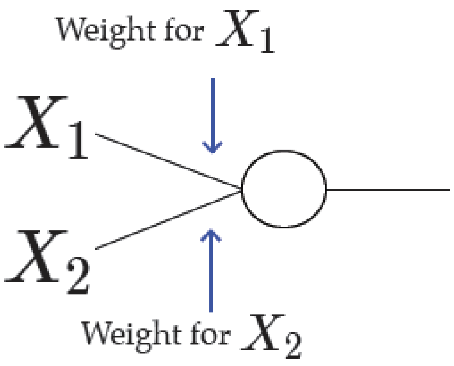
Neural Net Structure

- A neural net is a modular way to build a classifier
- The neuron is the basic functional unit in a neural network



Neurons

- A neuron does 2 things:



1) Weighted sum of inputs

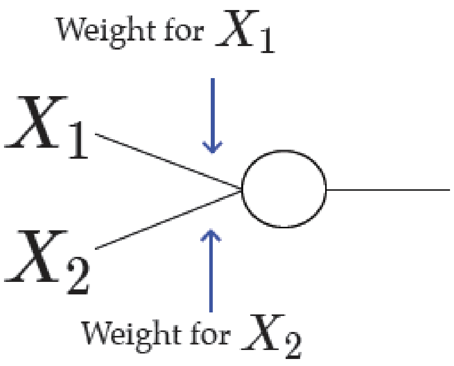
$$w_1 * X_1 + w_2 * X_2$$

2) Nonlinear transformation

$$\phi(w_1 * X_1 + w_2 * X_2)$$

Neurons

- A neuron does 2 things:



1) Weighted sum of inputs

$$w_1 * X_1 + w_2 * X_2$$

2) Nonlinear transformation

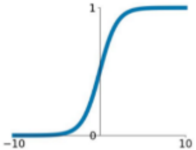
$$\phi(w_1 * X_1 + w_2 * X_2)$$

Activation function - a non-linear transformation

Activation Functions

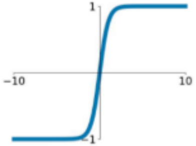
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



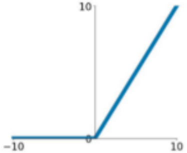
tanh

$\tanh(x)$



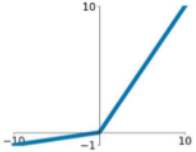
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

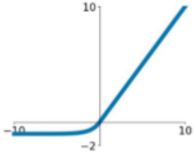


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

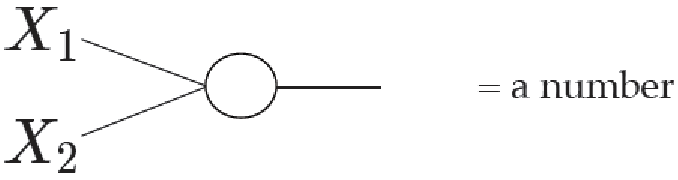
ELU

$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$



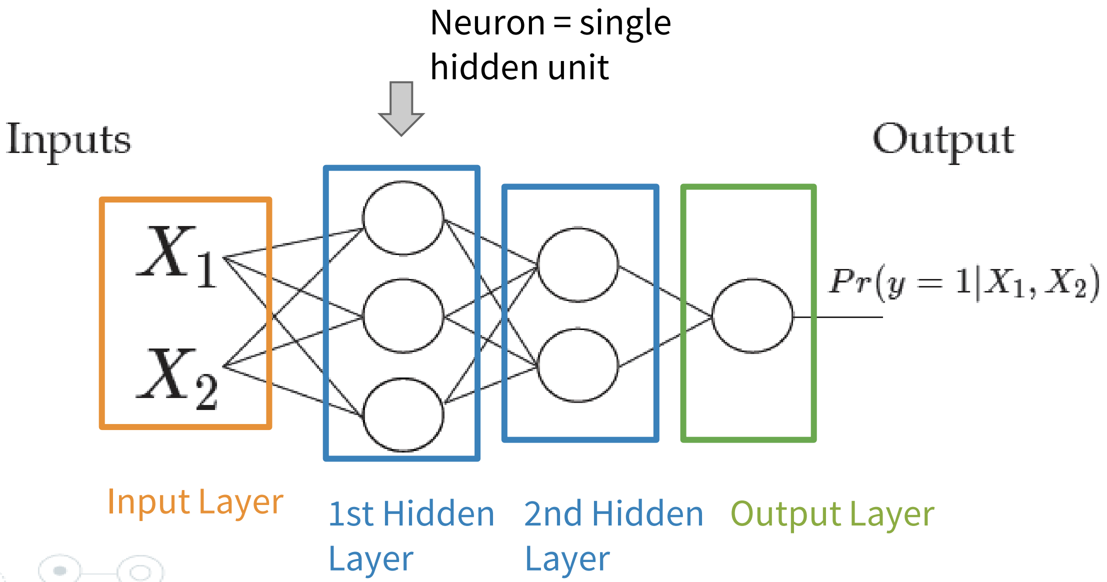
Neuron

- A neuron produces a single number that is a nonlinear transformation of its input connections



Neural Net Structure

- Neural nets are organized into **layers**

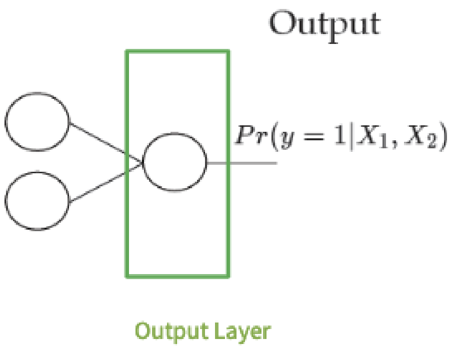


This is where we left it during the first lecture

Loss Functions

- We need a way to measure how well the network is performing
  - Is it making good predictions?

- A **loss function** is a function that returns a single number which indicates how closely a prediction matches the ground truth, i.e. the actual label
  - We want to minimize the loss to achieve more accurate predictions
  - Also known as the objective function, cost function, loss, etc.



Loss Functions

One of the simplest loss functions is **binary cross-entropy** which is used for binary classification

y_i is the true label

$$p_i = \mathrm{P}(y_i = 1 | X_1, X_2)$$

$$l(y_i, p_i) = -y_i * log(p_i) - (1 - y_i) * log(1 - p_i)$$

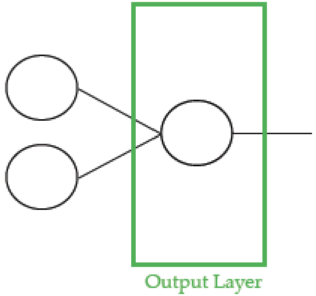
y	p	Loss
0	0.1	0.1
0	0.9	2.3
1	0.1	2.3
1	0.9	0.1

Output Layer and Loss

- The output layer needs to “match” the loss function
- Correct shape
- Correct scale
- For binary cross-entropy, the network needs to produce a single probability:

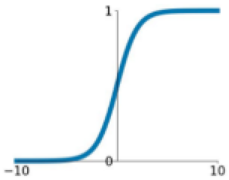
$$p_i = \mathrm{P}(y_i = 1 | X_1, X_2)$$

- One unit in the output layer represents this probability
- Activation function must “squash” the output to be between 0 and 1



Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



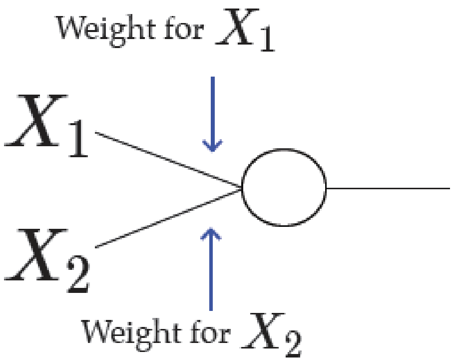
Output Layer and Loss

- We can change the output layer and loss to model many different kinds of data

Task	Last-layer activation	Loss function
Binary classification	sigmoid	Binary cross-entropy
Multiclass, single-label classification	softmax	Categorical cross-entropy
Multiclass, multilabel classification	sigmoid	Binary cross-entropy
Regression to arbitrary values	None	Mean square error (MSE)
Regression to values between 0 and 1	sigmoid	MSE or binary cross-entropy

The Optimizer

Remember this?



1) Weighted sum of inputs

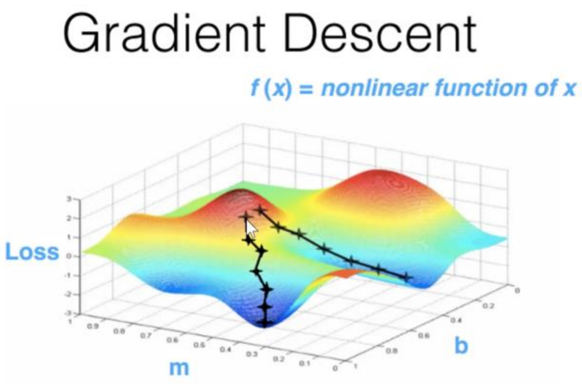
$$w_1 * X_1 + w_2 * X_2$$

2) Nonlinear transformation

$$\phi(w_1 * X_1 + w_2 * X_2)$$

We have specified the network and the loss function, but what about values for the weights? We want weights that minimize the loss function - how do we calculate them?

The Optimizer



Q: How do we minimize the loss function?

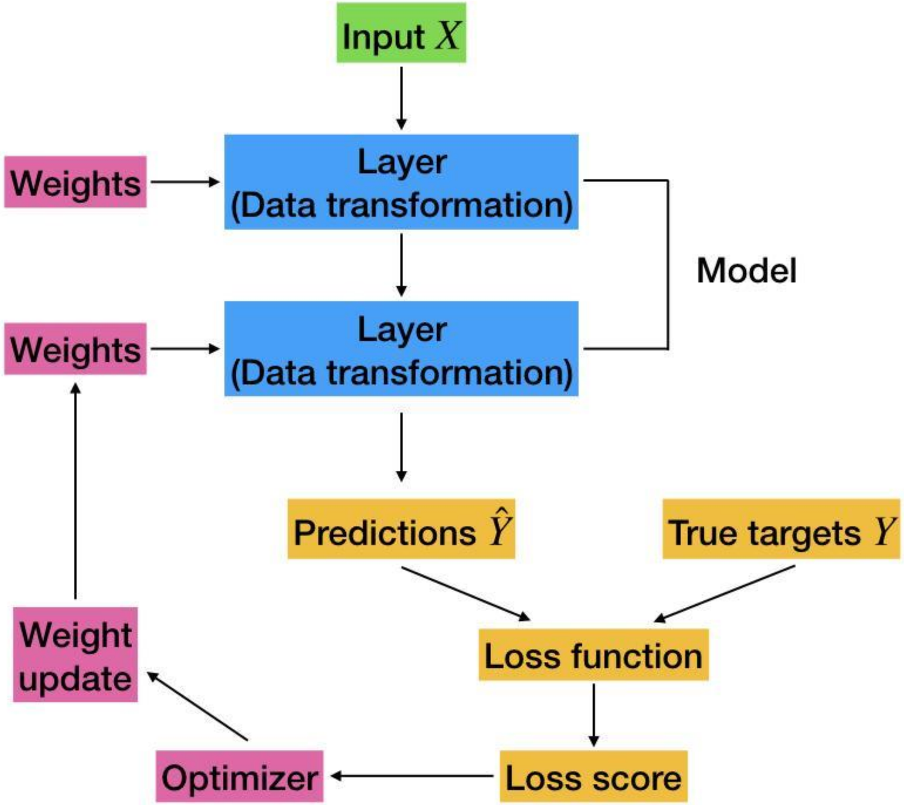
A: Stochastic Gradient Descent (SGD)

1. Give weights random initial values
2. Evaluate the partial derivative of the loss function with respect to each weight at the current weight value on a mini-batch
3. Take a step in the direction opposite the gradient
4. Repeat

The Optimizer

- Many variations on the basic idea of SGD are available
  - Rmsprop
  - Adagrad
  - Adadelta
  - Momentum
  - NAG
  - etc.

Workflow



Input X
Weights
Layer
(Data transformation)
Model
Weights
Layer
(Data transformation)
Predictions $\hat{Y}$
True targets Y
Weight
update
Loss function
Optimizer
Loss score