Learning Deep Learning with PyTorch

(1) Introduction

Qiyang Hu
UCLA IDRE
February 19, 2020

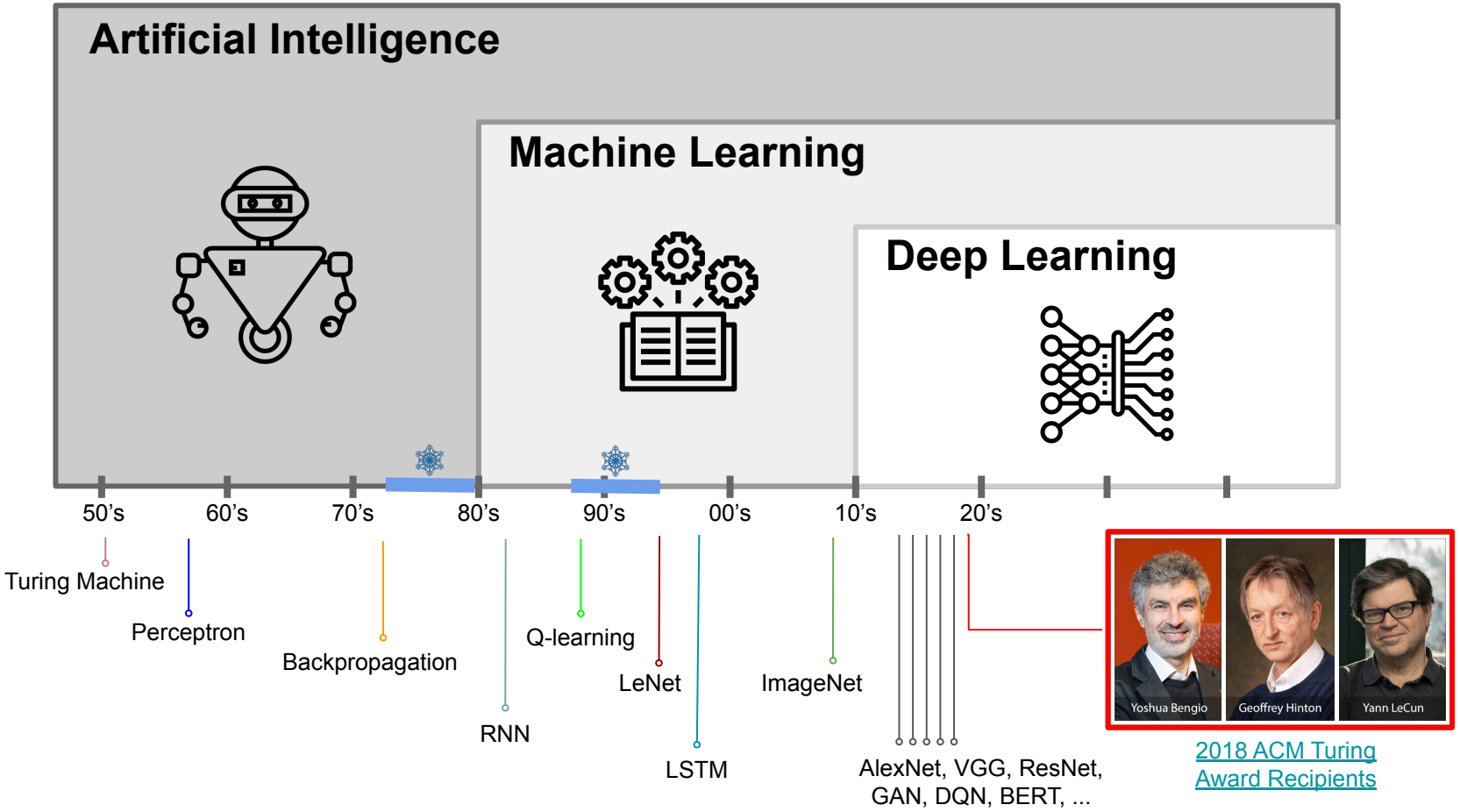
About the series

- Not a comprehensive course!
  - Scratching the surface of deep learning
  - High-level descriptive review
  - Mastering the content needs your own effort after each class

- Workshop plan in this quarter
  - Introduction (Feb 19, 2020)
  - Learning Mechanics of Deep Learning (Feb 21, 2020)
  - Knowing PyTorch (Feb 26, 2020)
  - Convolutional Neural Networks (Feb 28, 2020)
  - Practical techniques in Deep Learning (Mar 4, 2020)

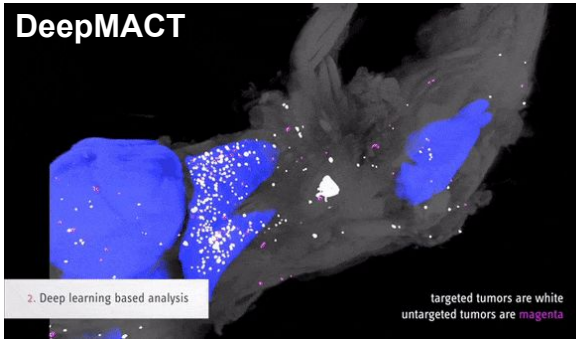
Slides only

w/ Pytorch example


Artificial Intelligence
Machine Learning
Deep Learning
50's
60's
70's
80's
90's
00's
10's
20's
Turing Machine
Perceptron
Backpropagation
RNN
Q-learning
LeNet
LSTM
ImageNet
AlexNet, VGG, ResNet,
GAN, DQN, BERT, ...
Yoshua Bengio
Geoffrey Hinton
Yann LeCun
2018 ACM Turing
Award Recipients

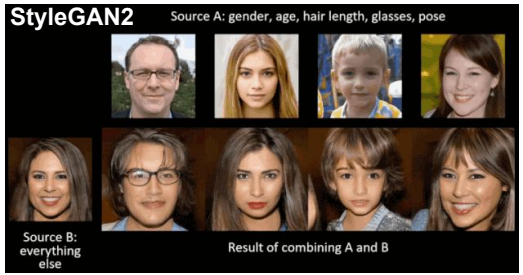
Amazing Deep Learning Achievements in 2019

Detectron2



Equation-solving AI

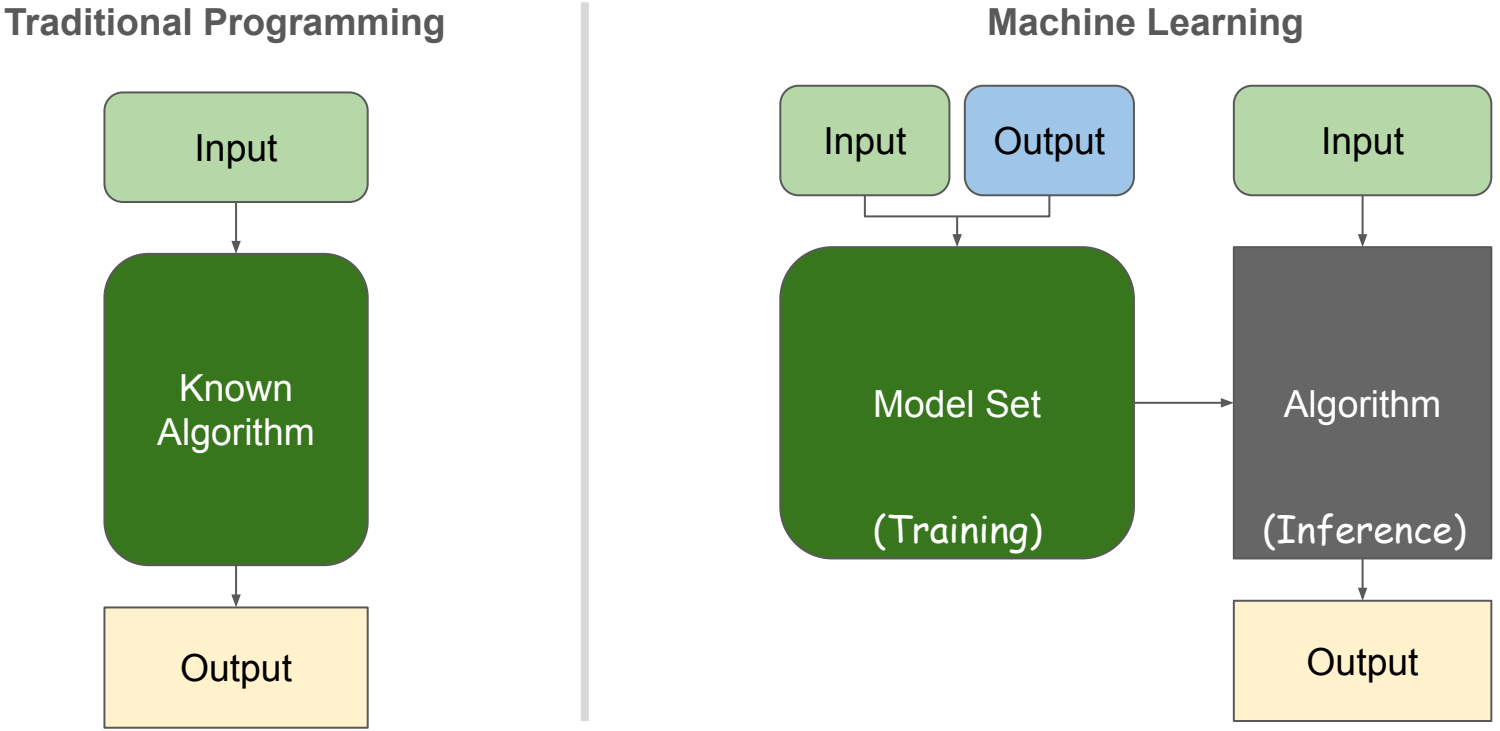
EQUATION	SOLUTION
$y' = \frac{16x^3 - 42x^2 + 2x}{(-16x^8 + 112x^7 - 204x^6 + 28x^5 - x^4 + 1)^{1/2}}$	$y = \sin^{-1}(4x^4 - 14x^3 + x^2)$
$3xy\cos(x) - \sqrt{9x^2\sin(x)^2 + 1y'} + 3y\sin(x) = 0$	$y = c\exp(\sinh^{-1}(3x\sin(x)))$
$4x^4yy'' - 8x^4y'^2 - 8x^3yy' - 3x^3y'' - 8x^2y^2 - 6x^2y' - 3x^2y'' - 9xy' - 3y = 0$	$y = \frac{c_1 + 3x + 3\log(x)}{x(c_2 + 4x)}$



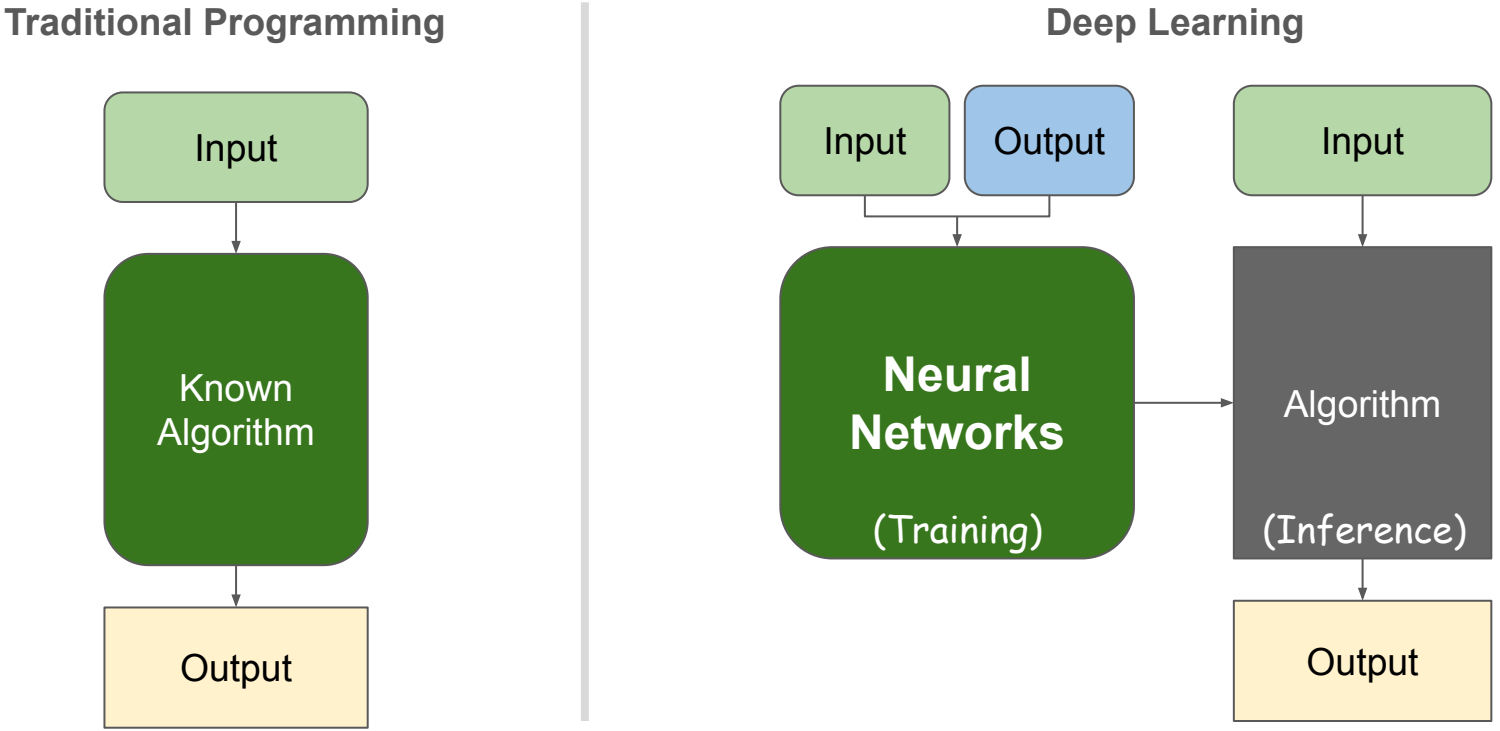
GPT-2

Qiyang Hu

What is Machine Learning?

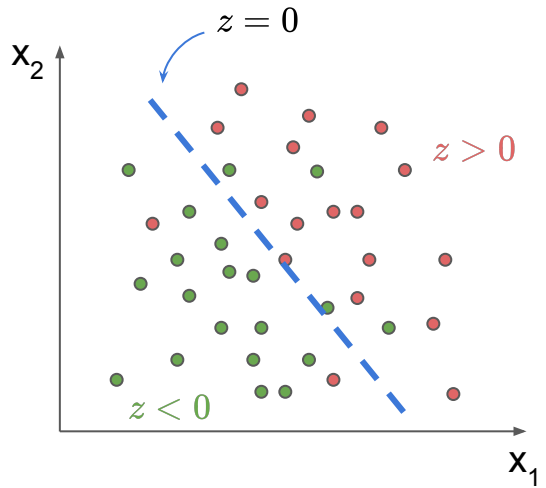


What is Deep Learning?

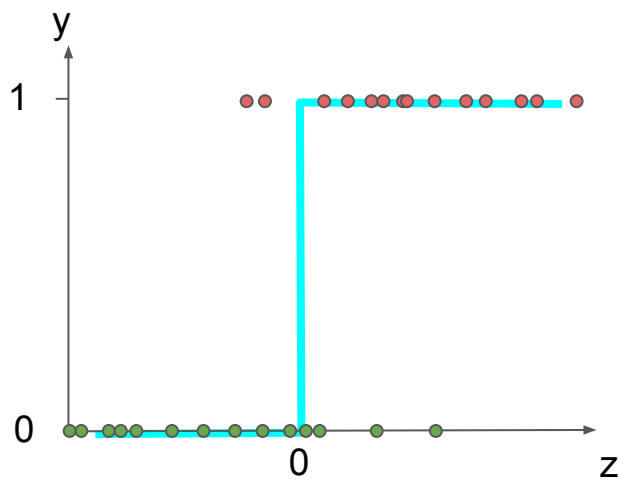


What is Neural Network?

- Recap for simple linear classification problem

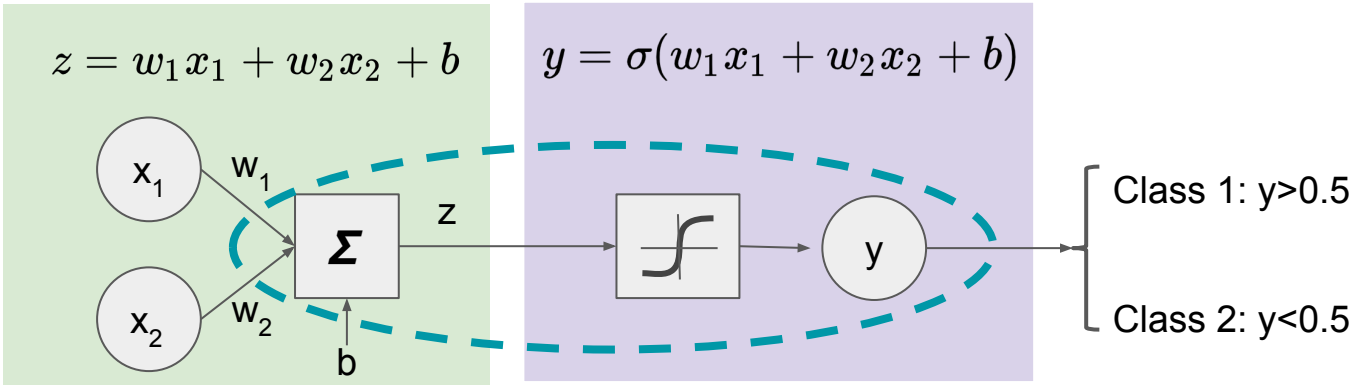


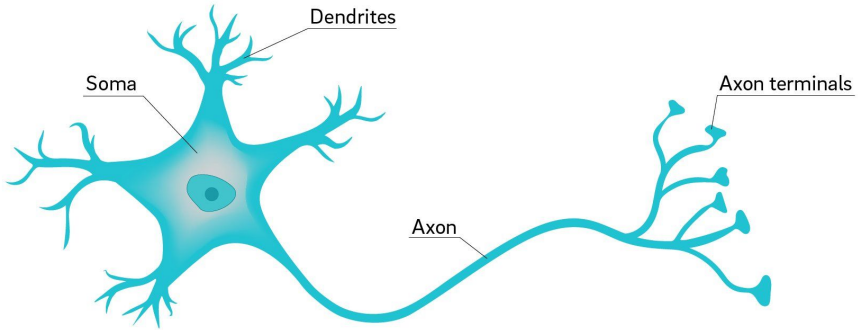
$$z = w_1 x_1 + w_2 x_2 + b$$



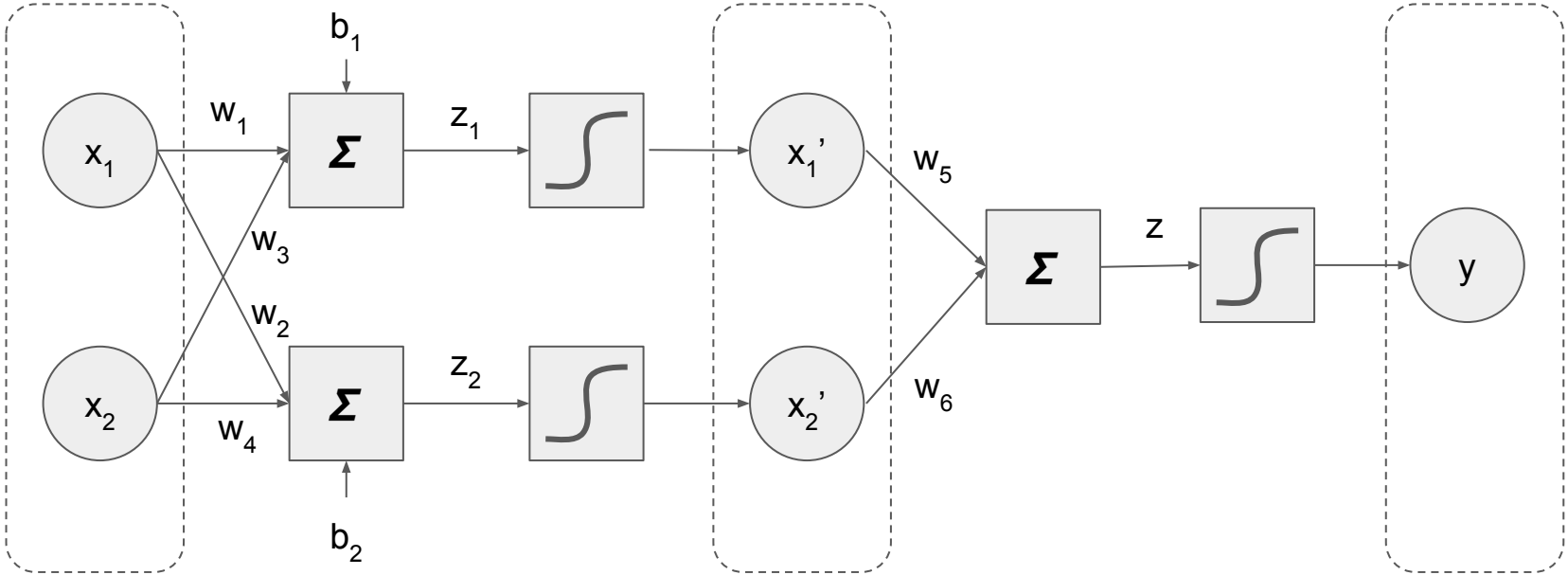
$$y = \sigma(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{if } z < 0 \end{cases}$$

A linear classifier ~ one artificial neuron





(Deep) Neural Networks ~ piling/stacking logistic-regression classifiers

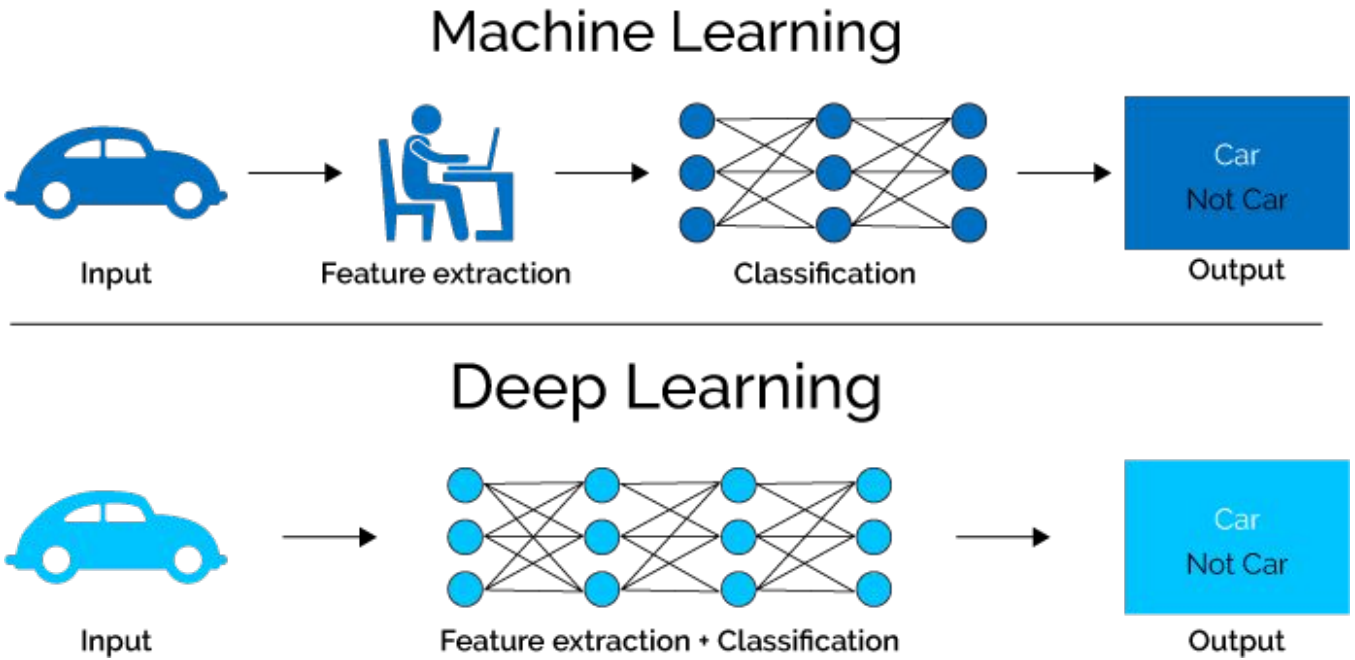


How deep a deep learning network can be?

- LeNet-5 (1998) 

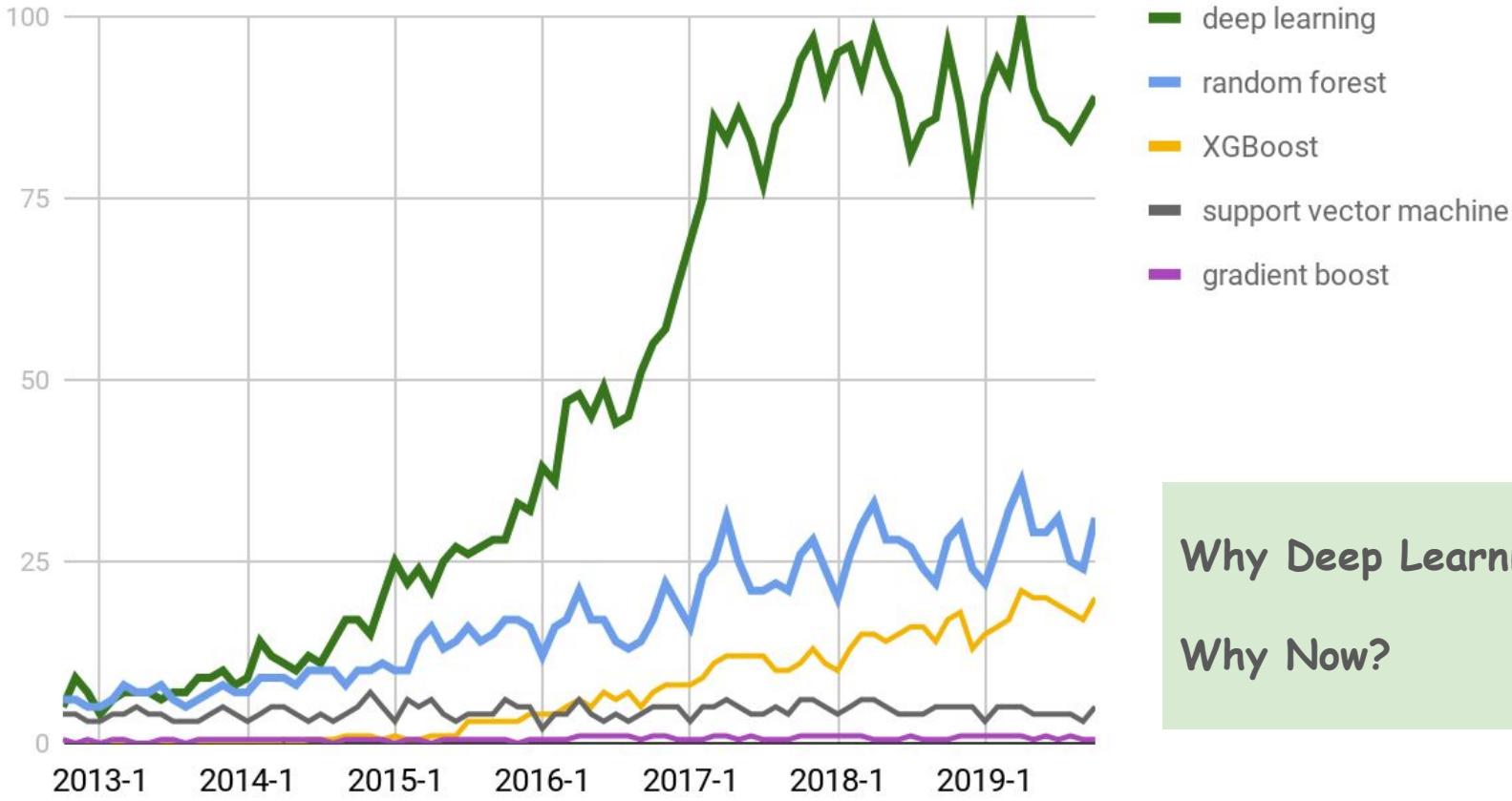
Year	CNN	Developed by	Place	Top-5 error rate	No. of parameters
1998	LeNet(8)	Yann LeCun et al			60 thousand
2012	AlexNet(7)	Alex Krizhevsky, Geoffrey Hinton, Ilya Sutskever	1st	15.3%	60 million
2013	ZFNet()	Matthew Zeiler and Rob Fergus	1st	14.8%	
2014	GoogLeNet(19)	Google	1st	6.67%	4 million
2014	VGG Net(16)	Simonyan, Zisserman	2nd	7.3%	138 million
2015	ResNet(152)	Kaiming He	1st	3.6%	

Machine Learning vs. Deep Learning



Source: https://www.xenonstack.com/blog/log-analytics-deep-machine-learning/

Interest over time from Google Trends



Why Deep Learning?

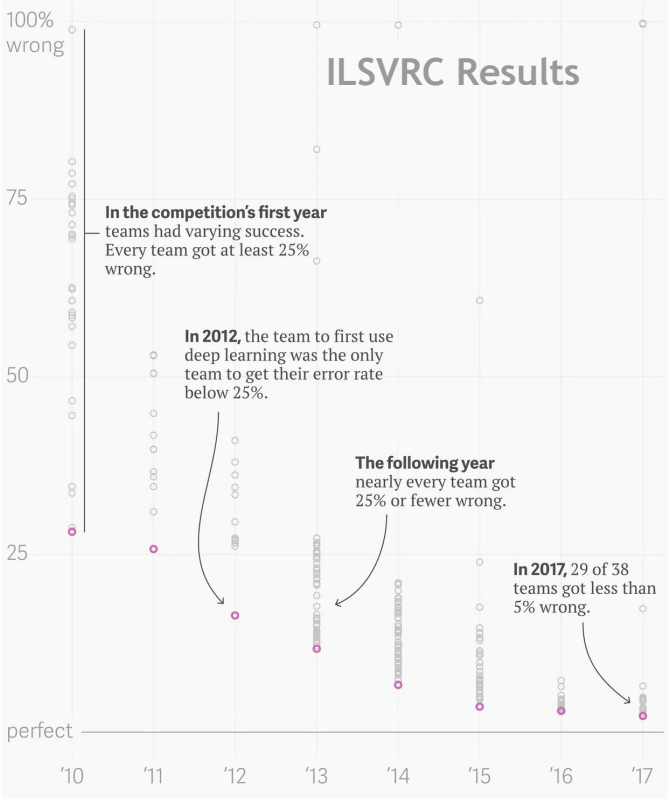
Why Now?

Driving Forces in Deep Learning (1): *Algorithms*

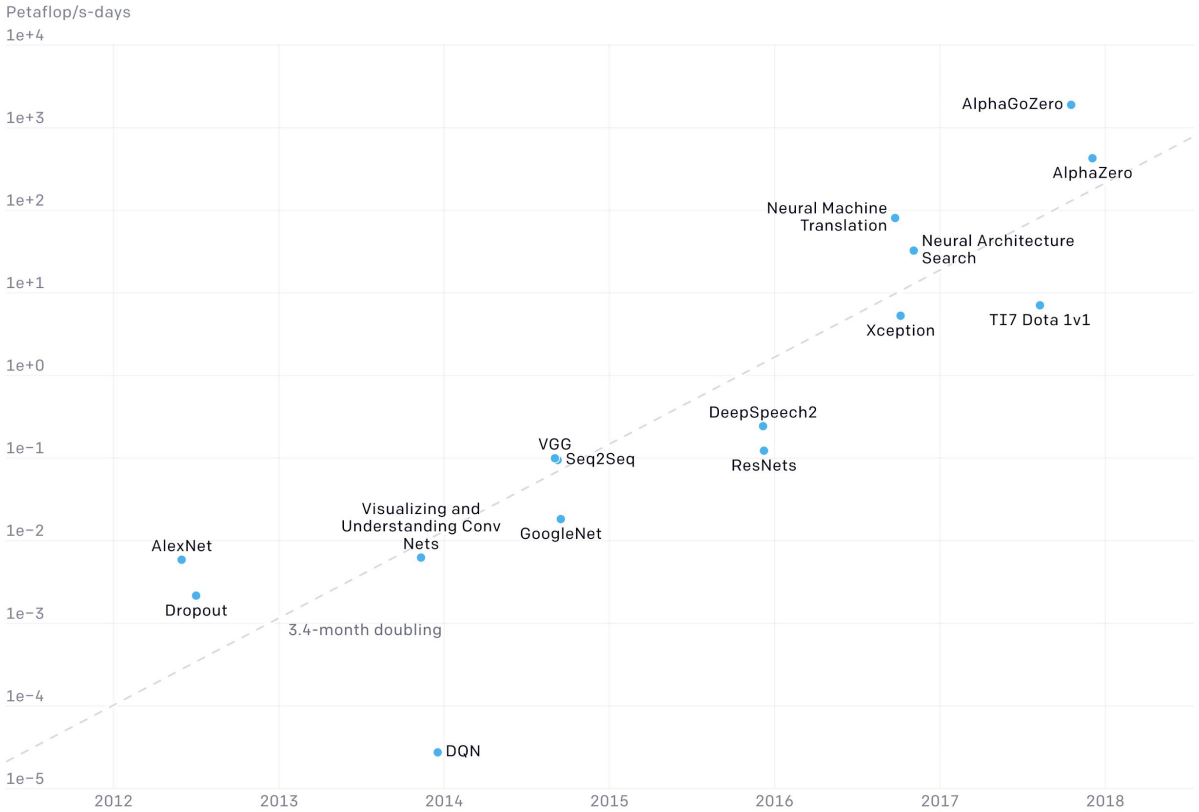
- Key Issues in Deep Learning:
  - Optimization for models with huge number of parameters
  - Gradient Propagation through stacks of layers
    - Gradient vanishing and exploding
- Algorithmic improvements in 2009-2010:
  - Better activation functions
  - Better weight-initialization schemes
  - Better optimization schemes
- Advanced techniques in 2014-2018:
  - Batch normalization/Drop-out
  - Residual/skip connections
  - Depth-wise separable convolutions

Driving Forces in Deep Learning (2): *Data*

- ImageNet (a large dataset of annotated photographs built on 2009)
  - 14 million+ images
  - 21,000 groups or classes
  - ILSVRC competition (1.2 million image, 1000 classes)
- Kaggle (Founded in 2010, acquired by Google in 2017)
  - 1,000,000+ registered users in 194 countries in 2017
  - Hosts 19K+ of datasets and 200K+ code snippets
  - Famous for the high-rewards competitions
- Datasets from the rise of internet
  - User-generated image tags on Flickr
  - Video dataset/tags on Youtube
  - Data from Wikipedia for NLP

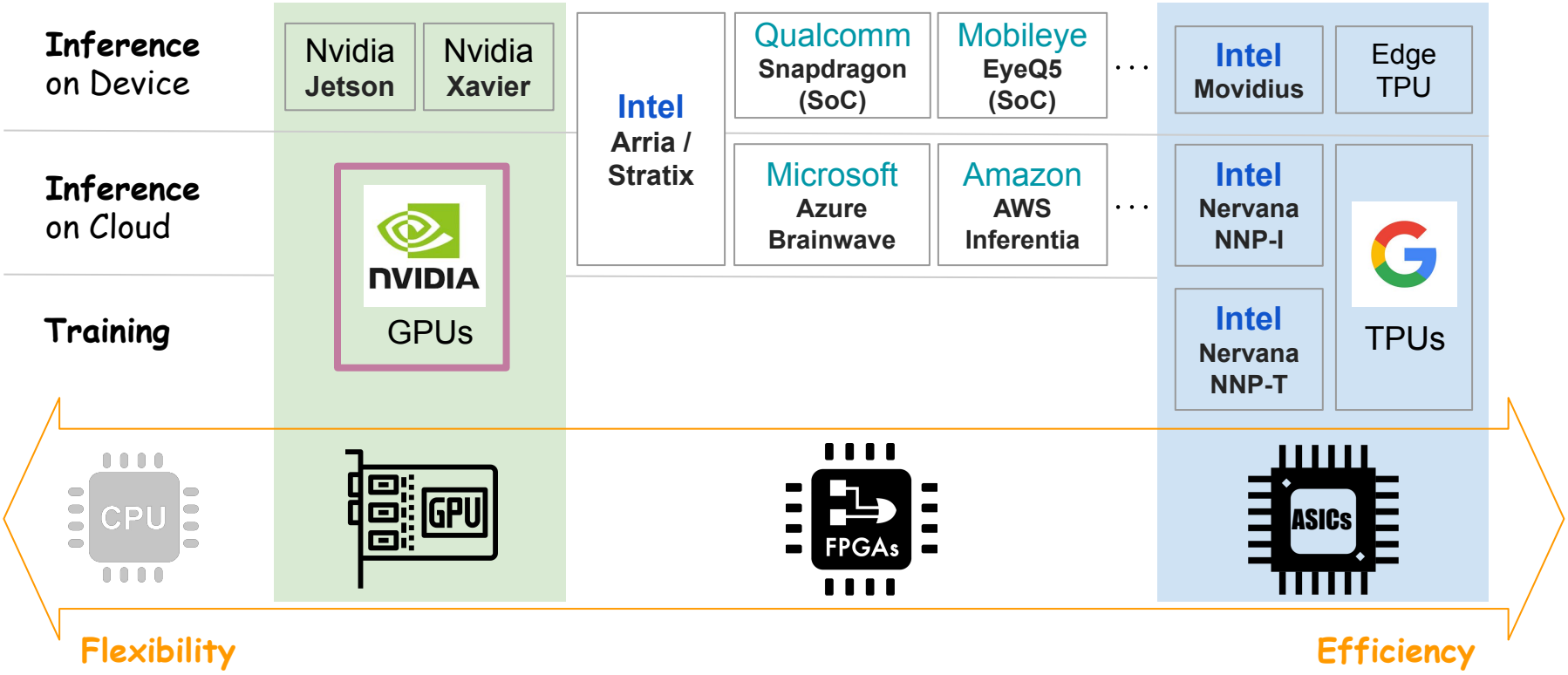


AI compute amount increases 10 times per year!



Source: [OpenAI Report]

Driving Forces in Deep Learning (3): *Hardware*



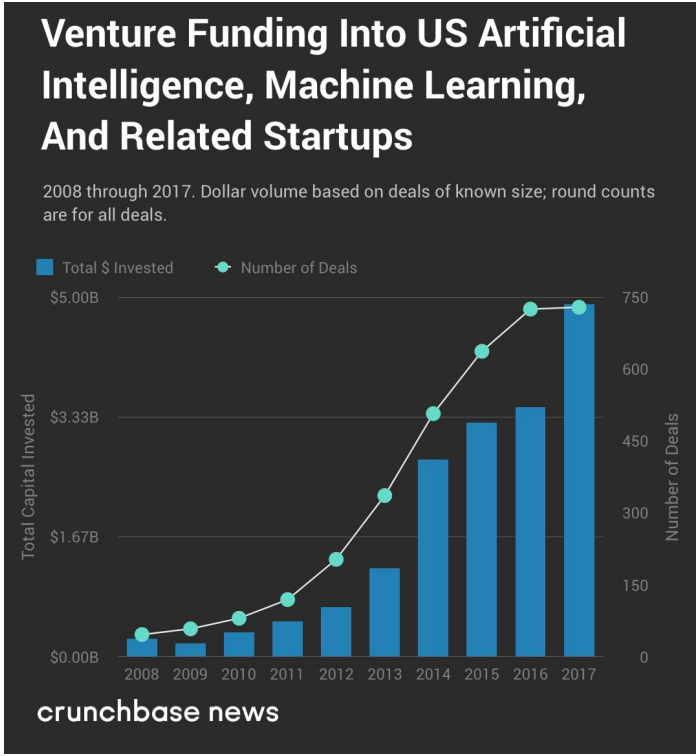
Free GPU Computation Resources

- Cloud-based resources (Google Colaboratory, Kaggle, Paperspace's Gradient)
  - A free Jupyter notebook env that requires no setup and runs entirely in the cloud.
  - Google Drive ➛ New ➛ More ➛ Google Colaboratory
  - Kaggle.com ➛ Log in ➛ Kernel ➛ New Kernel
- Hoffman2 (GPU resources)
  - Work under python shell in terminal:
    - qrsh -l gpu,P4
      module load python/anaconda3
      conda activate pytorch-1.3.1-gpu
  - Work in Jupyter notebook session:
    - For the very first time, add the virtual env to kernel in the ***above*** qrsh session:
      - python -m ipykernel install --user --name=pytorch-1.3.1-gpu
    - Using h2jupynb:
      - ./h2jupynb -v anaconda3 -g yes -c P4 -l 10.0

	Colab	Kaggle	Hoffman2
CPU Type	Intel Xeon 2.30GHz	Intel Xeon 2.30GHz	Intel Xeon 2.80GHz
Slots/Threads available	1 core / 2 threads	1 core / 2 threads	8 cores / no hyper-threads
RAM available	12 GB	18 GB	24 GB
Disk available	311 GB	626 GB	1 TB
GPU Type	Tesla T4 (2018)	Tesla P100 (2018)	Tesla P4 (2016)
GPU SP Floating-Point Perf	8.1 TFLOPs	10.6 TFLOPs	5.5 TFLOPs
GPU Memory	16 GB	16 GB	8 GB
Active Time Limit	8 hours	6 hours	24 hours

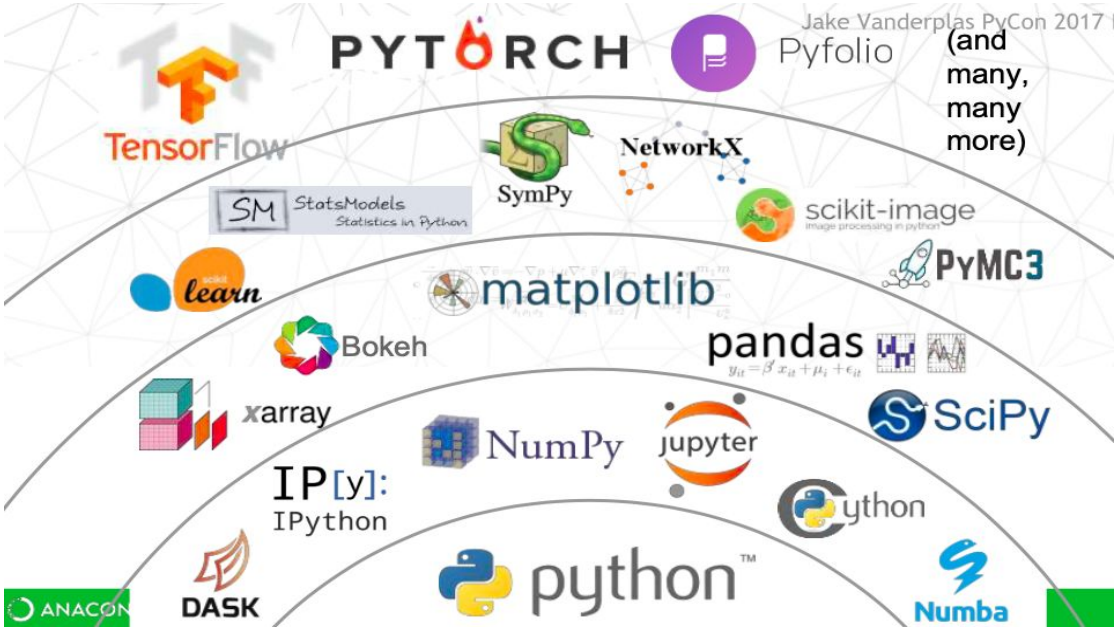
Driving Forces in Deep Learning (4): *Investment*

- Venture Capital Investment soars
  - Source
  - 20x increase in 8 years
  - Most for deep-learning
- AI acquisitions
  - Google: $500M for DeepMind (2013)
  - Intel: $400M for Nervana Systems (2016)
  - Tons of M&As undisclosed
- Demand drives supply
  - 100x more people working on deep learning



Driving Forces in Deep Learning (5): *Toolsets*

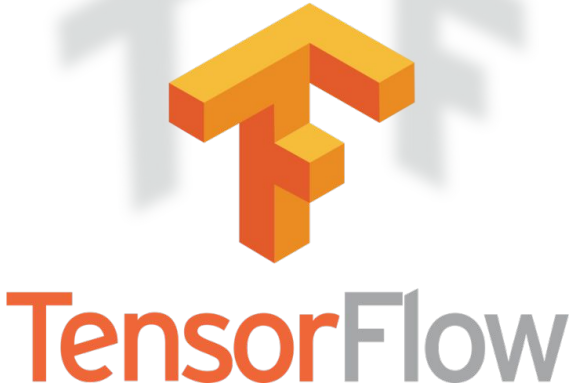
- From C++/Cuda to scripting languages (Python, R)



Deep Learning Framework Battles

Finals?

vs.



Key Terminology in Machine/Deep Learning

- Datasets:
  - Label: a desired output (e.g. house price)
  - Feature: a known input (e.g. address, condition, household income, etc)

- Model: relationship between input & output
  - Parameter: to be learned from data, e.g. weight, coefficients
    - Weight: a coefficient for a feature in linear model
    - Bias: an intercept or offset from an origin
  - Hyperparameter: often set by heuristics, e.g. learning rate, depth of trees, batch, epoch.
  - Batch: a subset from the division of training datasets
  - Epoch: all data in training sets has had an opportunity to update the internal model parameters

Complete Glossary

A lot of “Learning”s to learn

- Supervised Learning (data with labels)
  - Regression
  - Classification
- Unsupervised Learning (data without labels) (Auto Encoders)
- Semi-supervised Learning (data with partial labels)
- Reinforcement Learning (reward rules to get data) (PPO, Deep Q-learning)
- Inverse reinforcement learning (no rules & no labels)
- Transfer Learning (data with unrelated labels)

  (zero-shot learning, one-shot learning, few-shot learning, etc.)

⇒ Continuous learning

⇒ Meta Learning (MAML, LSTM)

Don’t forget to

- Sign in your info to the class
  - To get the email notifications

- Contact me for questions or discussions
  - huqy@idre.ucla.edu
  - Office: Math Sci #3330
  - Phone: 310-825-2011