Machine Learning
CSCE 5215

Deep Learning

Introduction to Deep learning

- Motivation and background
  - ubiquity and engineering feats that are possible now
  - learning → semi-supervised learning
- Neural nets
  - perceptron - ability and limitations
  - multi-layer networks and backpropagation
  - convolutional neural networks
- Recent advances enabling deep learning’s rise
  - parallelization, big data, and algorithmic advances
- Similarity to neural processing in the brain

Why deep learning should matter to you…

- Deep learning is already all around you, powering the speech and image recognition algorithms you use every day (google now, siri, cortana)
- Unlike previous neural net excitement, this is not about what CAN be learned, but rather what is BEST at learning, and “brain-like” learning is winning.
- This combination of automated feature extraction and exploitation is revolutionizing science.
    - instead of talking about features used for task learning, we can just learn directly from the data
    - results are “I used this algorithm and this data” instead of “this equation” or “this feature”
- In another 10-15 years, similar automated structural learning will be as ubiquitous as regression is today
    - e.g. in many circles, classification by an individual, hand-selected feature will be frowned upon

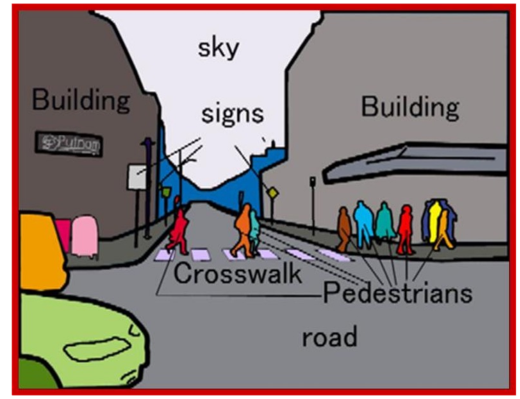
History of Deep Learning Ideas and Milestones

- 1943: Neural Networks
- 1957-62: Perceptron
- 1970-86: Backpropagation, RBM, RNN
- 1979-98: CNN, MNIST, LSTM, Bidirectional RNN
- 2006: “Deep Learning”, DBN
- 2009: ImageNet + AlexNet
- 2014: GANs
- 2016-17: AlphaGo, AlphaZero
- 2017: 2017-19: Transformers

The challenging problems..

These are hard problems for machines

- Speech recognition
- Object identification
- Natural language processing



But people have relatively little trouble, so we know they are solvable.

And it turns out, the way our best current approach works (deep learning) is analogous to the way the brain processes information, and we'll take a moment at the end to delve into this comparison

Introduction

- Learning: real world and how it relates to classification and regression
- Supervised learning
- Unsupervised learning
- Semi-supervised advantages

Note, deep learning networks can be applied to supervised learning problems, unsupervised learning problems, or even reinforcement learning. However their power comes from the combination of feature extraction and exploration seen in semi-supervised learning domains

What is Learning?

Learning is about increasing your ability to predict the future and enact change accordingly. Let's focus on the prediction part.

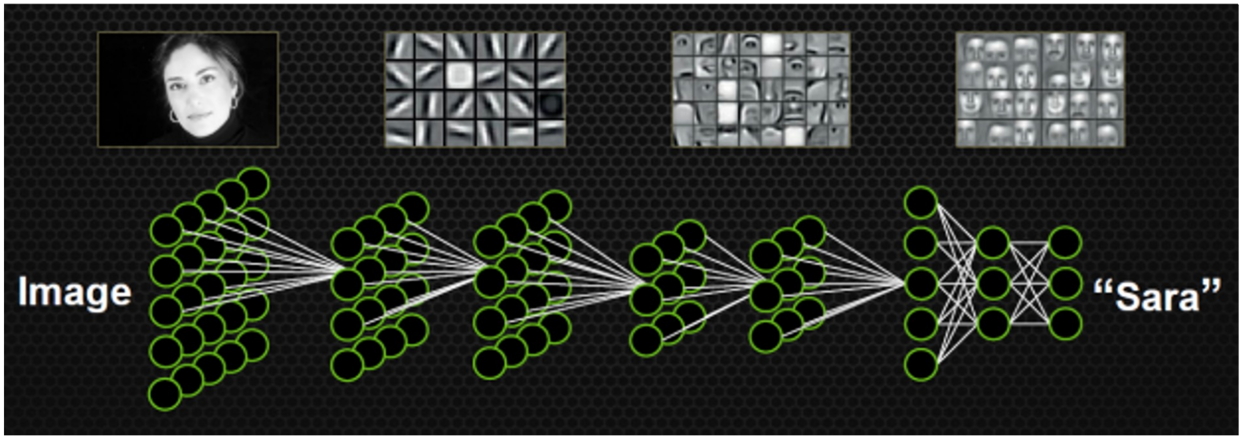
Generally, we are trying to predict one of two things

- A number → regression
  - Tomorrow's temperature
  - Stock prices ($$$)
  - How far the basketball hoop is
- A class or condition → classification
  - audio speech → text (was that "bat" or "hat"?)
  - detecting predators, prey
  - a deer vs a sign post while driving

Examples of Machine Learning models

- Supervised, regression
  - linear regression, regularized with lasso and ridge
  - quadratic, cubic splines...
  - Neural Networks
- Supervised, classification
  - Support Vector Machines
  - Logistic regression
  - Naive Bayes Classification
  - Decision Trees
- Unsupervised
  - K-means clustering

Deep learning



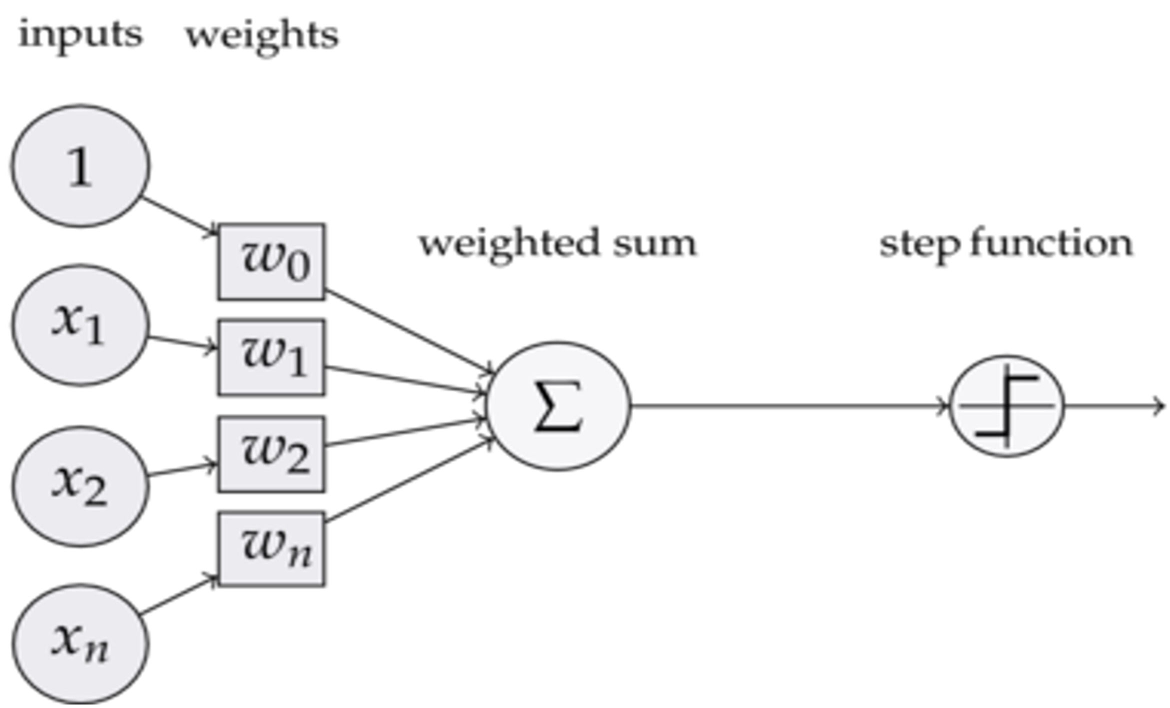
"A branch of machine learning that attempts to model high-level abstractions in data by using multiple processing layers ... composed of multiple non-linear transformations" - wikipedia

Typically, these "high-level abstractions" are used for classification

Let's build up to what this means, by starting with one layer...

Perceptron

- The simplest neural network learning strategy
- A one-layer model with output neurons created by thresholding a linear summation of inputs



Perceptron training rule

For each example j in our training set D, perform the following steps over the input $\mathbf{x}_j$ and desired output d_j:

a. Calculate the actual output:

$$y_j(t) = f[\mathbf{w}(t) \cdot \mathbf{x}_j] = f[w_0(t) + w_1(t)x_{j,1} + w_2(t)x_{j,2} + \cdots + w_n(t)x_{j,n}]$$

b. Update the weights:

$$w_i(t+1) = w_i(t) + \alpha(d_j - y_j(t))x_{j,i}, \text{ for all feature } 0 \le i \le n.$$

Note the learning rate. If too high, the weights overfit toward the most recent output. If too low, convergence is slow.

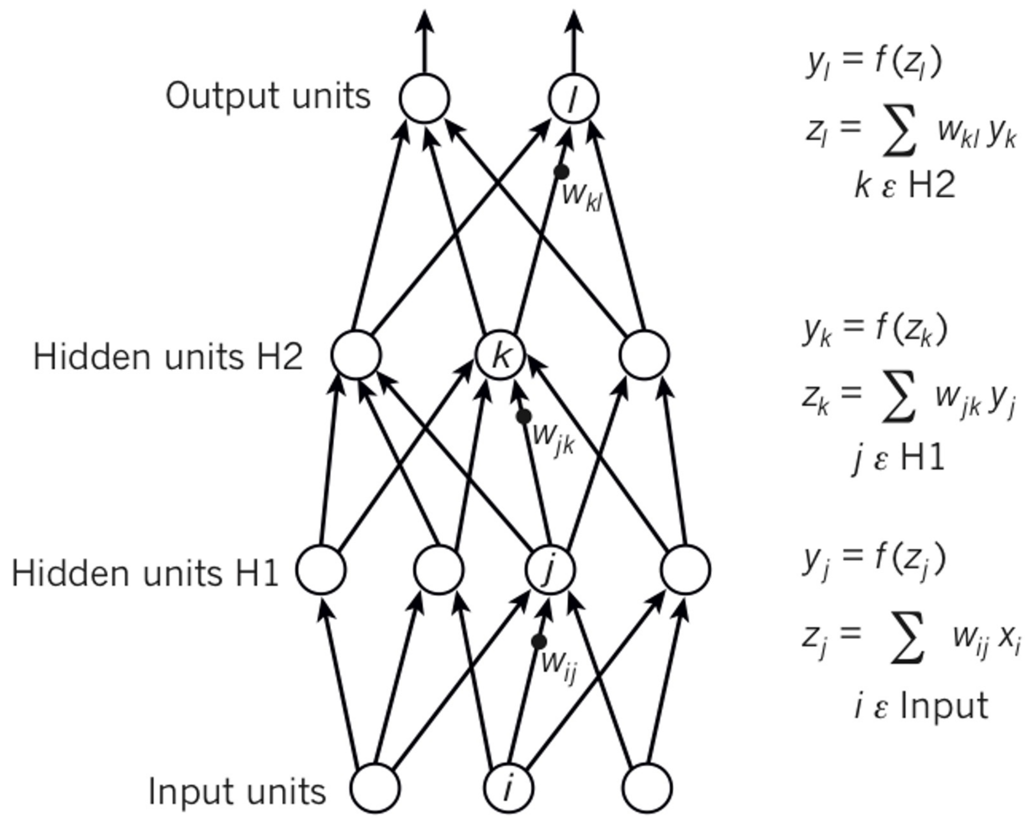
But perceptrons are limited to linear decision boundaries… multiple layers are needed for more complex learning

Multilayer Network

For each layer:

- **linear summation** of inputs from previous layer
  - learned weights
  - may be locally constrained
- **nonlinear transformation** of sum
  - often a threshold

But how are the weights calculated without an explicit hidden-layer training signal?



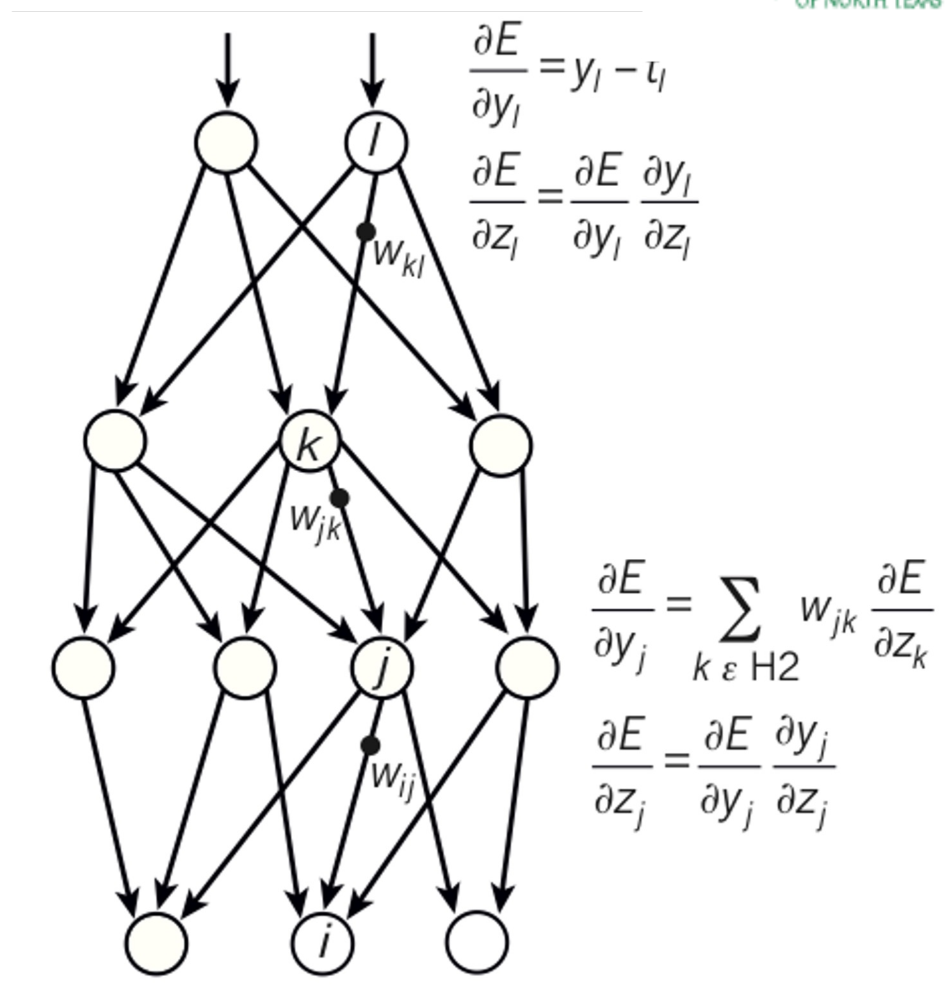
Back propagation

Adjust with weights to minimize the error.

Propagate adjustments backward from the output layer

Still, conceptually similar to:

$w_{new} = w_{old} + \alpha\ (w_{desired} - w_{old})$



Convolutional neural networks

CNN's take advantage of common problem characteristics - location invariance of features, and moderate invariance of location in conjunctions of features.

- Convolution: Linear filters are applied over the entire input space
  - this is because the same features may be present anywhere in the image
  - convolution is fast, generalized over location well, and is memory efficient compared to training separate networks for each location.
- Non-linear thresholding: standard Neural Networks practice
- (max) Pooling: over a small area, the outputs of multiple neurons are combined (by the max operator)
  - introduces invariance over position for later layers
- Normalization: Ensuring activity values stay in reasonable ranges

Convolutional neural networks image recognition

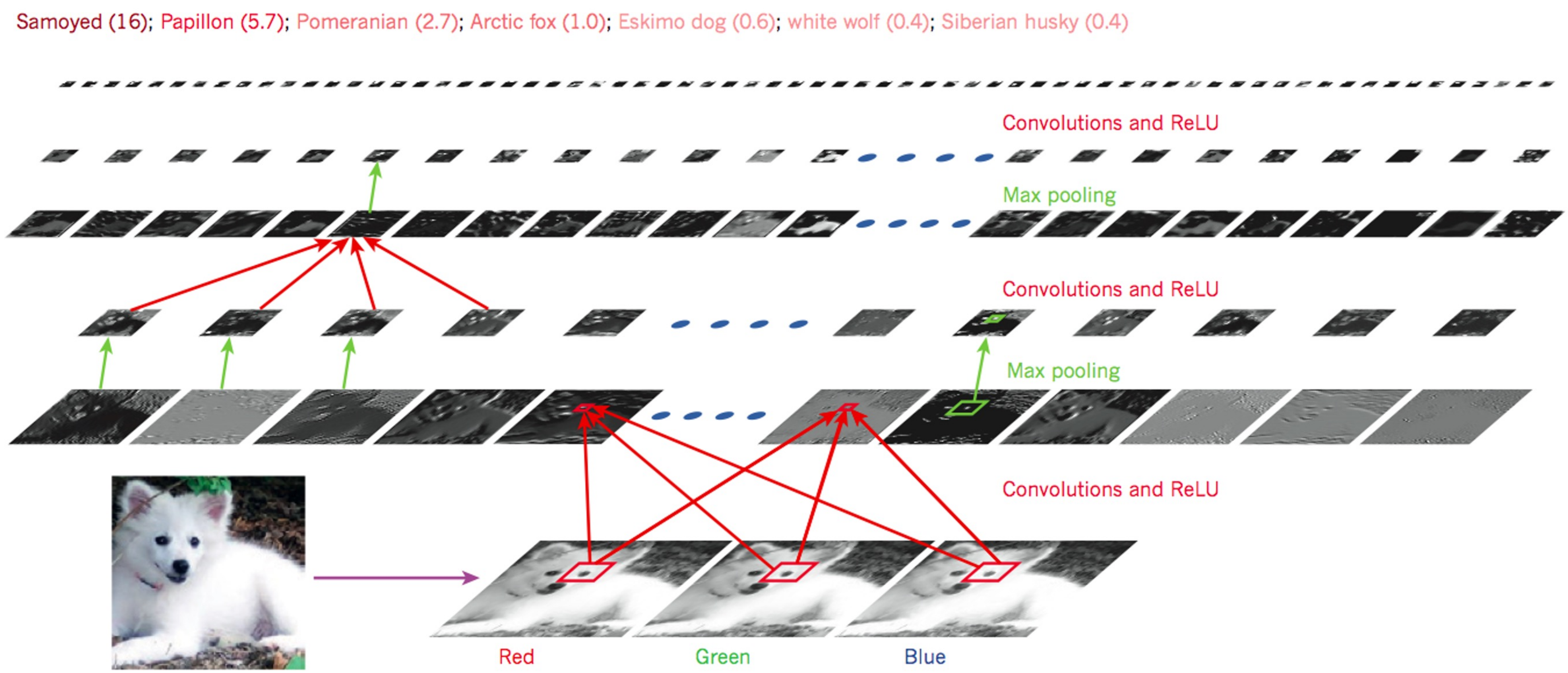


Figure 2 | Inside a convolutional network. The outputs (not the filters) of each layer (horizontally) of a typical convolutional network architecture applied to the image of a Samoyed dog (bottom left; and RGB (red, green, blue) inputs, bottom right). Each rectangular image is a feature map corresponding to the output for one of the learned features, detected at each of the image positions. Information flows bottom up, with lower-level features acting as oriented edge detectors, and a score is computed for each image class in output. ReLU, rectified linear unit.

Why all the excitement?

A convergence of hardware and algorithmic changes have thrust deep learning into the spotlight as it has been outperforming traditional methods recently

Ability to massively parallelize

- GPU’s increasing speed by 10-20 times
  GPU is 4-5 times faster than CPU
- dedicated hardware for convolutional neural nets
- massive data collection to train (e.g. siri, cortana, google, amazon alexa)

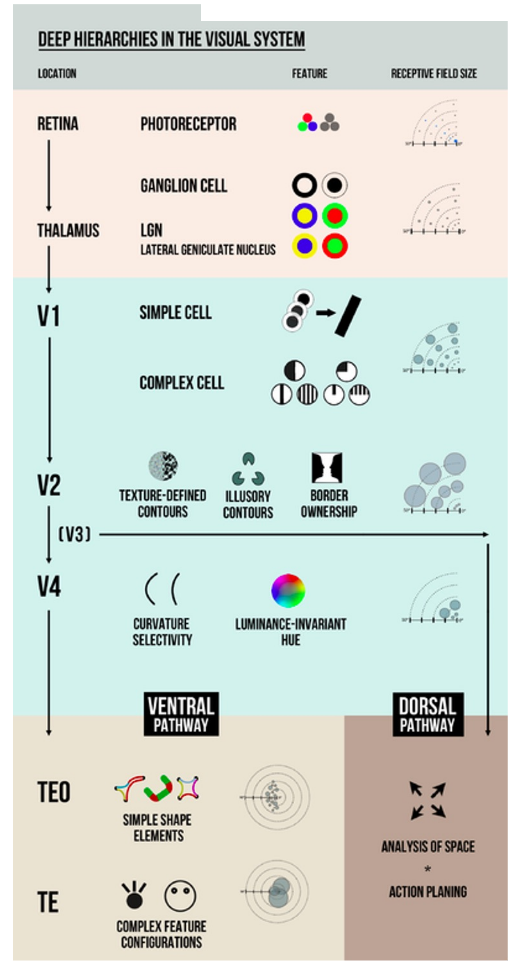
https://ieeexplore.ieee.org/document/8751930- **Performance Analysis and CPU vs GPU Comparison for Deep Learning**

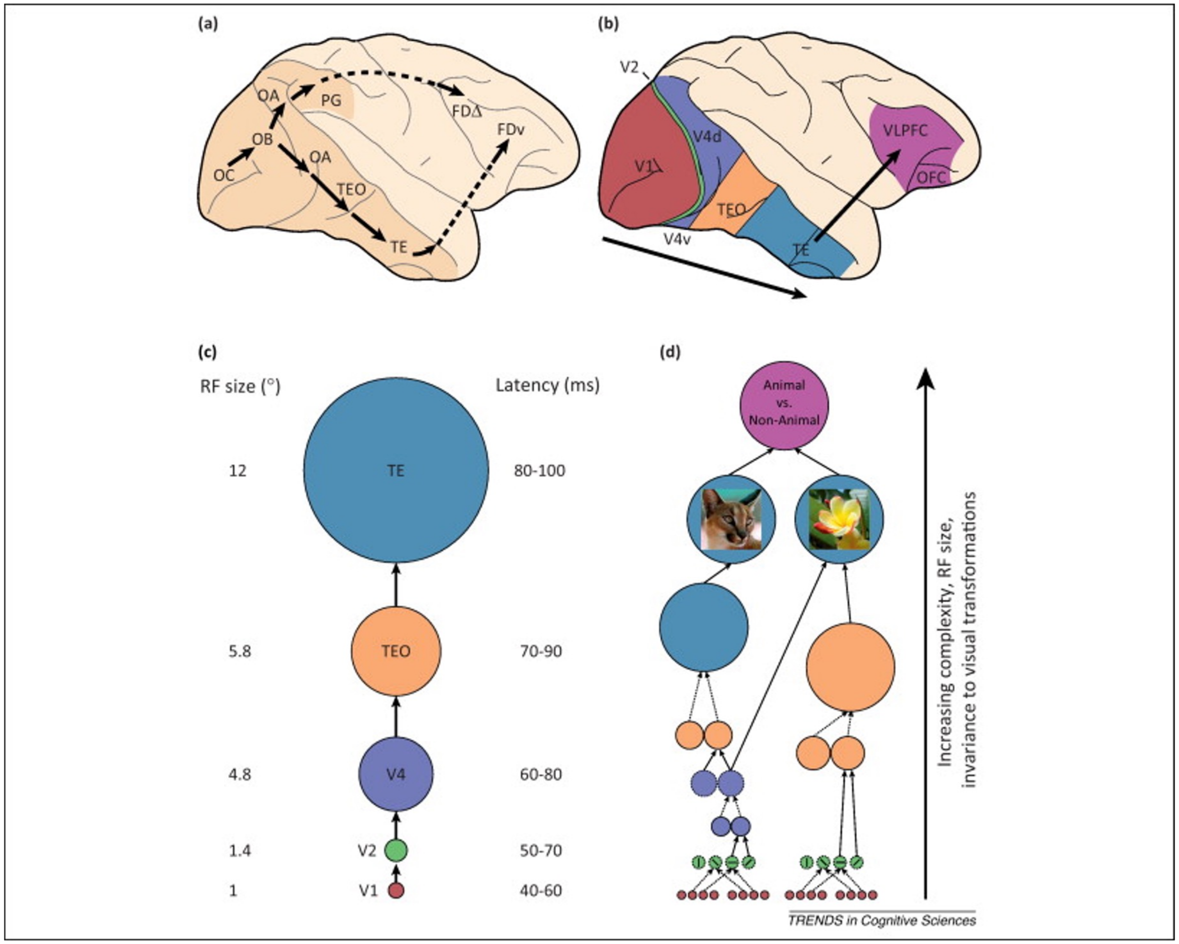
Algorithmic advances enabling deep learning

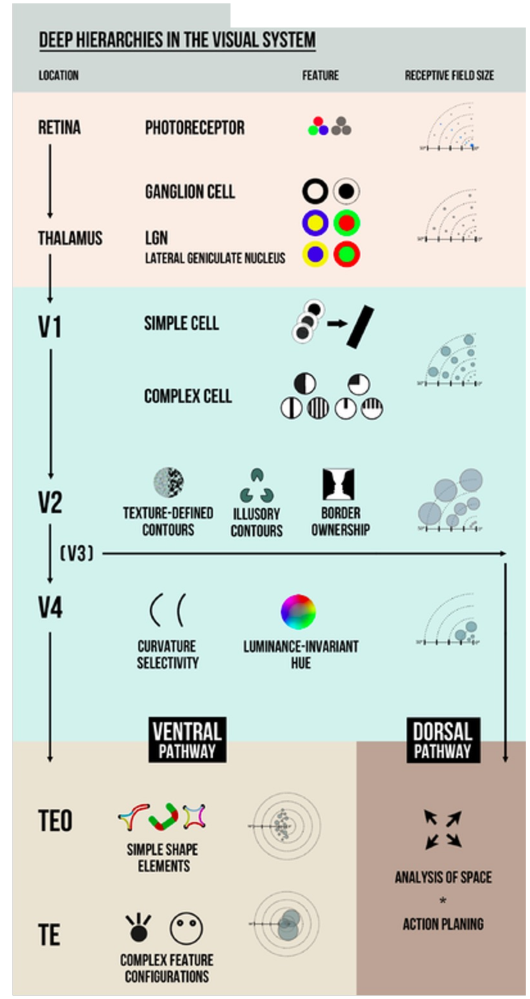
- hierarchy through backpropagation (been around since the 80's)
- stochastic gradient descent - adjust based on few samples at a time
- unsupervised "pre-training" of layers - unsupervised learning of features (optional)
- convolutional neural networks - copying learning everywhere
  - speeding computation and introducing invariant learning
- regularization techniques
  - avoiding overfitting, a common issue in neural nets

Deep learning and the brain

- are highly parallel
  - have fast feedforward processing
  - linear summation → nonlinear transform
  - surprisingly similar across areas (by definition in convolutional neural nets)
- are hierarchical
  - simple → complex features
  - smaller → larger receptive fields through local pooling
- rely on structural learning and exploitation
  - low-level primarily based on statistics of input
  - high-level are goal dependent
  - intermediate features are influenced by both

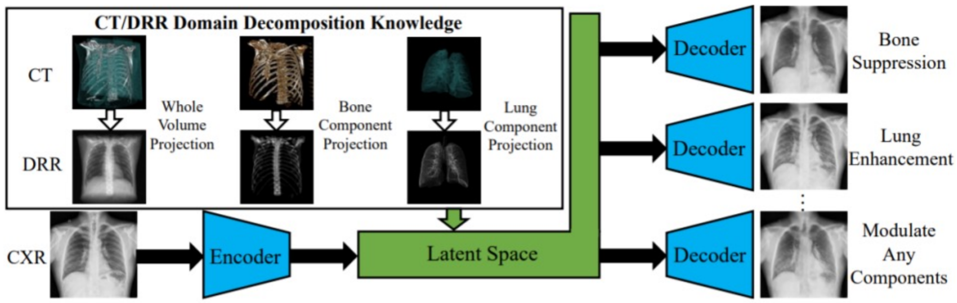



(a)
OA
PG
FDΔ
FDv
OB
OA
OC
TEO
TE
(b)
V2
V4d
V1
VLPFC
OFC
TEO
TE
V4v
(c)
RF size (°)
Latency (ms)
12
TE
80-100
5.8
TEO
70-90
4.8
V4
60-80
1.4
V2
50-70
1
V1
40-60
(d)
Animal vs. Non-Animal
Increasing complexity, RF size, invariance to visual transformations
TRENDS in Cognitive Sciences

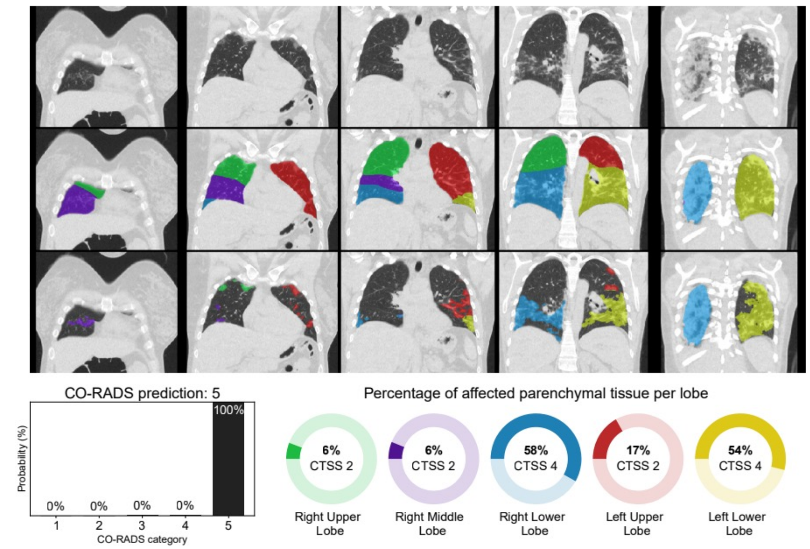

DEEP HIERARCHIES IN THE VISUAL SYSTEM
LOCATION
FEATURE
RECEPTIVE FIELD SIZE
RETINA
PHOTORECEPTOR
GANGLION CELL
THALAMUS
LGN
LATERAL GENICULATE NUCLEUS
V1
SIMPLE CELL
COMPLEX CELL
V2
TEXTURE-DEFINED CONTOURS
ILLUSORY CONTOURS
BORDER OWNERSHIP
(V3)
V4
CURVATURE SELECTIVITY
LUMINANCE-INVARIANT HUE
VENTRAL PATHWAY
DORSAL PATHWAY
TEO
SIMPLE SHAPE ELEMENTS
TE
COMPLEX FEATURE CONFIGURATIONS
ANALYSIS OF SPACE
*
ACTION PLANING

Summary to deep learning

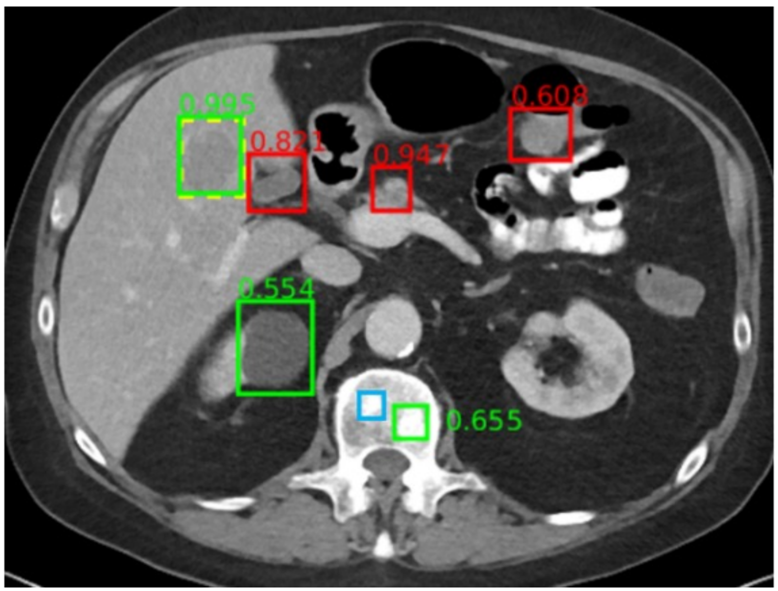
- Motivation and background
  - ubiquity and engineering feats that are possible now
  - learning → semi-supervised learning
- Neural nets
  - perceptron - ability and limiations
  - multi-layer networks and backpropagation
  - convolutional neural networks
- Recent advances enabling deep learning’s rise
  - parallelization, big data, and algorithmic advances
- Similarity to neural processing in the brain



Leveraging the anatomy knowledge embedded in CT to decompose a chest x-ray



Example output of the CORADS-AI system for a COVID-19 case. Top row shows coronal slices, the second row shows lobe segmentation and bottom row shows detected abnormal areas of patchy ground-glass and consolidation typical for COVID-19 infection. The CO-RADS prediction and CT severity score per lobe are displayed below the images



Example universal lesion detector for abdominal CT. In this axial image through the upper abdomen, a liver lesion was correctly detected with high confidence (0.995). A renal cyst (0.554) and a bone metastasis (0.655) were also detected correctly. False positives include normal pancreas (0.947), gallbladder (0.821), and bowel (0.608). A subtle bone metastasis (blue box) was missed