Deep Learning for Neuroimaging: an introduction

Pamela K. Douglas
PRNI Educational Course
OHBM 2020

deep learning models (the hype)

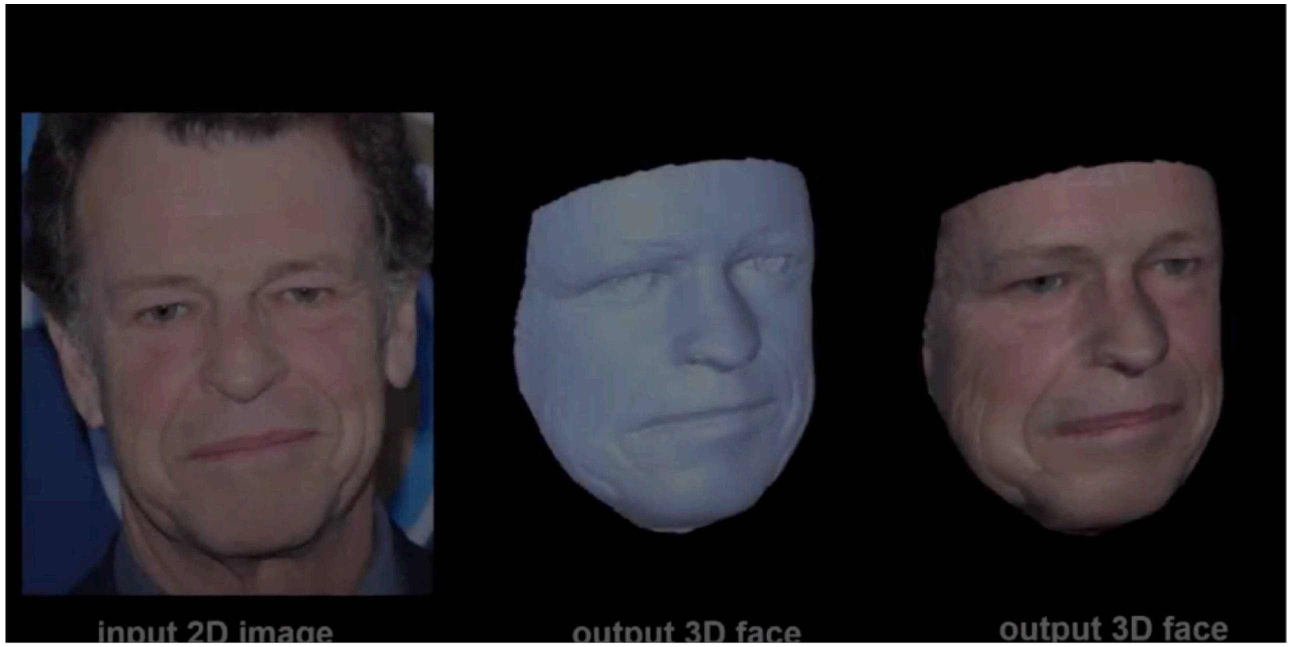
Deep Reinforcement Learning

Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." *arXiv preprint arXiv:1312.5602* (2013).

Panoptic Segmentation : Self Driving Cars

Mohan & Valada (2020)

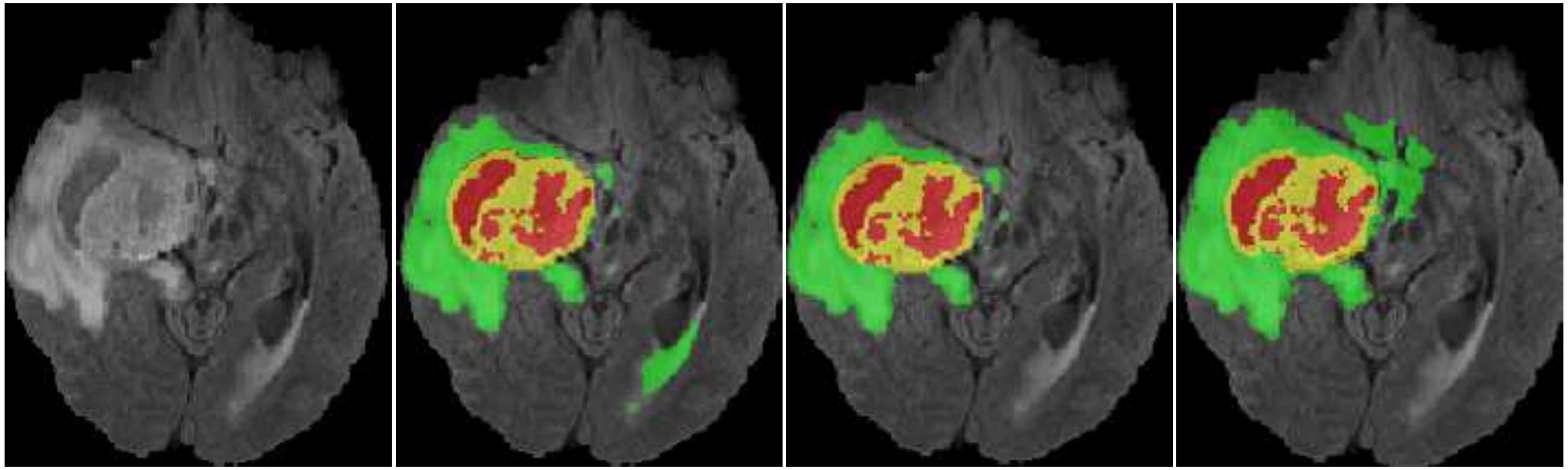
3-d Facial Recognition from 2-d



Sela et al. "Unrestricted Facial Geometry Reconstruction Using Image-to-Image Translation" (2017)

new discoveries from deep learning

- General wisdom: radiologists should examine the tumor borders (alone) to determine staging and predict outcomes



- Convolutional neural network predictions based on texture features from within the tumor volumes are diagnostic of cerebral gliomas and survival prediction

Alex, V. Et al. (2017)
Douglas, DB & Wintermark (in progress)

overview

- What is an Artificial Neural Network?
- What is Deep Learning?
- How is deep learning useful for neuroimagers?
- Resources & Links

what are artificial neural networks?

- neural networks are statistical models loosely inspired by biological neurons and their connectivity
- An early bridge between spiking neural activity and categorization - a hallmark of cognition (Kriegeskorte 2015)
- In a classic supervised setting, a NN model learns parameters θ that best approximate a function that maps inputs to the desired outputs

$$y = f(x;\theta, w) = \phi(x;\theta)^T$$

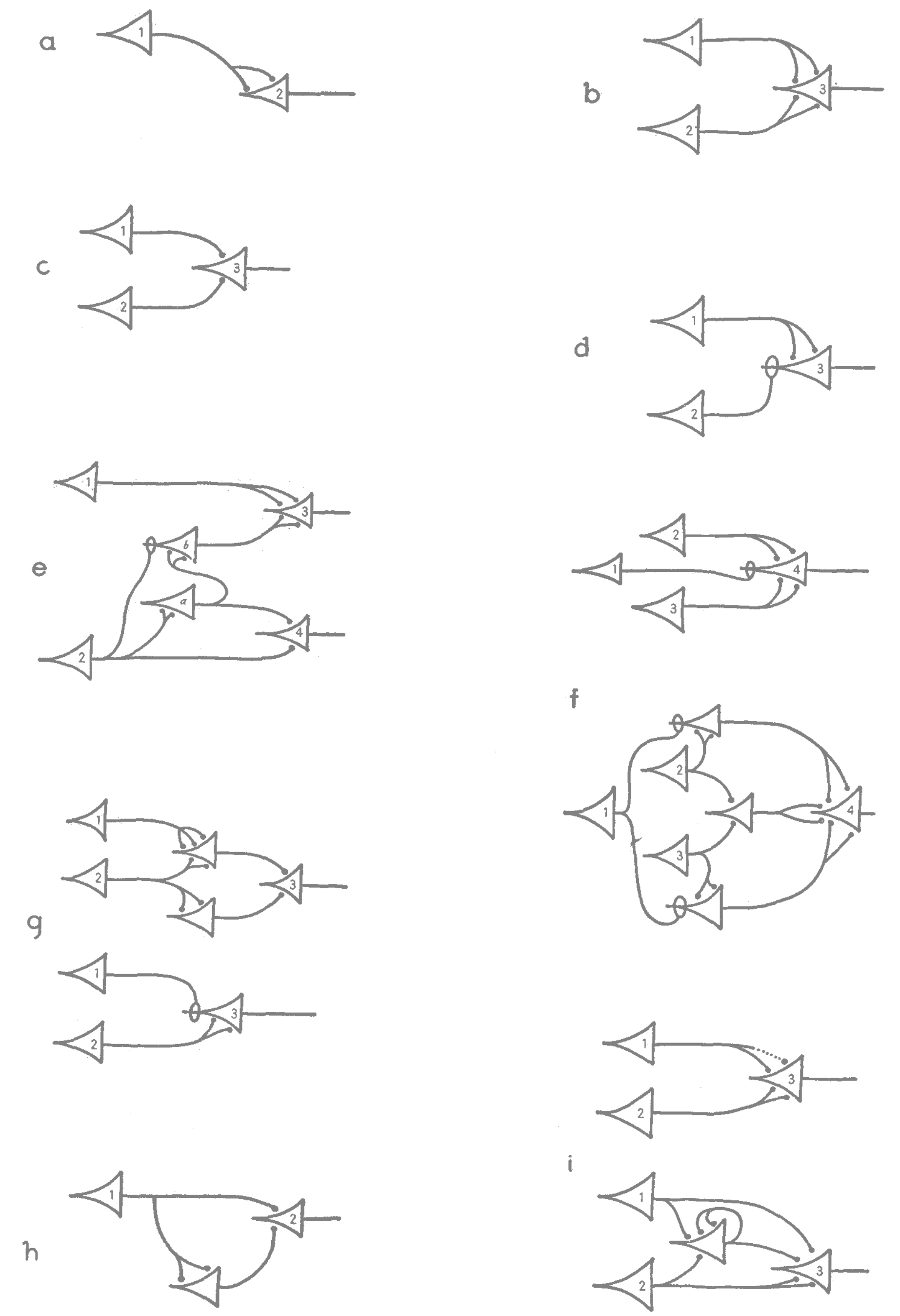


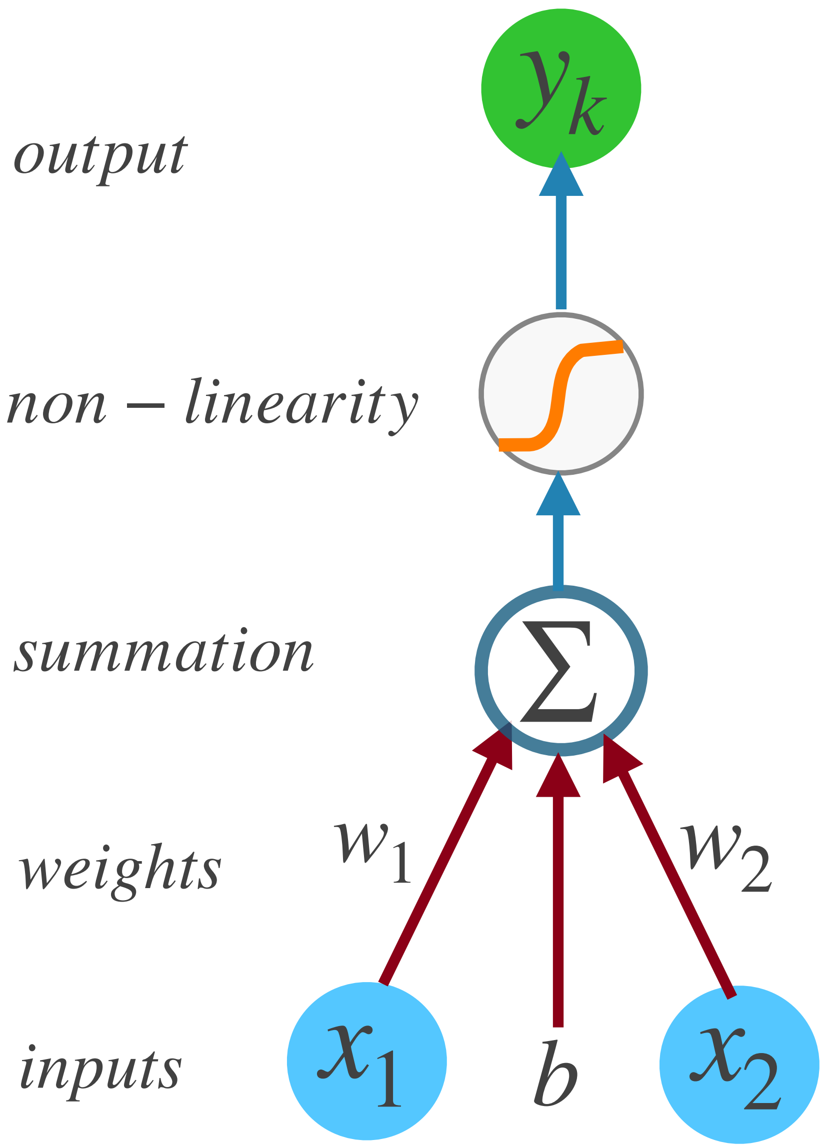
FIGURE 1

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

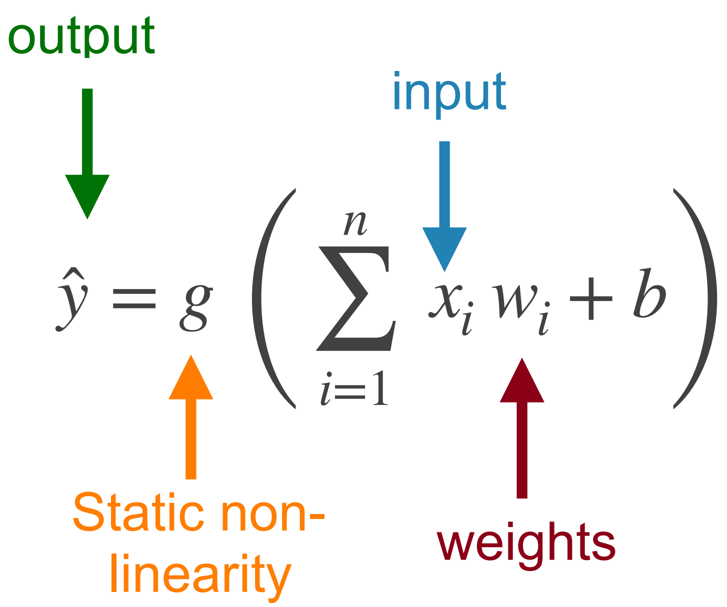
WARREN S. MCCULLOCH AND WALTER PITTS

1943

neural network architecture: basic unit



- Outputs are a function of these non-linear activations

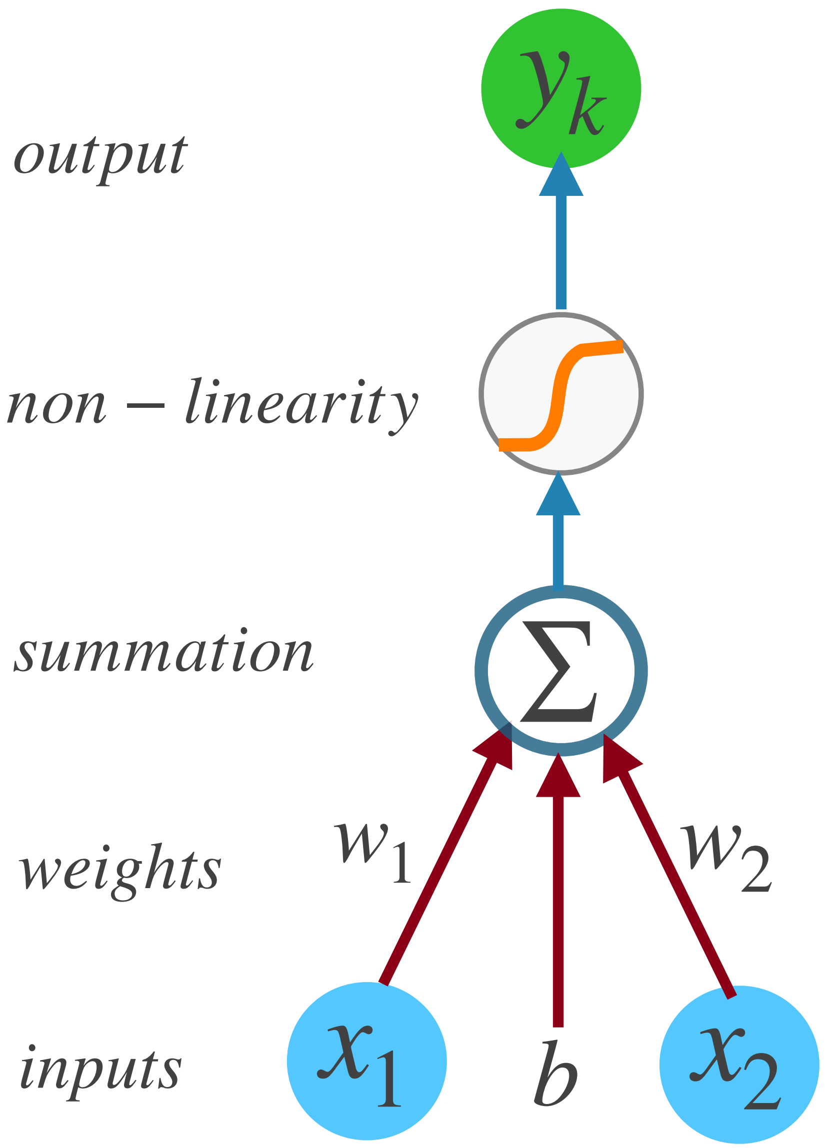


$$\hat{y} = g\left(\sum_{i=1}^{n} x_i w_i + b\right)$$

- They are non-linear; activation functions introduce non-linearities
- Like neurons, units receive & summate inputs from multiple units

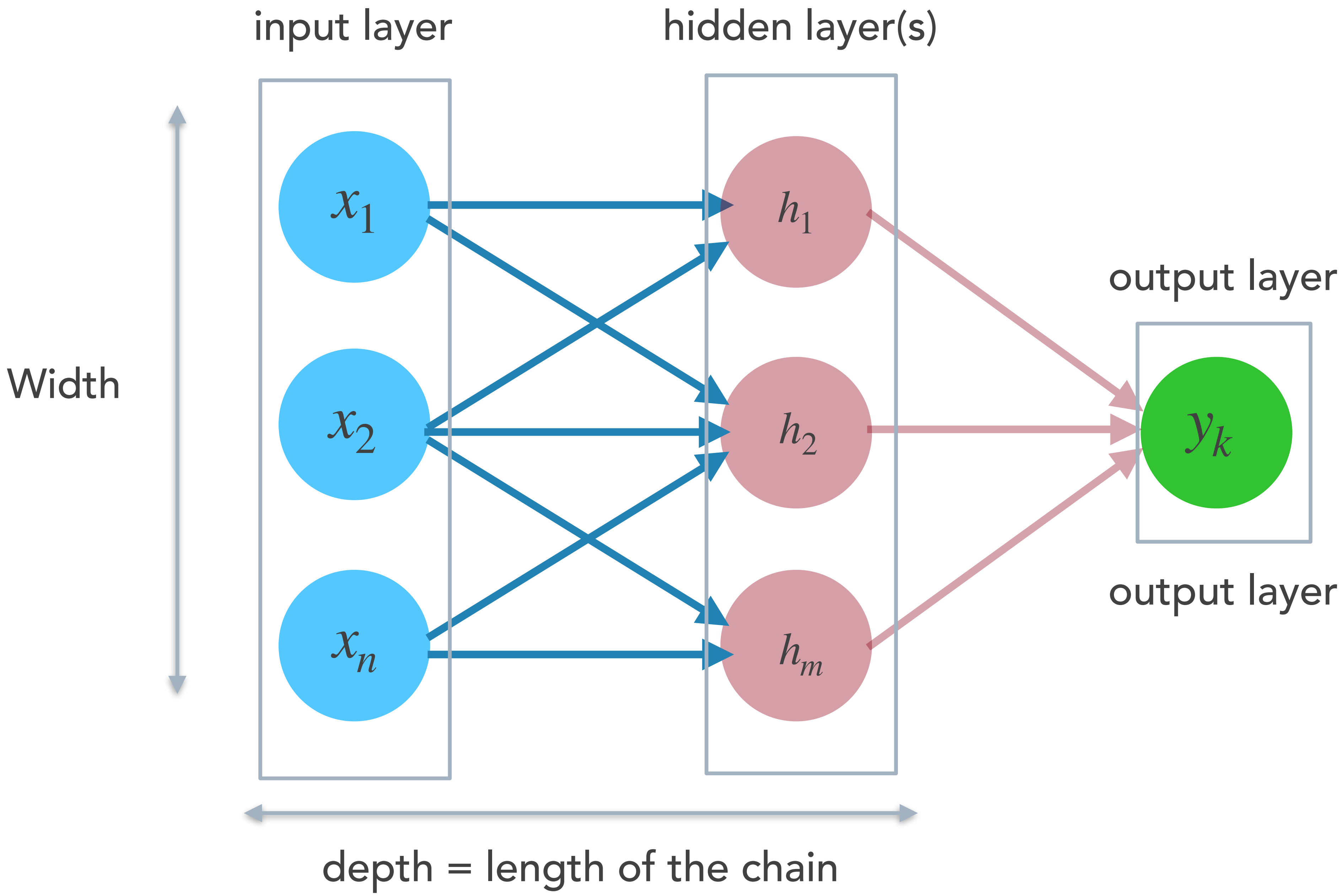
Inspired by Figure 1a from Kriegeskorte (2015)

neural network architecture: basic unit



- The goal of the model is to approximate a non-linear function that maps input variables $\{x_i\}$ to outputs $\{y_k\}$ by adjusting weight parameters (w_i)…

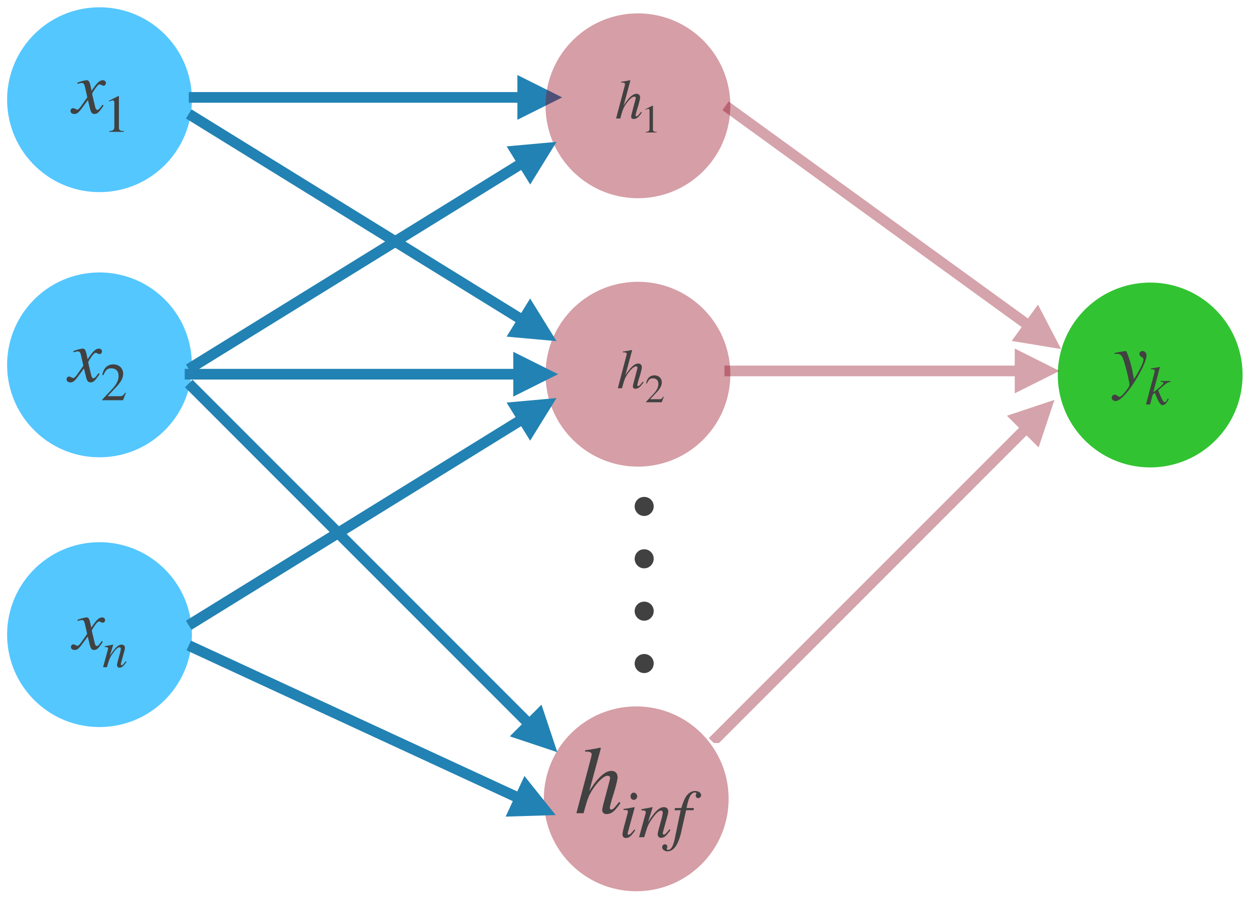
feedforward networks: chain of functions



- Feedforward models implement a chain of functions typically represented by acyclic computational graphs with input, hidden, and output variables represented by nodes
- Weight parameters are represented by links or directed edges between nodes

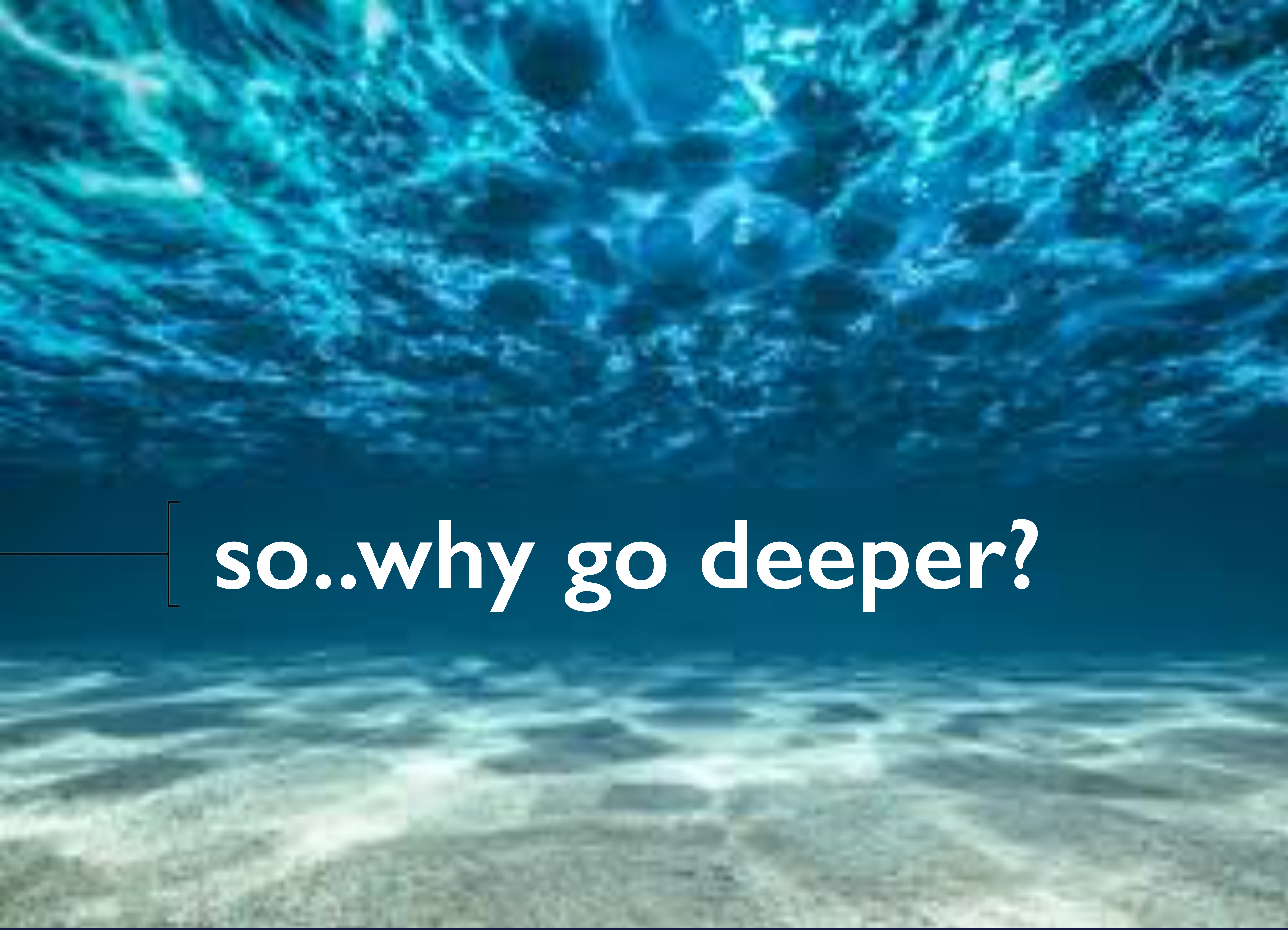
Bishop ,"Pattern Recognition & Machine Learning" Book

universal approximation theorem



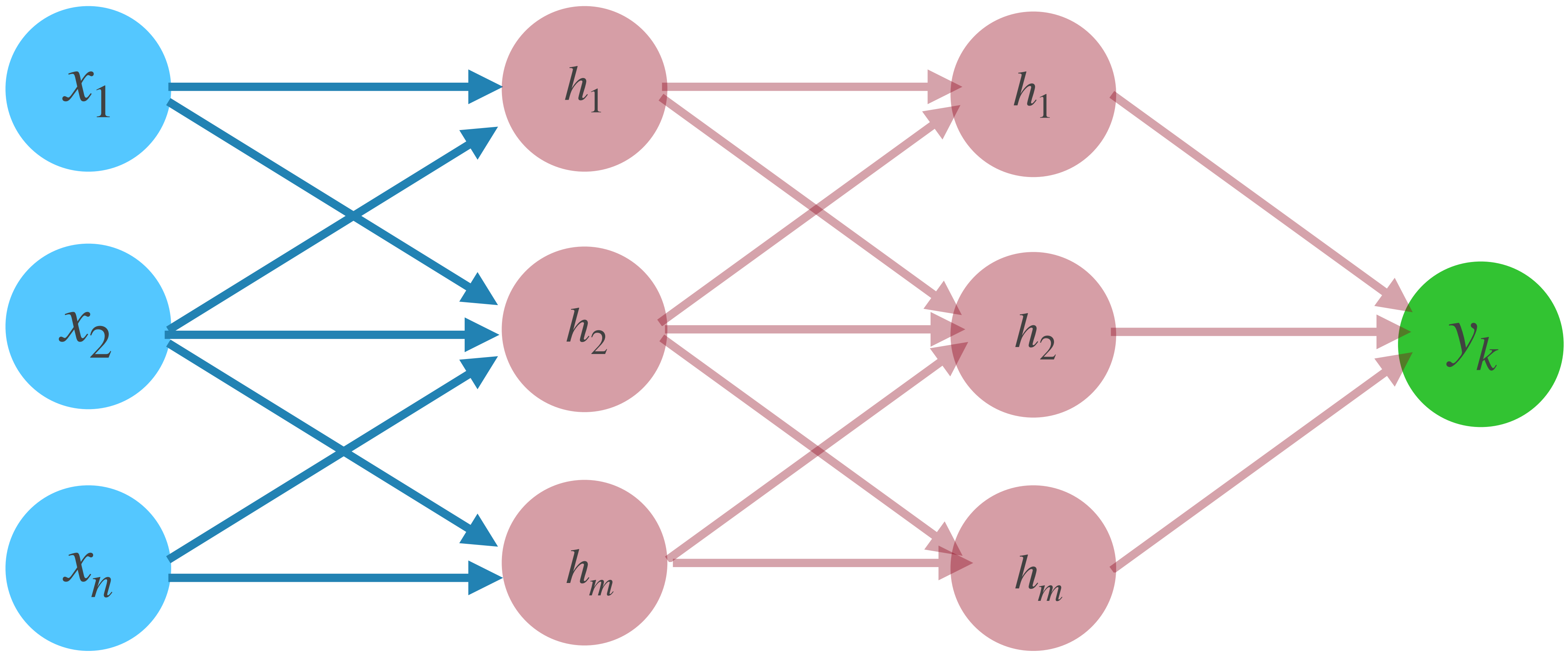
- A **feedforward NN** model with at least one hidden layer and nonlinear activation or squashing function is a universal function approximator
- In practice, one hidden layer is enough to represent (not necessarily learn) an approximation of any function to an arbitrary degree of accuracy.

Hornik et al. 1989; Cybenko ,1989; Leshno et al. 1993



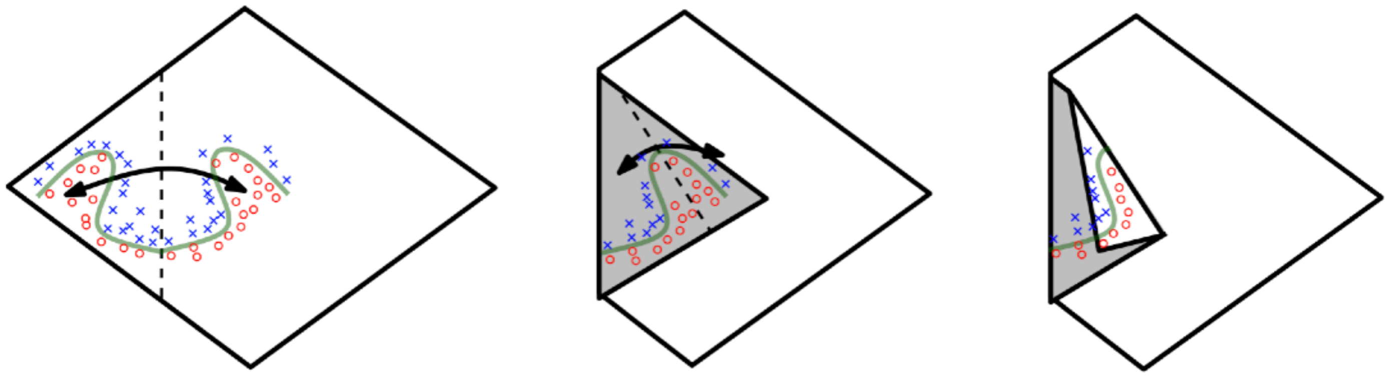
so..why go deeper?

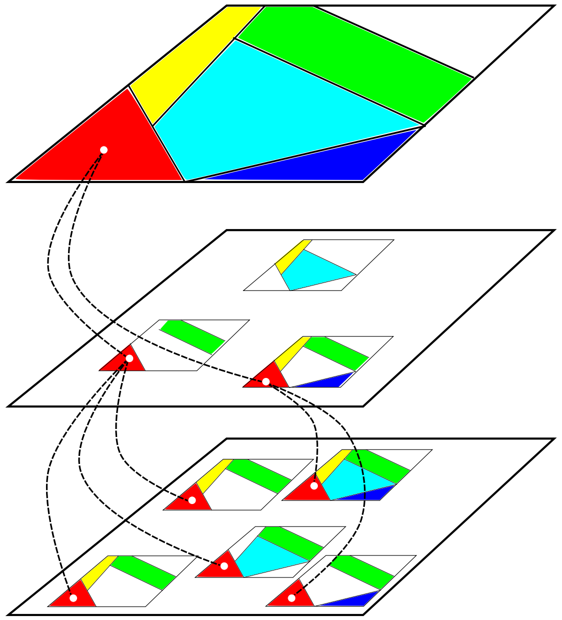
deep learning: hierarchical models (>1 hidden layer)



- Instead of hand crafted or manually engineered features — Deep feedforward networks learn & discover complex representations composed of simpler representations through their layers
- This may be useful if a task is comprised of a sequence of multiple steps
- Or if a representation is composed of more simple representations (e.g., vision)

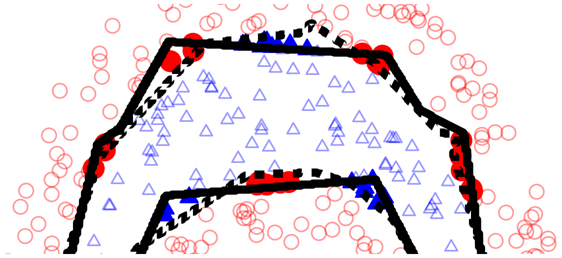
deep learning: the advantage of depth





- Empirically, depth results in greater generalization
- Often, shallow networks require exponentially more parameters and tend to overfit & Deep models can represent complex functions more concisely (e.g., Bengio 2009)
- Sparse models with less parameters are less susceptible to numerical issues

  - For a fascinating study on numerical issues & reproducibility in neuroimaging see OHBM poster, "*Fuzzy: An Ecosystem for Evaluating the Stability of Pipelines through Monte Carlo Arithmetic*" Kiar et. Al (2020)



Montufar et al. 2014; Goodfellow et al. 2016

ingredients for deep learning

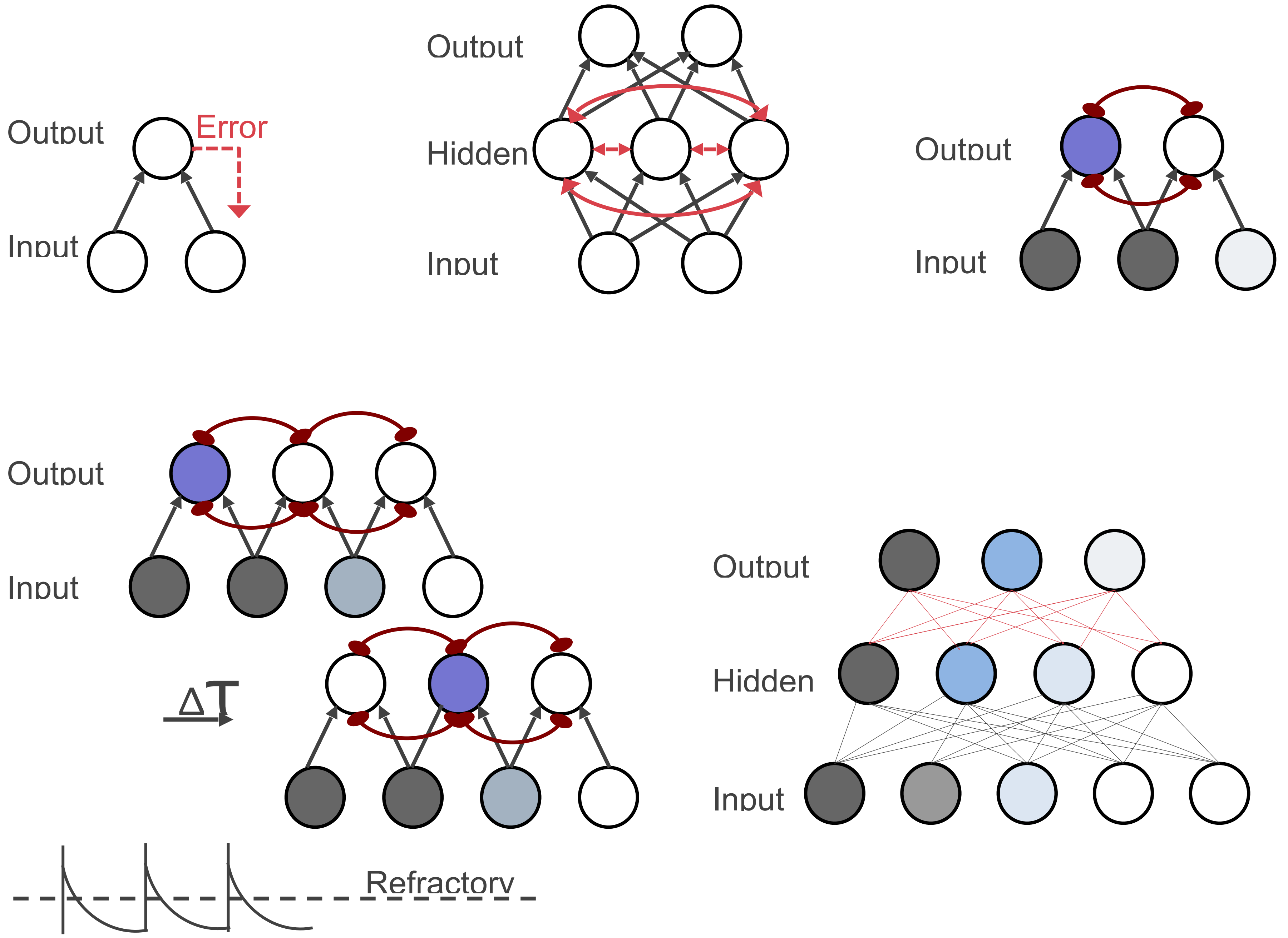
1.) Model Architecture

2.) Objective/Cost Function

3.) Optimization Procedure

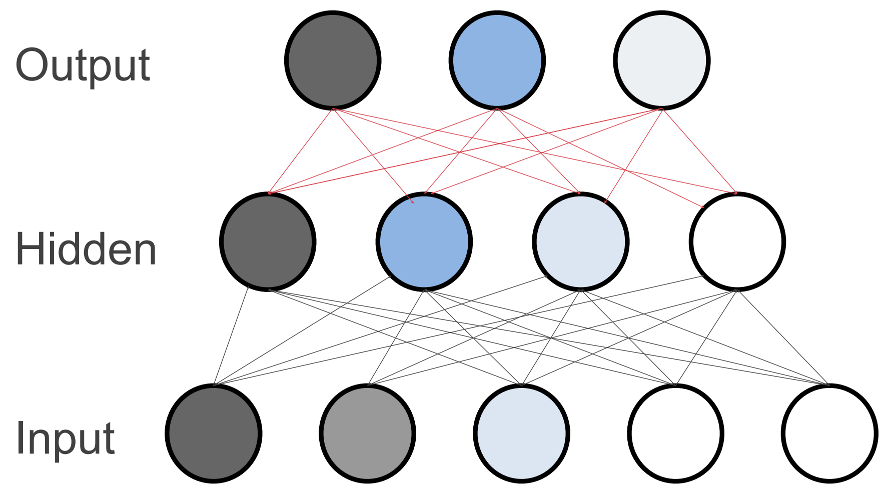
4.) Data

large taxonomy of models



- how to choose?

I.) model architecture: design considerations



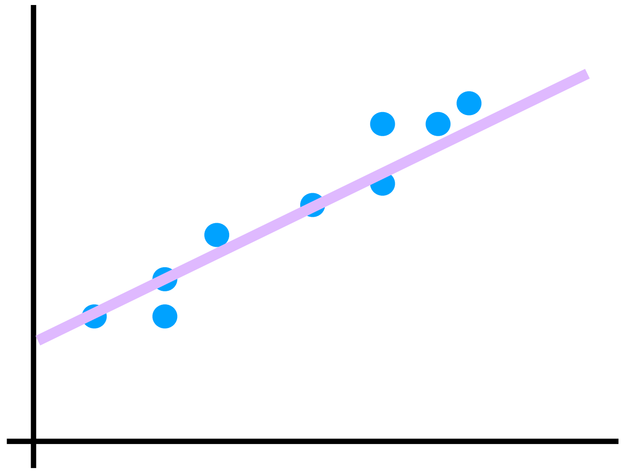
- *How many layers?*
- *How many units?*
- *Connectivity?*

- Too shallow —> too many parameters; Excessive depth can sometimes lead to vanishing (or rarely, exploding) gradients) (Hochreiter 1991; Bengio et al. 1993)
- No free lunch: averaged over all possible data-generating distributions, every algorithm will have the same error rate on unseen samples (Wolpert 1996; for Neuroimaging example, Douglas et al. 2010)
- Biology to constrain network topology: if using deep learning as a model for brain information processing (Kriegeskorte & Douglas 2018),

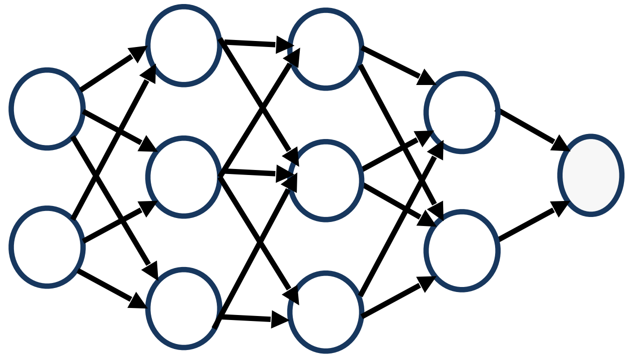
2.) objective function

- Objective function: Just like with traditional ML, the objective function computes the disparity between the model and the training data

- Minimization: If framed as a minimization, it is often called a cost function or a loss function

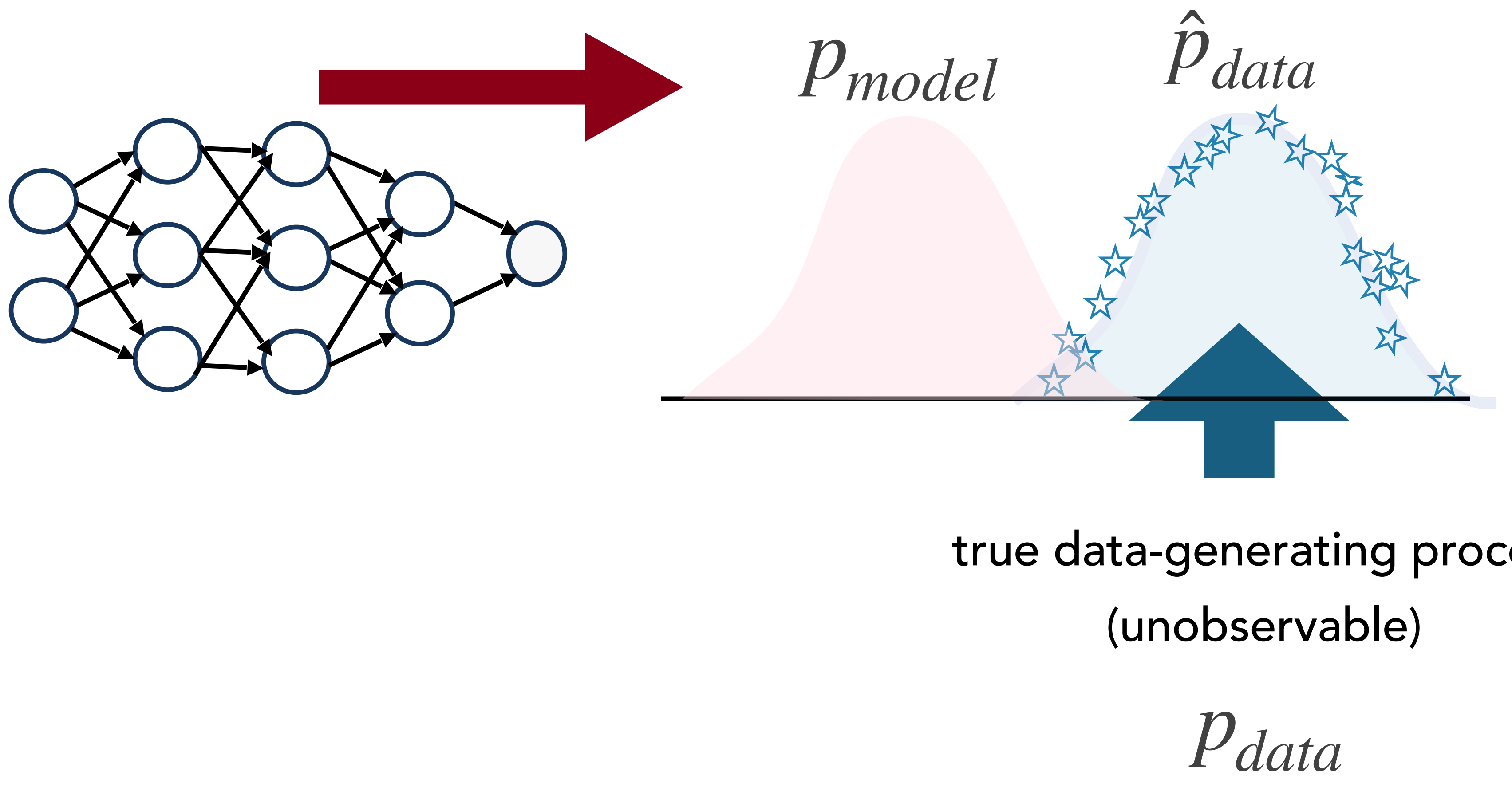


MSE



?

maximum likelihood estimation



Maximum Likelihood Estimation (MLE):

- provides a framework for estimating model parameters given our training data via optimization;
- can be thought of a as attempt to make model probability distribution, p_{model} match empirical distribution, $\hat{p}_{data}$
- special case of maximum a posteriori (MAP) with uniform priors

Myung (2002) A tutorial on Maximum Likelihood Estimation

maximum likelihood estimation

Goal: find parameters that maximize the likelihood of observing the data given the model

$$\theta_{ML} = argmax \sum_i log\ p_{model}(x^{(i)}; \theta)$$

note : log likelihood is more computationally efficient

Or equivalently, we can minimize the dissimilarity between distributions using KL divergence

$$D_{KL}(\hat{p}_{data} || p_{model}) = E_{x \sim \hat{p}_{data}}[log\ \hat{p}_{data}(x) - log\ p_{model}(x)]$$

training data　　　model

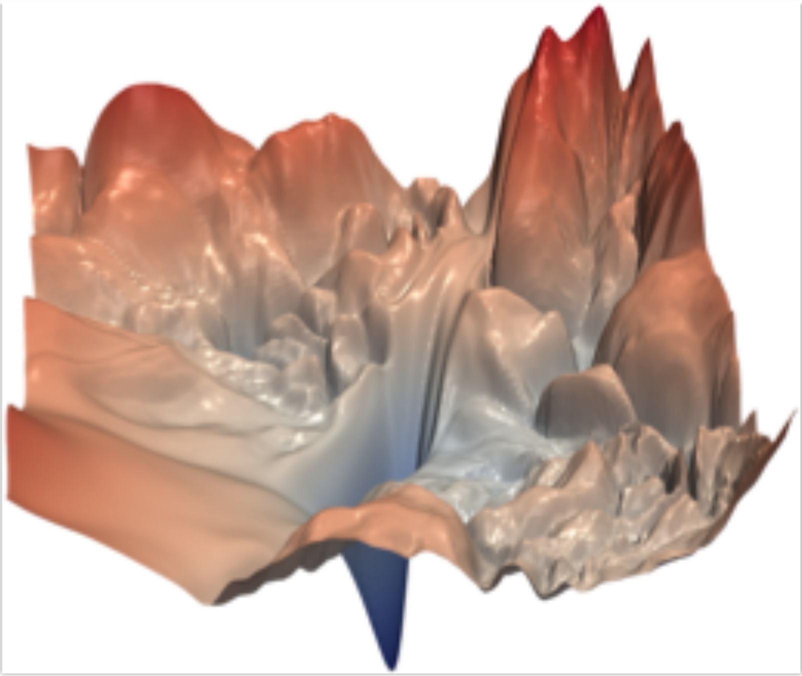
First term does not depend on model, and we are left with the cross-entropy

$$-E_{x \sim \hat{p}_{data}}[log\ p_{model}(x)]$$

3.) (numerical) optimization procedure

- Minimizing Cost function: The optimization procedure aims to finds the model parameters that correspond to a good representation of the (training) data, and the lowest loss/cost

- Finding a minima can be complicated

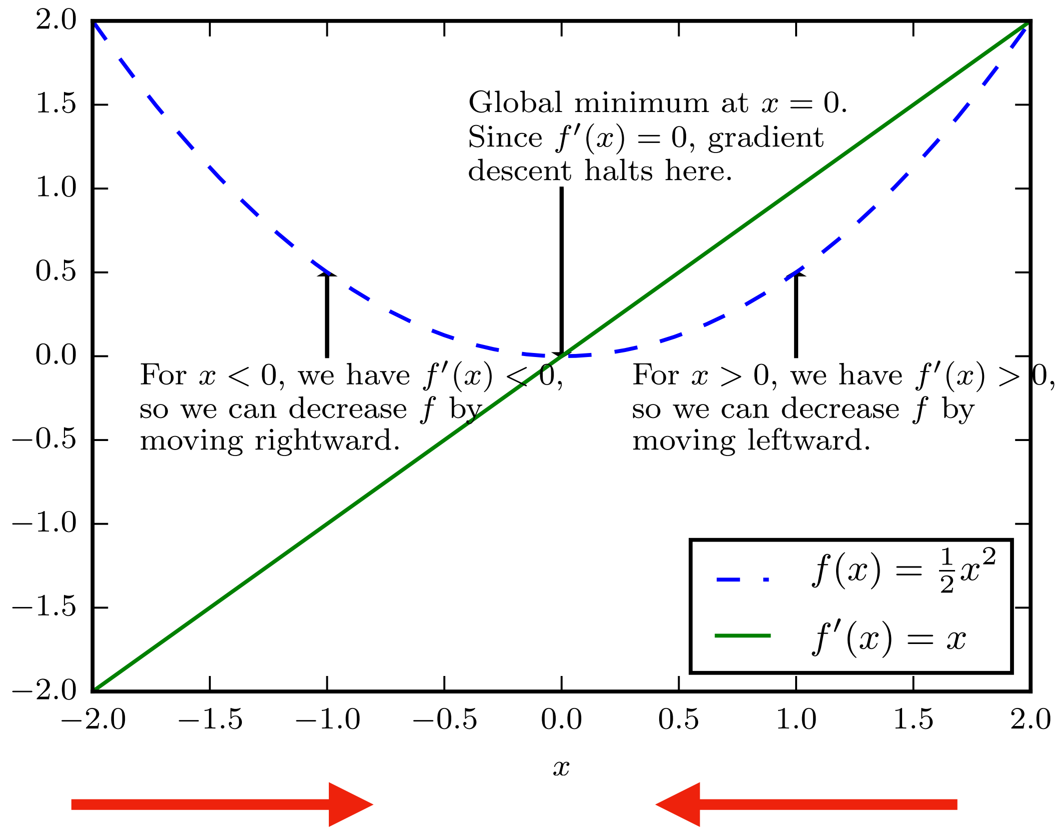


Loss Surface

Li et al. (2017) https://arxiv.org/pdf/1712.09913.pdf.

gradient based learning

single input - take derivative f′(x)



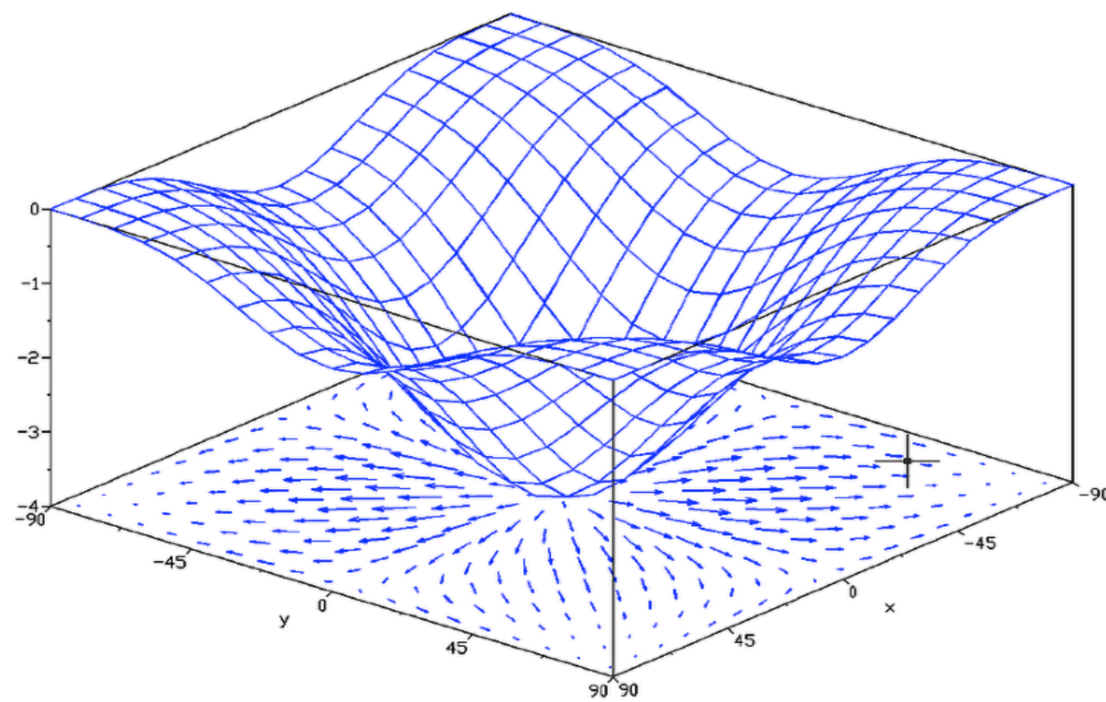
Move in opposite direction from the derivative

$$x' = x - \epsilon \nabla_x f(x)$$

Learning rate

Multiple inputs - take gradient

$$\nabla_x f(x) = \frac{\partial}{\partial x_1} f(x) + \frac{\partial}{\partial x_2} f(x) + \ldots \frac{\partial}{\partial x_n} f(x)$$



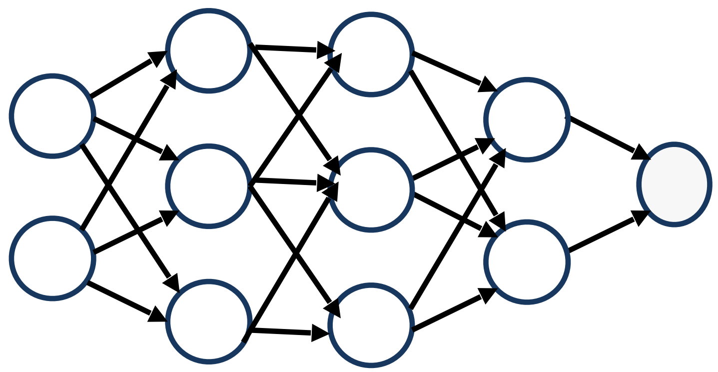
gradient descent

Stochastic Gradient Descent (SGD): a popular choice that randomly selects an example or a mini batch of examples to estimate the expected gradient for each update

From The Deep Learning Book (Goodfellow, Bengio, Courville)

backpropagation: clever way to calculate the gradient

Forward Pass: Tells us the Model's Current Predictions



Iterate until convergence

Backpropagation: Computes gradient
Gradient Descent: performs learning (iteratively adjust parameters) based on gradient

Uses the chain rule:

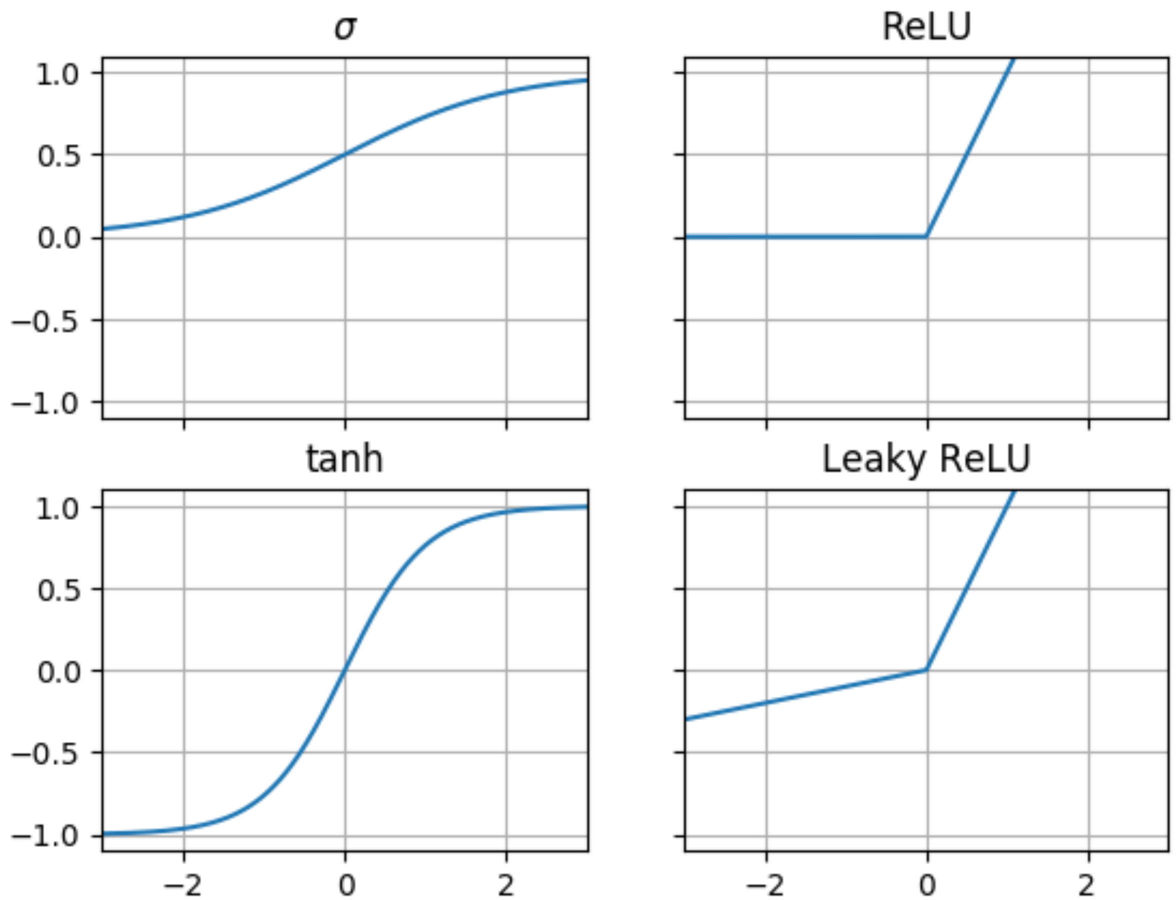
$$\nabla_x f(x) = \frac{\partial}{\partial x_1} f(x) + \frac{\partial}{\partial x_2} f(x) + \ldots \frac{\partial}{\partial x_n} f(x)$$

Efficient algorithm that avoids repeating computations

Rumelhart & McClelland 1988

activation function

Rectified Linear (ReLu) : used 90% of the time



For many years, general wisdom amongst practitioners suggested avoiding the ReLU function, due to its flat area. It is now considered the default activation function.

Which one is most biologically plausible?

Tanh or signmoid may resemble current /voltage relationship for ion channels more closely

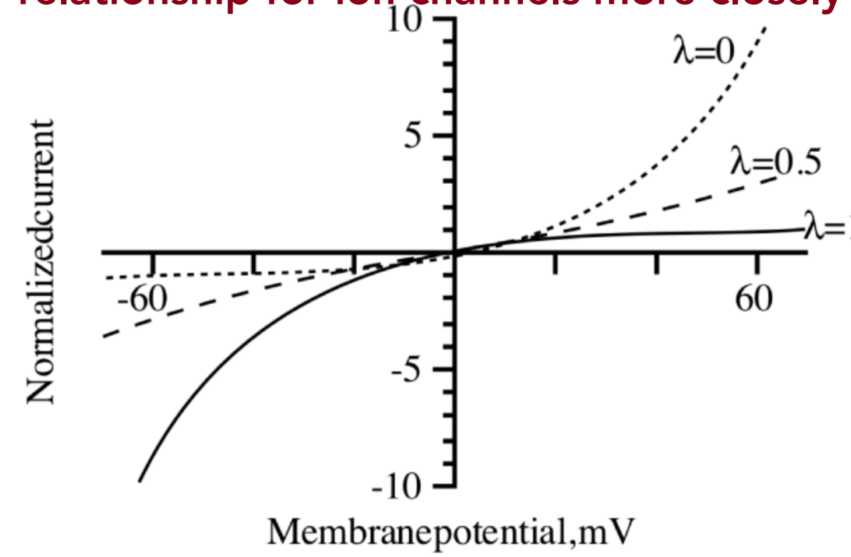
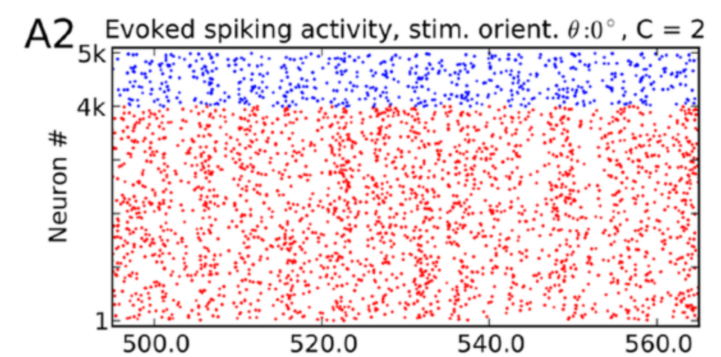


Fig. 1 Current-voltage relationships for the single barrier model (see Eqn. 32) with zero equilibrium potential. λ is the fraction of the transmembrane potential seen at the barrier peak. Membrane potential is positive inside the cell and current is positive outward, as usual. Current is plotted in normalized form, as *I/(const)*exp(-*G*/*RT*)*z FA*, see Eqn. 32.



But neural selectivity and firing rates may (sometimes) be approximately linear - resembling ReLU

Figure from Alan Young's course notes at Johns Hopkins

ingredients for deep learning

1.) Model Architecture

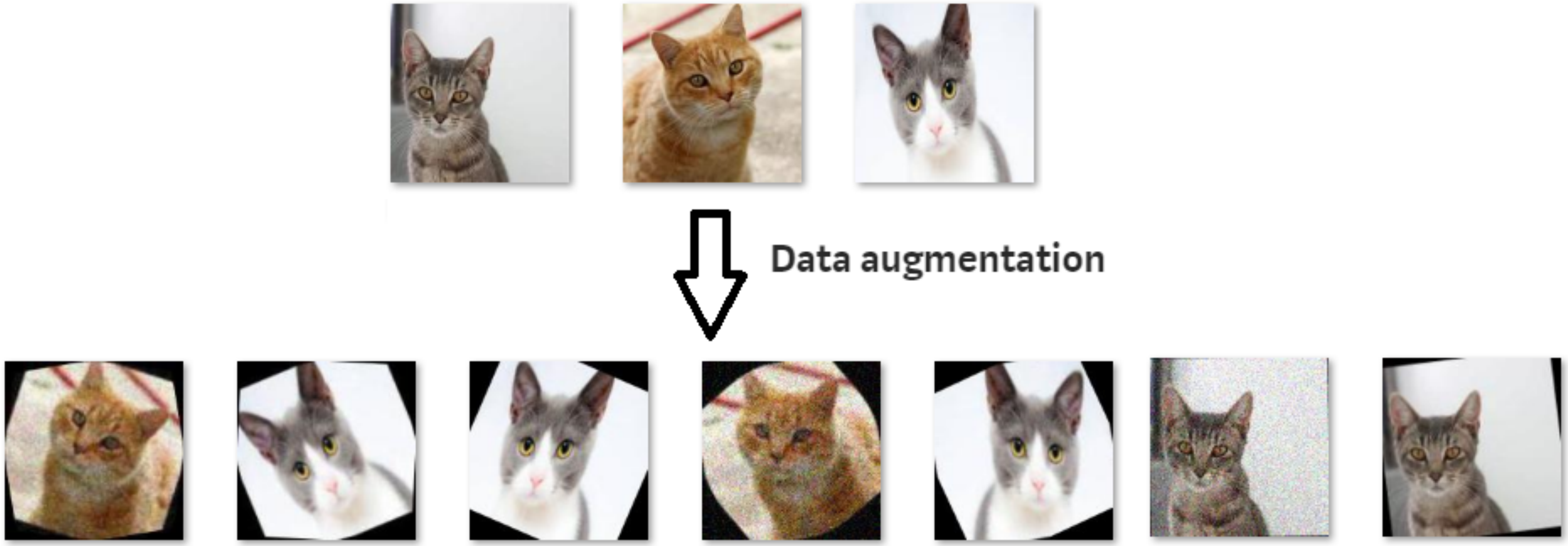
2.) Objective/Cost Function

3.) Optimization Procedure

4.) Data

data & augmentation

- The more data the more effective the deep learning strategy… MLE solutions converge to the true parameter value as data increases.

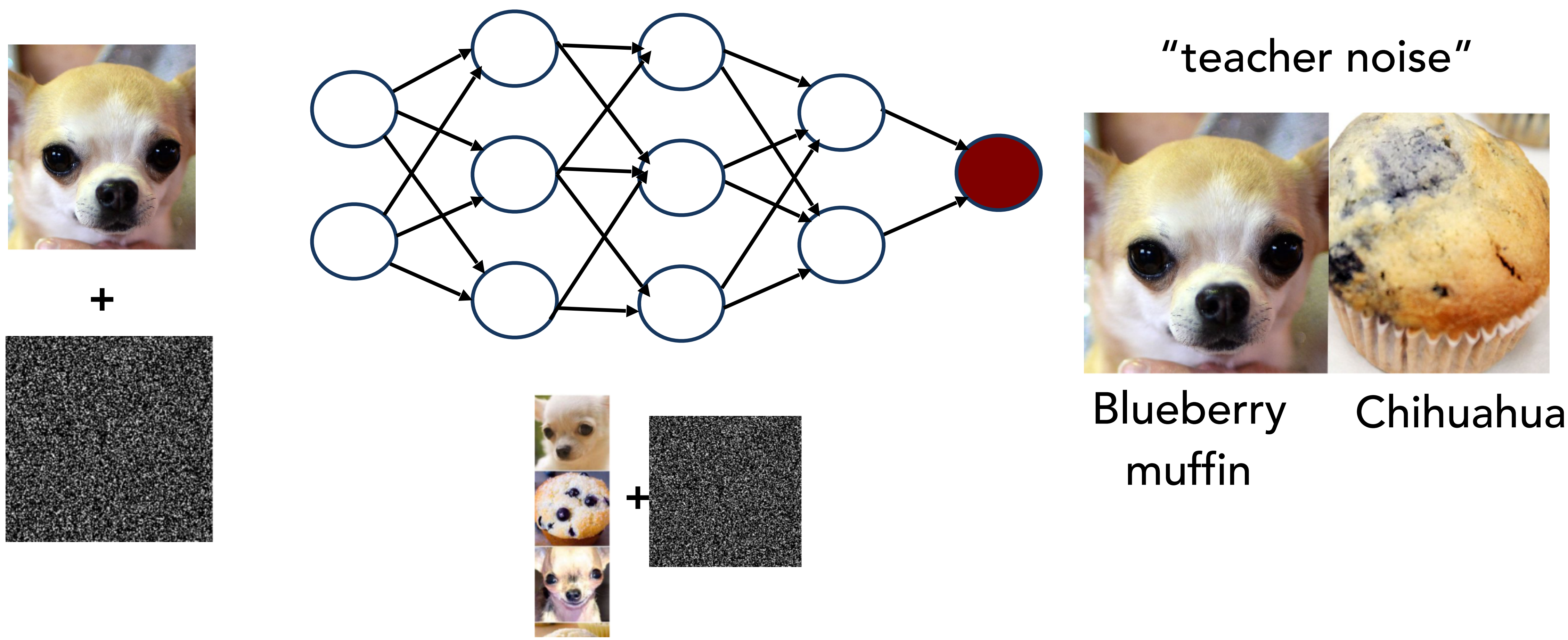


- Creating additional data by applying small **translations**, **rotations**, **cropping**, **scaling**, and **color shifts** to your original data can boost generalization

Image from Thomas Hiblot; e.g., Wang & Perez (2017)

augmentation with noise

- noise can be useful for regularization, data augmentation & adversarial training.

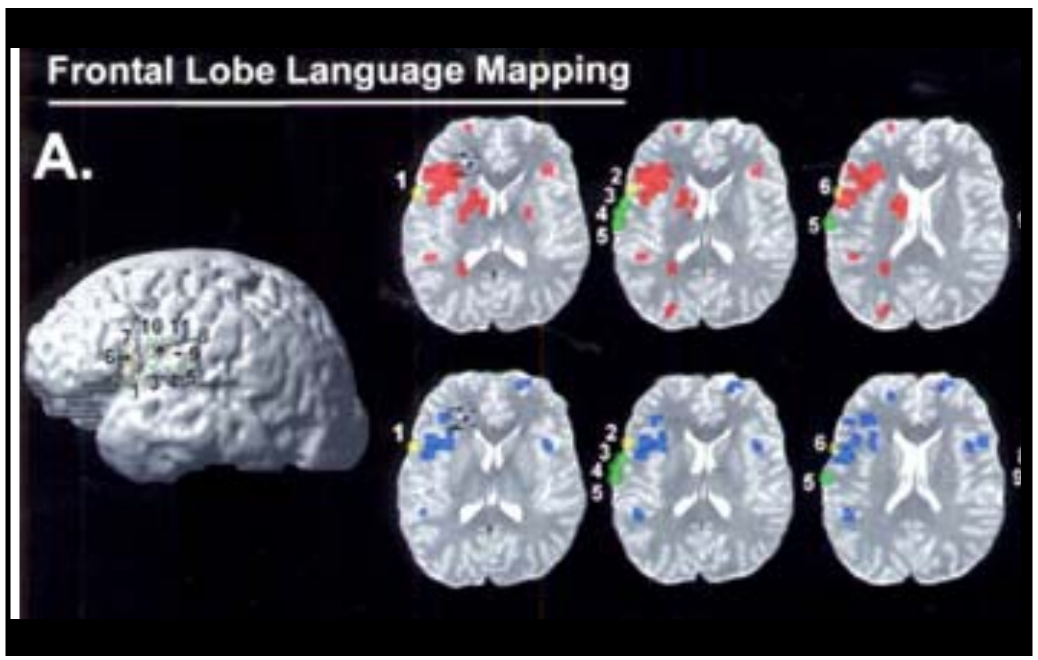


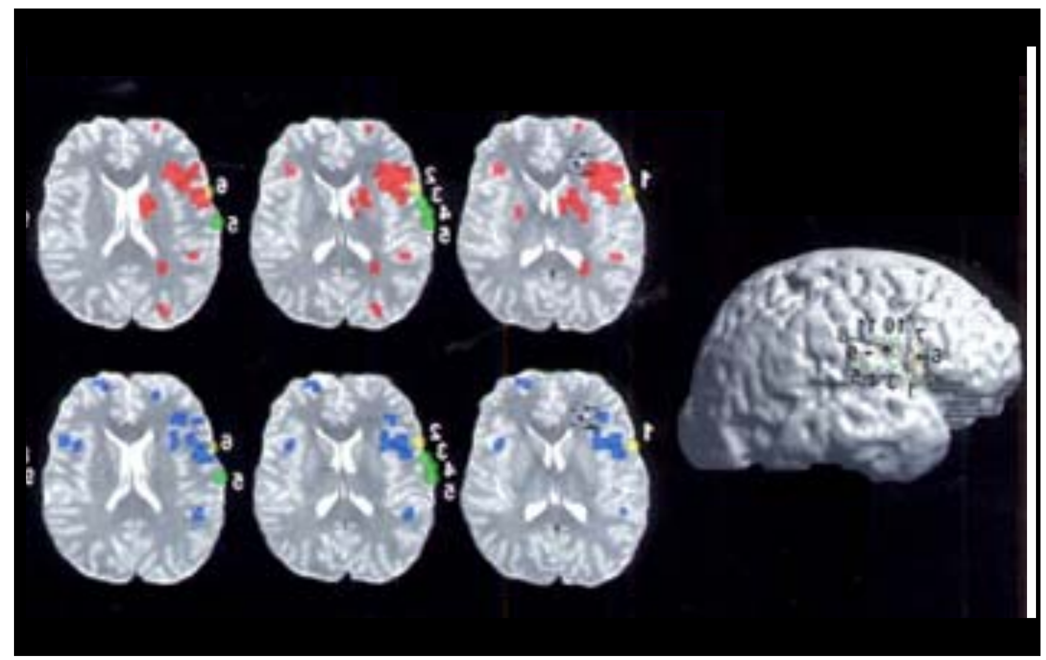
- noise can be added to inputs, hidden layers, output labels, numerical calculations, or optimization schemes

data augmentation

- Left / right flips should be avoided (in neuroimaging)

"Preoperative fMR imaging of language in patients with AVMs"





Radiological convention?

Pouratian, N, Bookheimer, S. Et al. (2002)

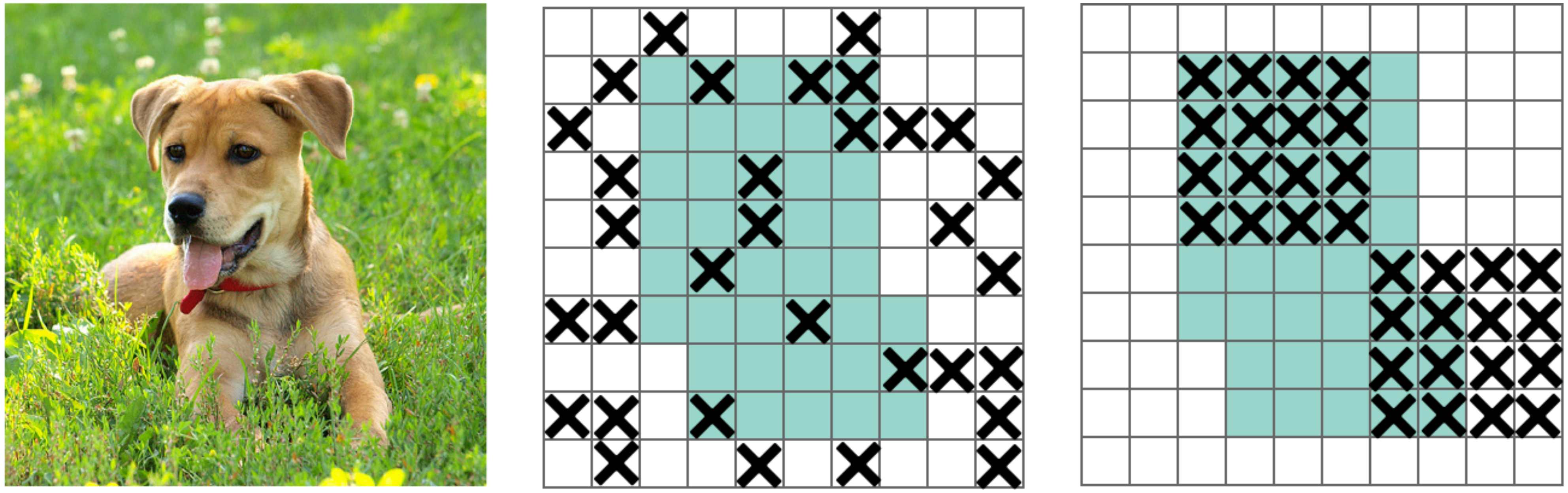
regularization for deep learning

- Regularization: add a penalty to the cost function, called a regularizer that tends to result in the model putting less weight (e.g., weight decay) or weight on fewer parameters (e.g., L1)

$$\tilde{J}(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{y}) = J(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{y}) + \alpha\Omega(\boldsymbol{\theta}),$$

- Typically used to penalize complexity or control capacity - especially useful for small data sets relative to the dimensions

regularization for deep learning

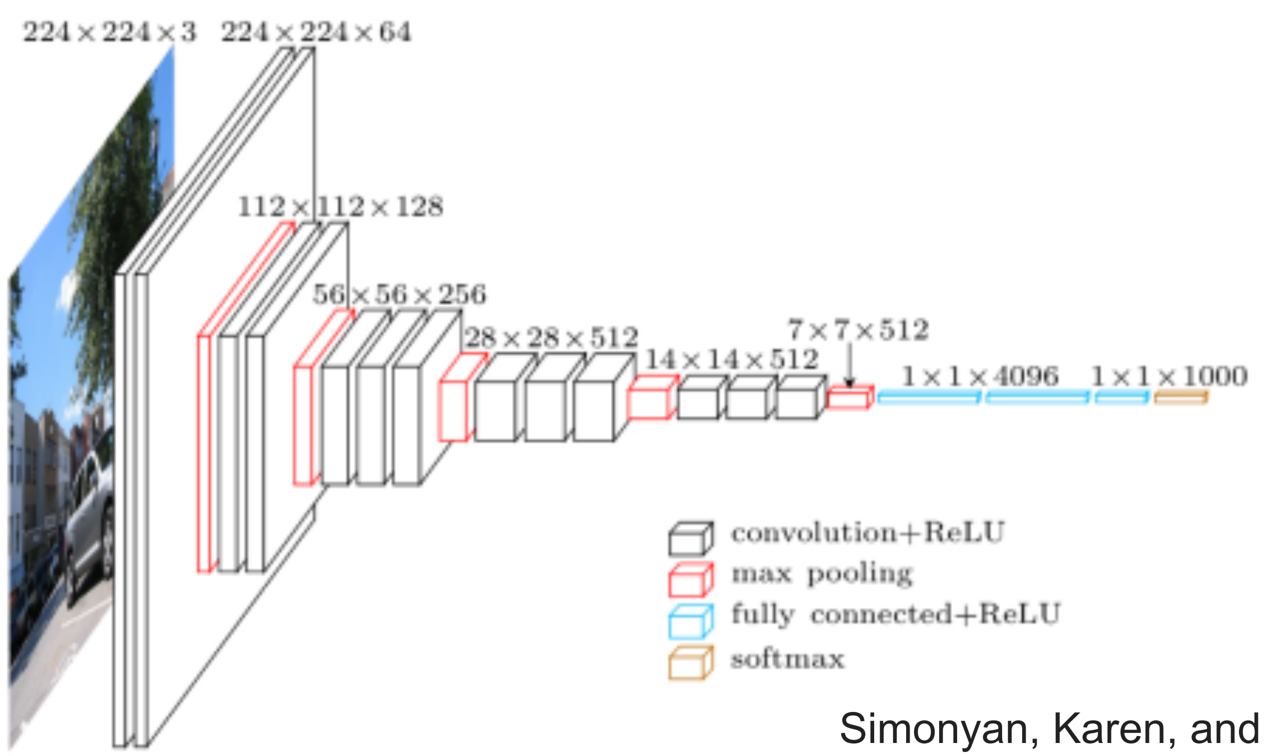


(a) (b) (c)

1. dropout (Srivastava et al. 2014)
2. stochastic rounding (Gupta 2015)
3. label noise (Rolnick et al. 2018)
4. droppath - drop entire layer during training (Larson et al. 2017)
5. dropblock (shown above; Giasi 2018)
6. Many others (Shake-Shake, etc)

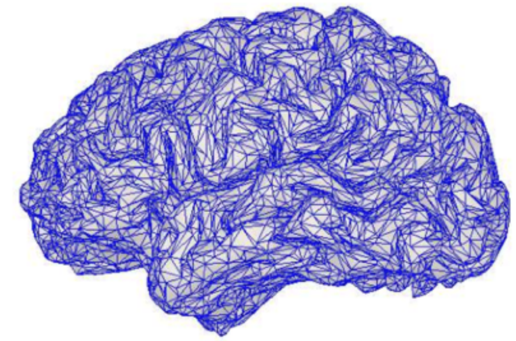
what about convolutional neural networks?

- Traditional matrix multiplication is replaced by convolution in at least one layer
- Convolution *similar* to "flip & shift" but usually no flip
- Excellent for analyzing grid like topology (e.g., images)
- Has receptive fields - like neurobiology
- Parameter sharing causes equivariance to translation
- Usually kernel is smaller than input -> sparse connectivity



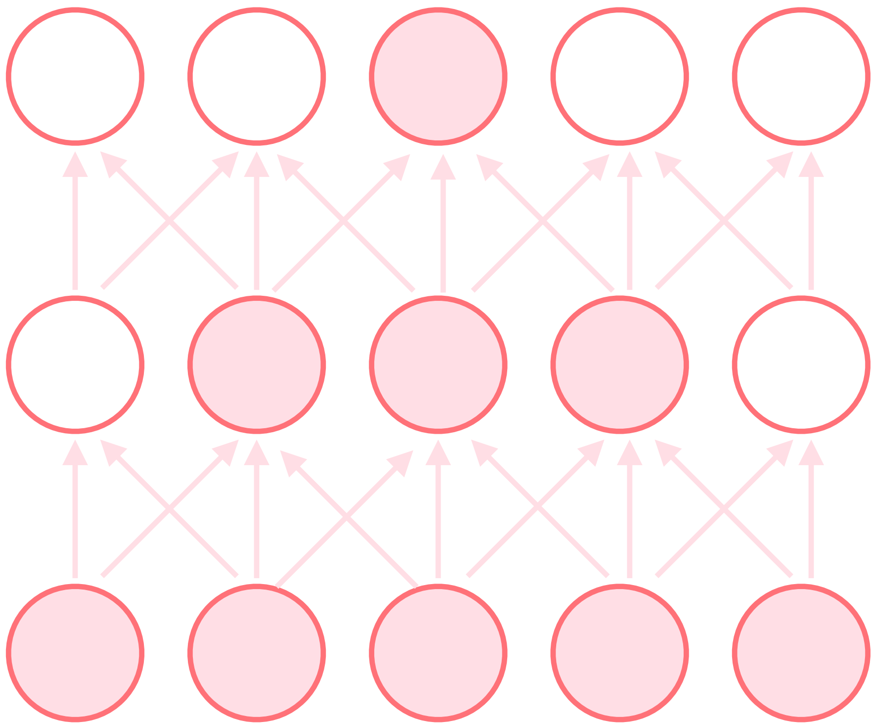
Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

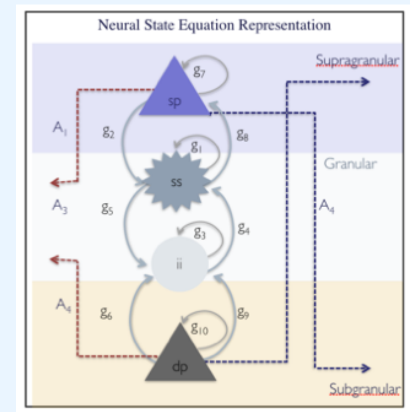
CNNs have receptive fields



Convolutional (Artificial) Neural Networks

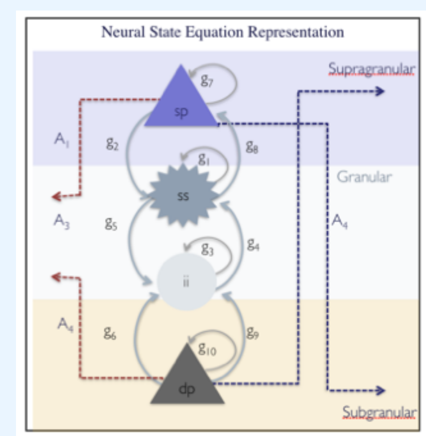
Biological Neural Networks





Higher Levels

larger receptive fields; update slowly; **more representational drift**



Lower Levels (e.g., V1)

have smaller spatial receptive fields; update rapidly; representations more stable

Higher Levels

larger receptive fields; indirectly connected to most of image

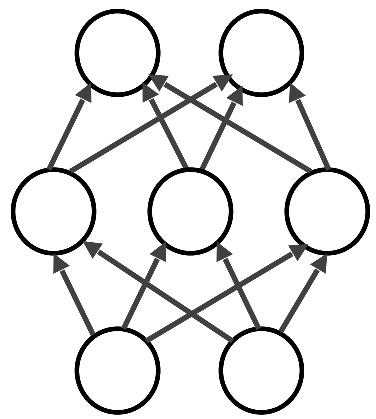
Inspired by deep learning book

Rule et al. 2020; Parr et al. 2017

recurrent neural networks: universal approximators of dynamics

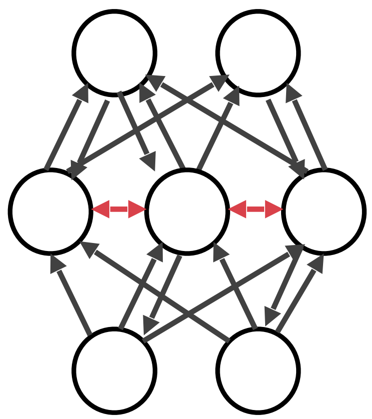
The brain is a deep and complex recurrent neural network.
(Kriegeskorte 2015)

Feedforward



Universal Function
Approximators

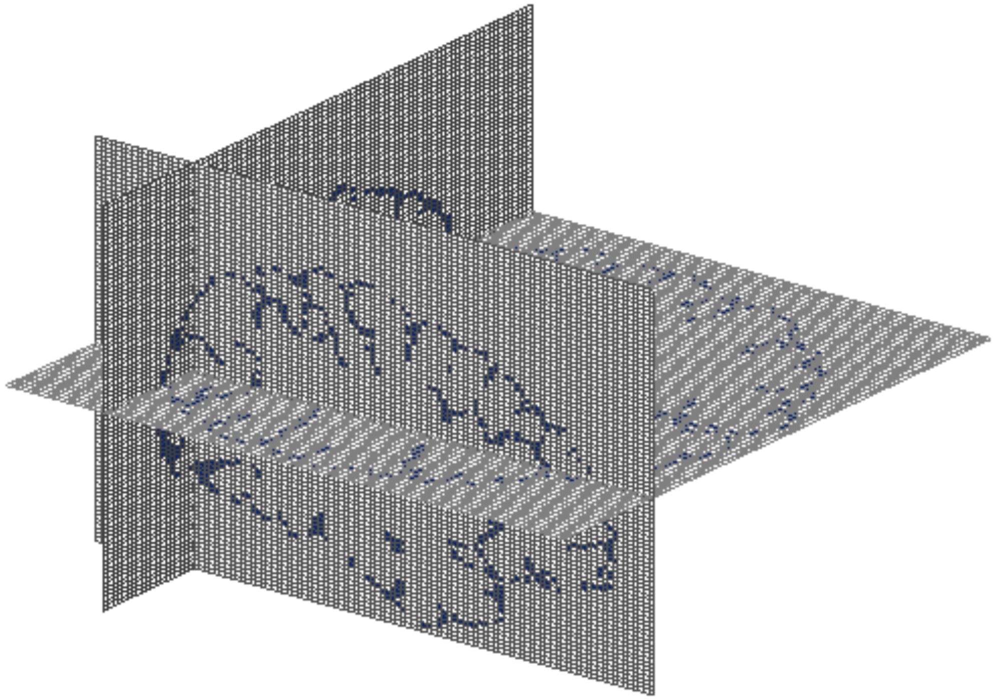
Recurrent



Universal
Approximators of
dynamic systems

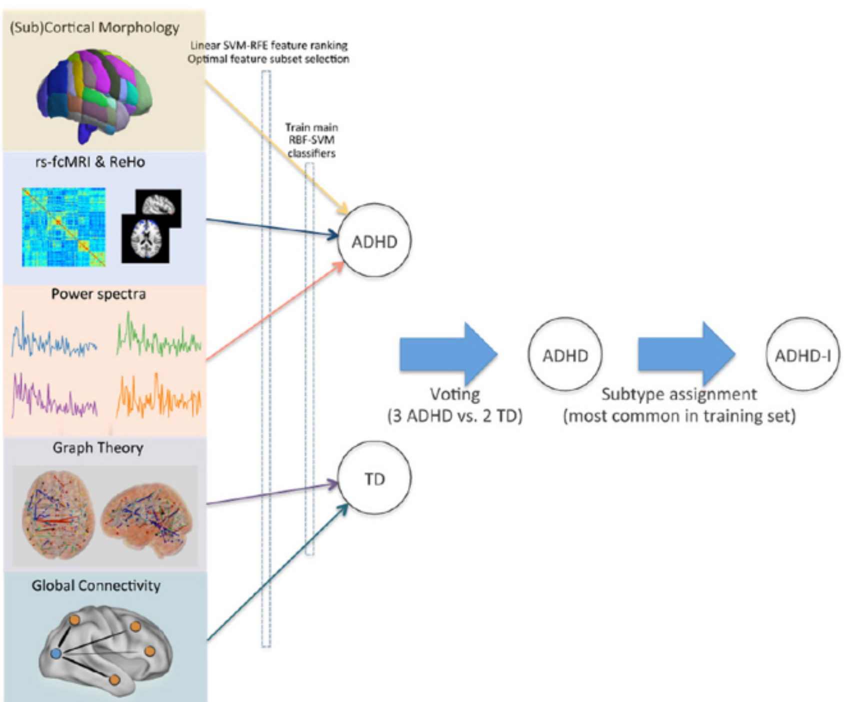
Schäfer & Zimmermann 2007

how are deep learning models useful for neuroimagers?

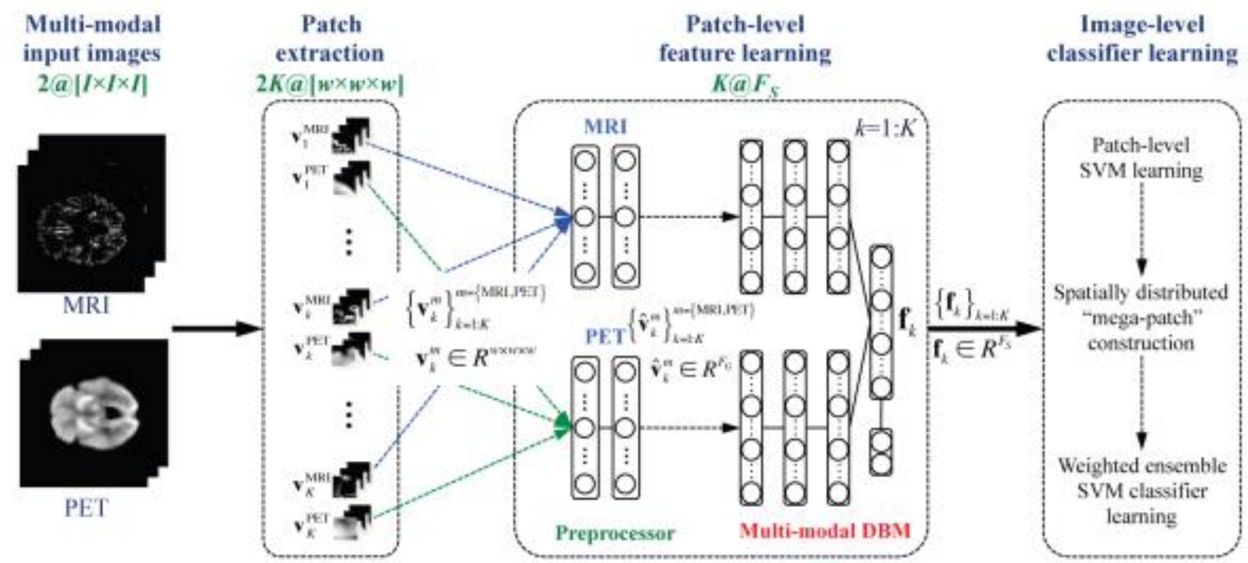


representational models & group membership

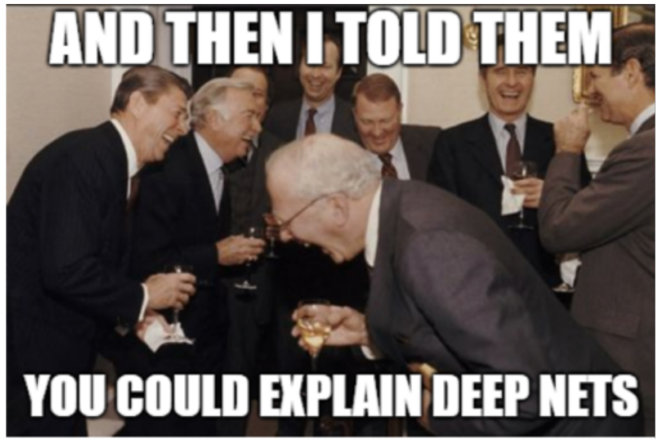
- Local interpretation - rank explanatory power of input features / voxels



MLP for ADHD/TD classification
Colby et al. (2012)

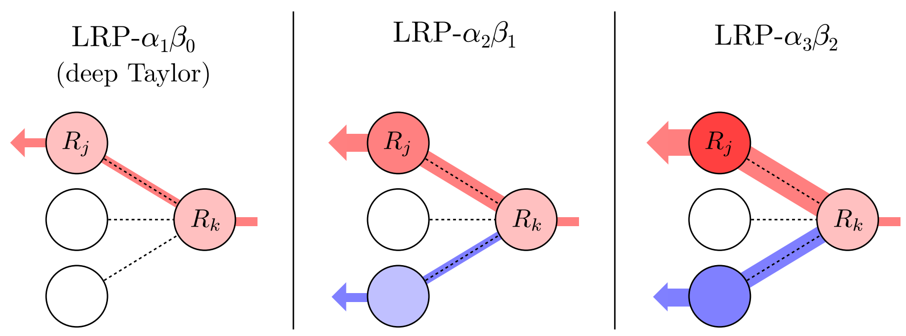


Alzheimers / MCI (Suk et al. 2014)



interpreting relevance maps requires great care

- Many saliency methods exist, but may require tuning hyper parameters or determining appropriate reference points in order to be robust against <u>adversarial</u> or <u>didactic</u> perturbation



Layer wise relevance propagation
(Bach et al. 2015)

"Ground Truth"
Panda

(adversarial) noise
common to MRI Setting

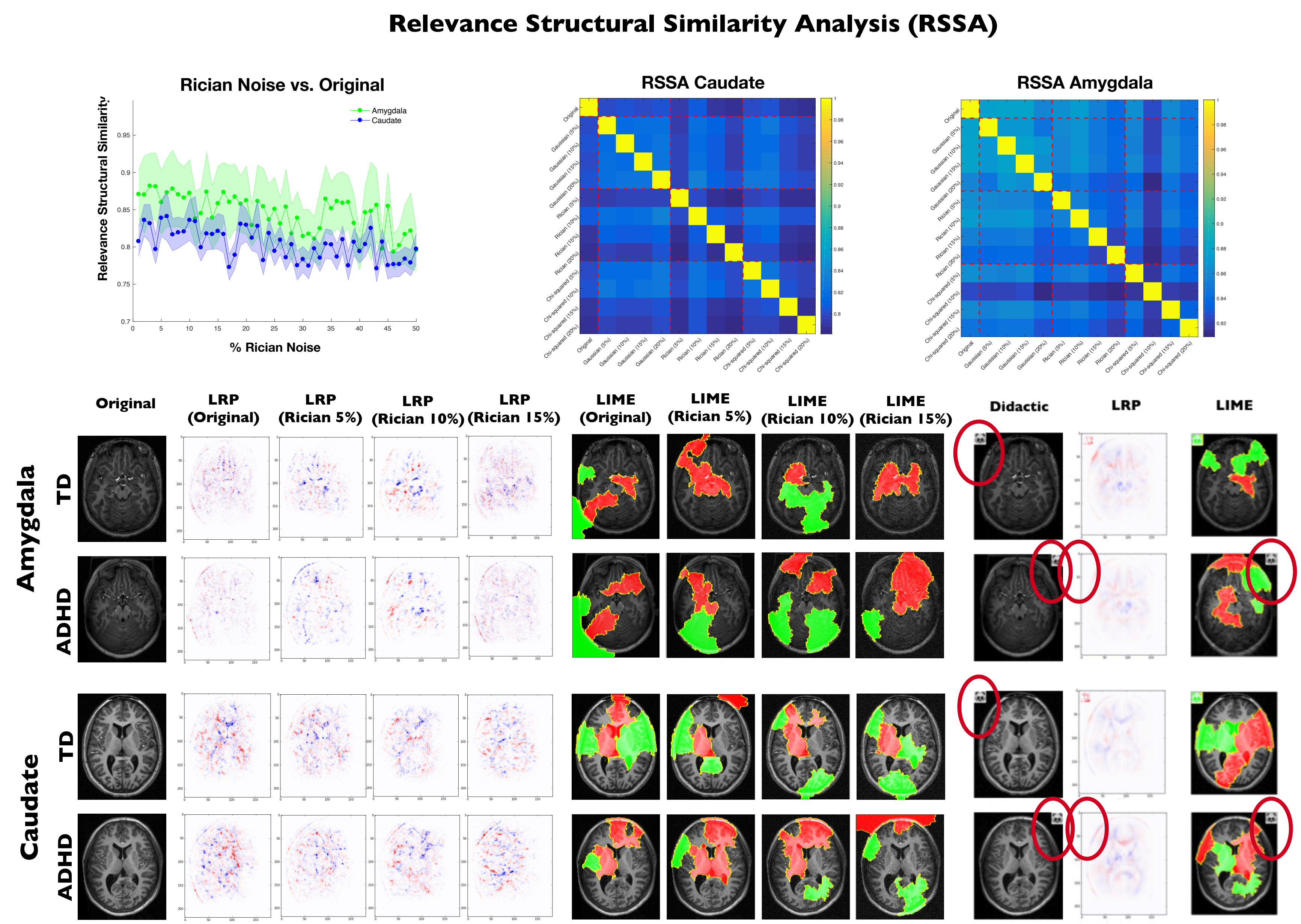
$\tilde{X} = X + \lambda \epsilon_R X$

didactic perturbation

Top right = ADHD
Top Left = TD

interpreting relevance maps requires great care

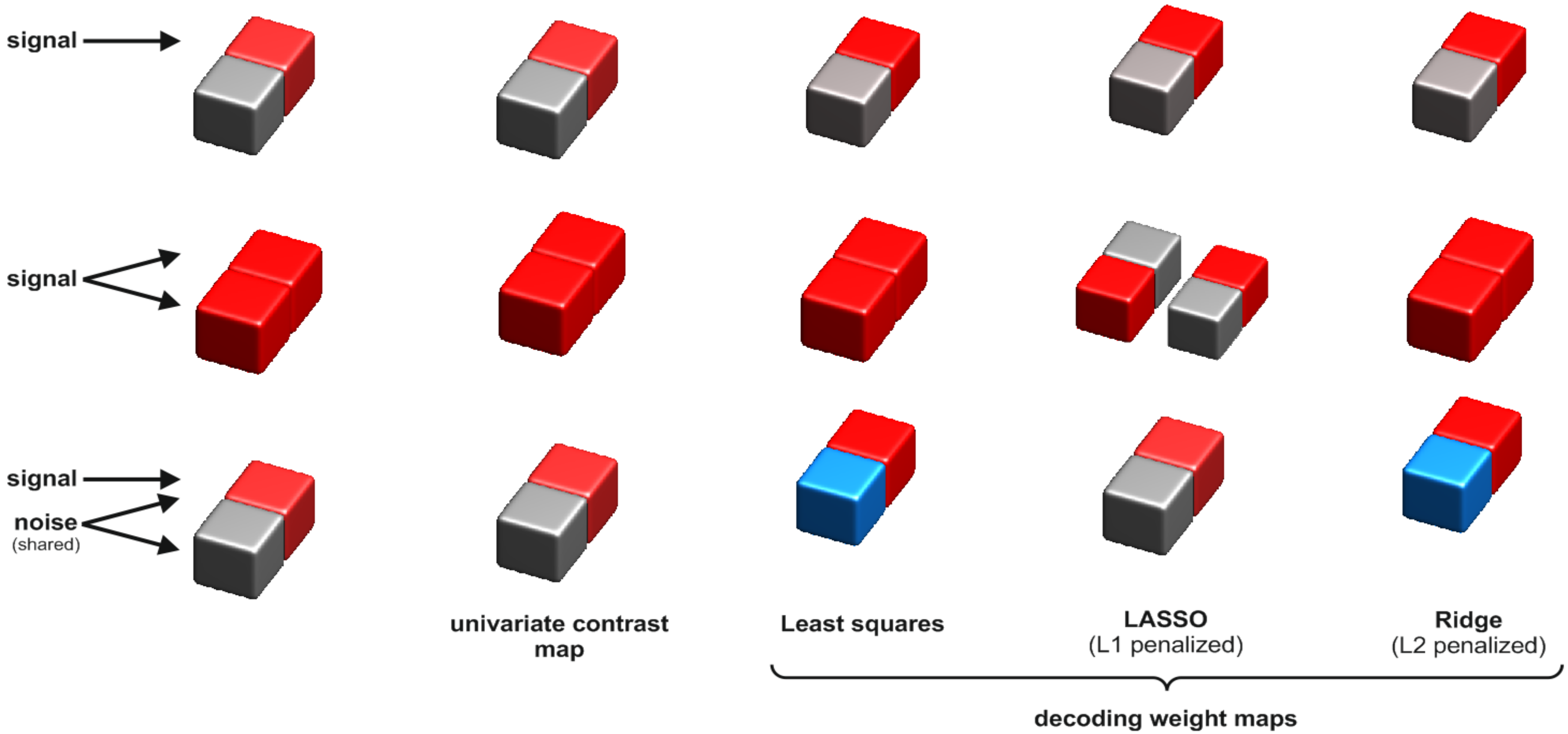
- Many saliency methods exist, but may require tuning hyper parameters or determining appropriate reference points in order to be robust against <u>adversarial</u> or <u>didactic</u> perturbation



Didactic panda was missed :(

Douglas & Farahani (2020) https://arxiv.org/pdf/2002.06816.pdf

L1 regularization in linear models produces a similar effect

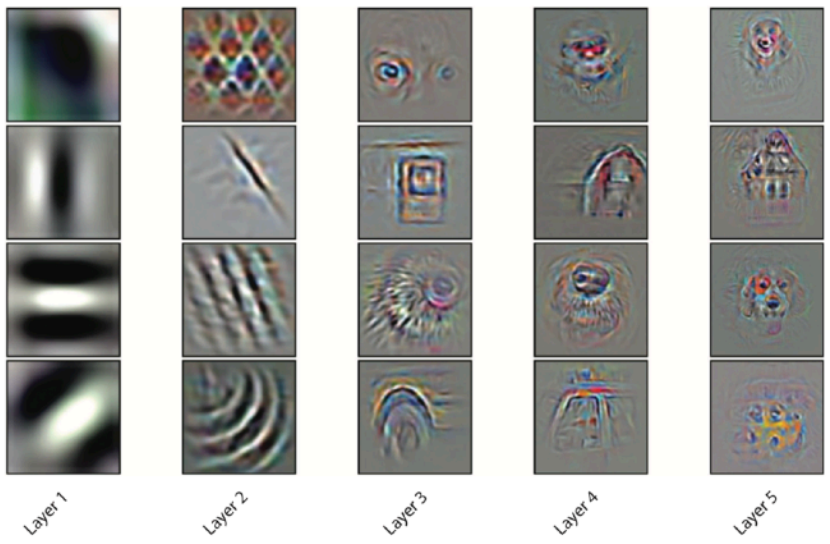


"Voxel selection by L1 penalty on brain maps is unstable because neighboring voxels respond similarly - and L1 estimators will choose somewhat randomly few of these correlated features" - Varoquaux et al. (2016)

Kriegeskorte & Douglas (Curr Opinion 2019) Available here: https://arxiv.org/pdf/1812.00278.pdf

deep learning: brain computational models

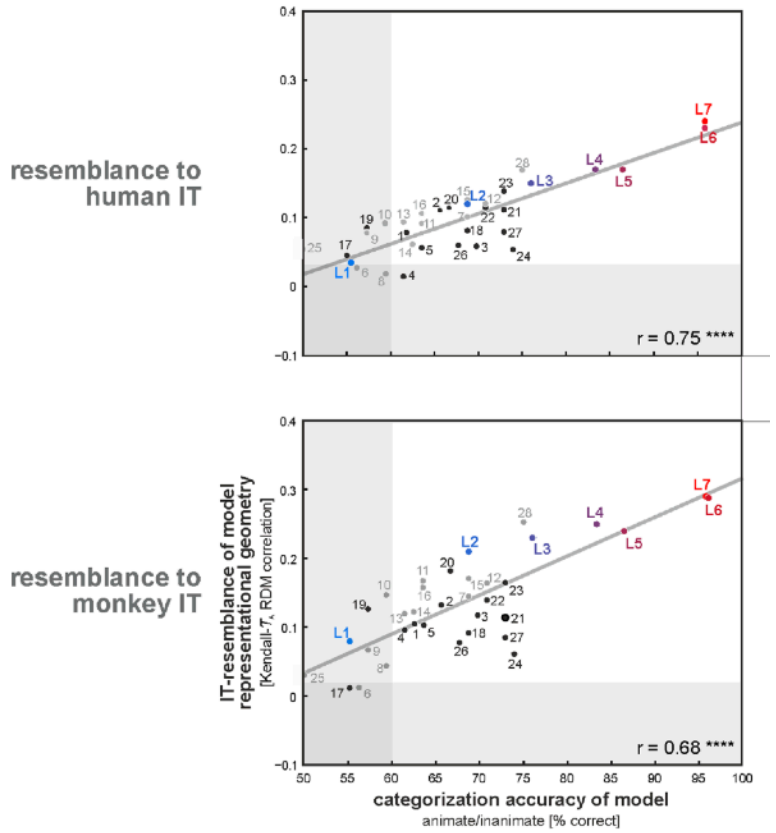
- Functional interpretation



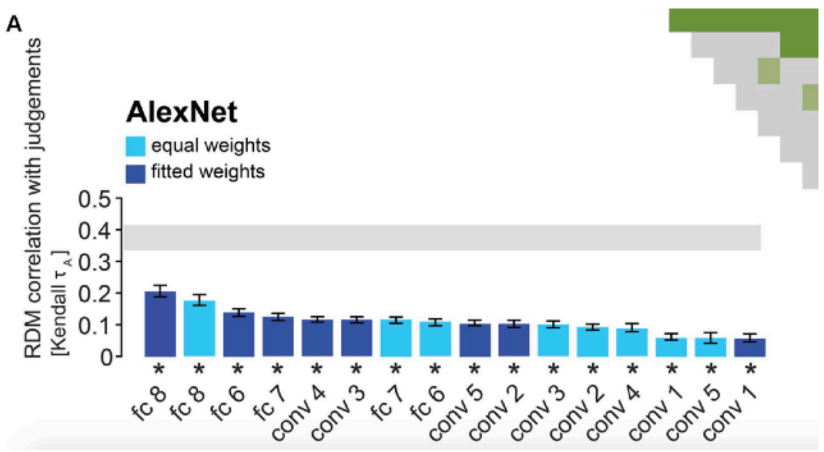
Internal representations are a useful model for representations in visual stream

Guclu & van Gerven (2015)

More representational drift at higher levels (Rule et al. 2019)



Higher levels that resembled IT performed better (Khaligh-Razavi et al. (2014)



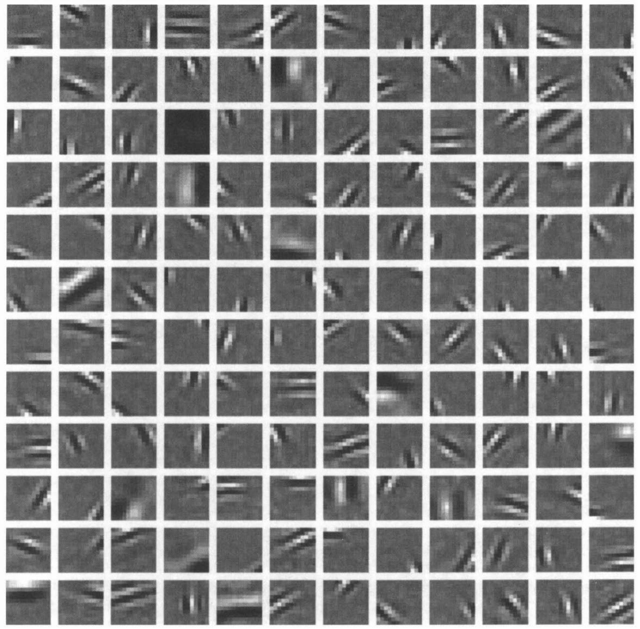
Have been useful in explaining human behavioral judgements about object similarity (Jozwik et al. (2017)

Kriegeskorte & Douglas (2018)

many deep learning models learn "Gabor like" features

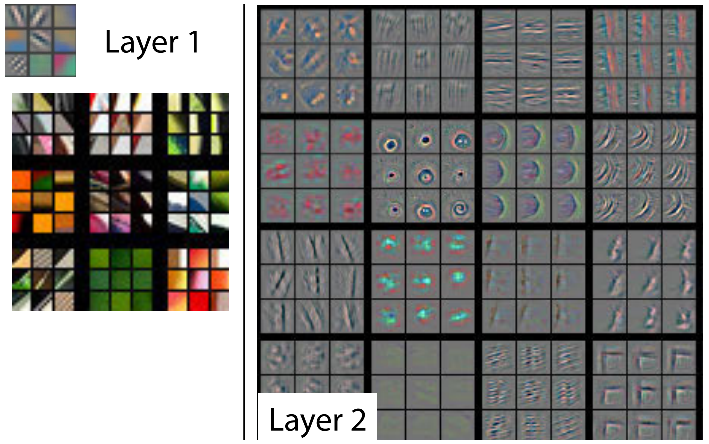
- Functional interpretation

Sparse Coding



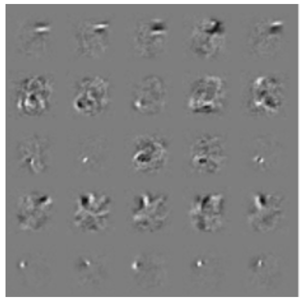
Olshausen & Field (1996)

Visualizing & Understanding
Convolutional Networks

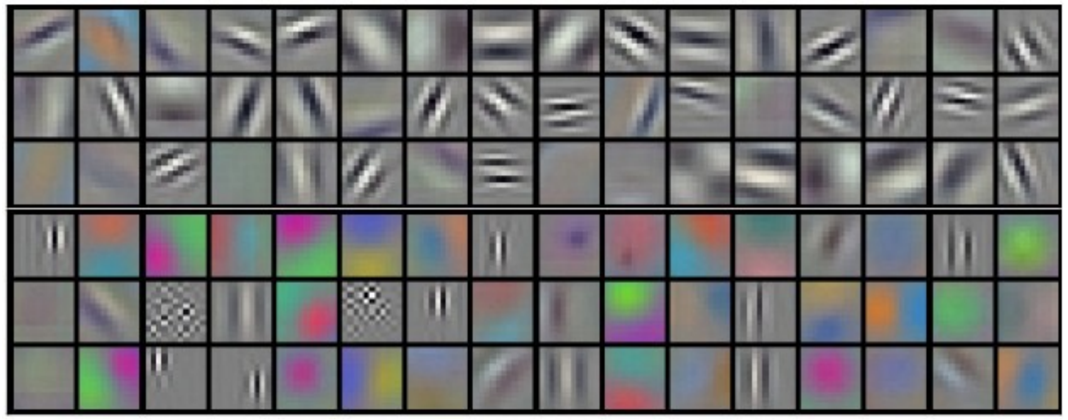


Zeiler & Fergus (2013)

Maxout units



Goodfellow et al. (2015)

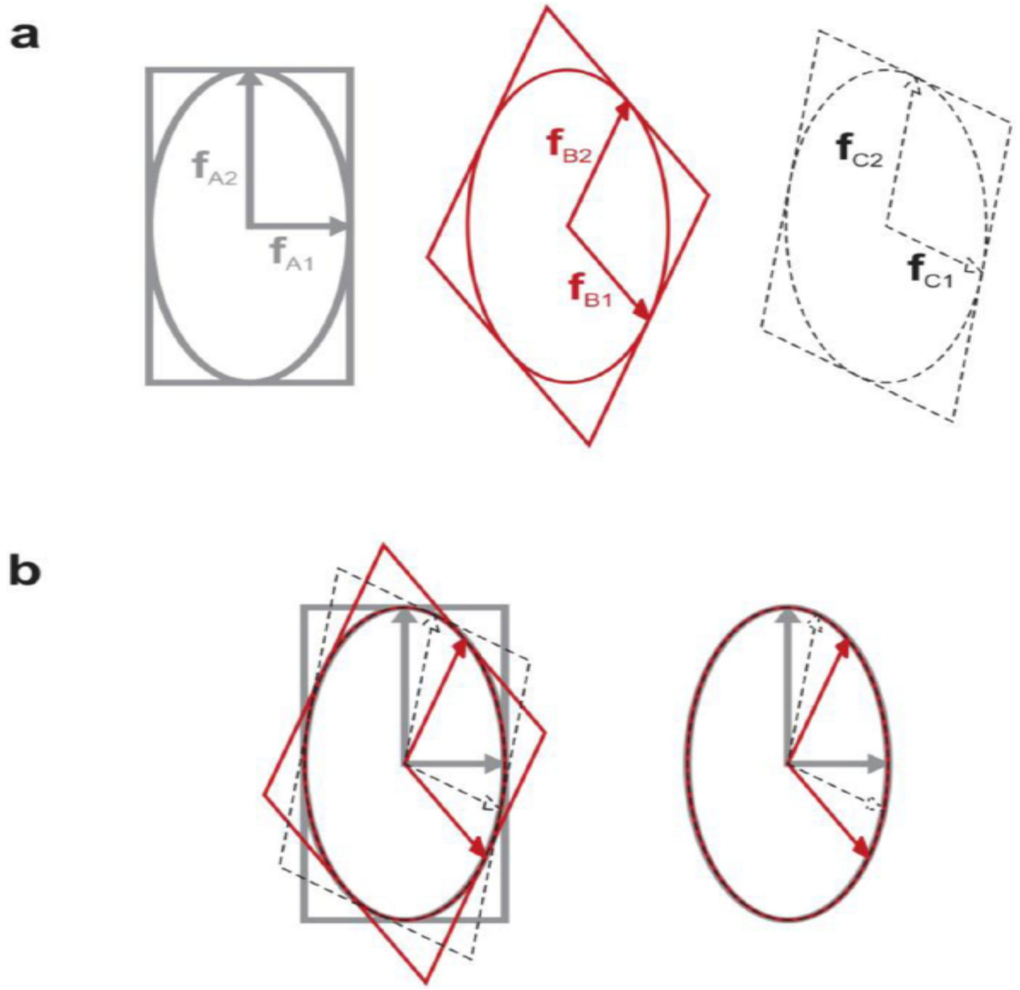


Krizhevsky et al. "ImageNet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.

& many more …

single model fallacy

- *Even a bad model can explain some variance of the data*
- *And sometimes, there are many equivalent "good models"*
- *From a systems ID point of view, this is analogous to an experiment or a model that is non-uniquely identifiable, because multiple parameter combinations work equally well*
- *Interpreting that a single models explains significant variance as evidence in favor of that model is the "Single Model Fallacy"*



Kriegeskorte & Douglas (2019) Current Opinion in Neurobiology

ingredients for deep learning

1.) ~~Model Architecture~~. Multiple Candidate Models

2.) Cost Function

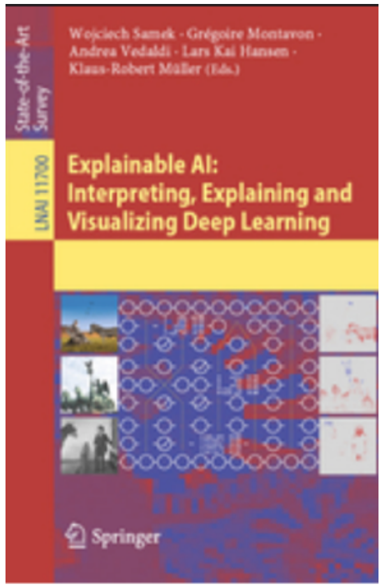
3.) Optimization Procedure

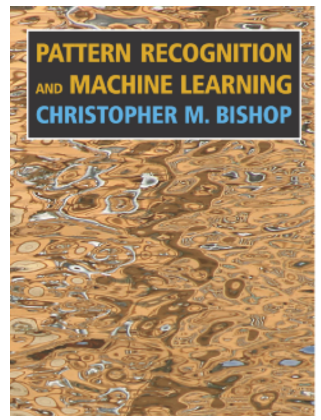
4.) Data

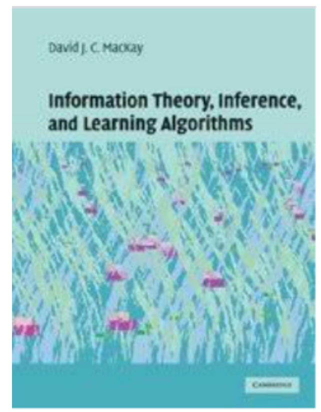
conclusions

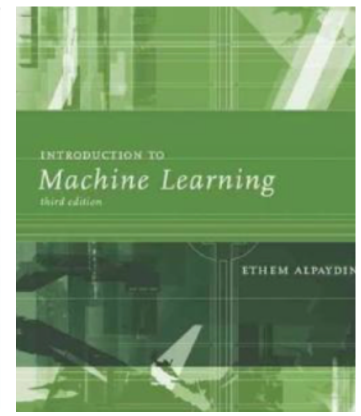
- Like brains, deep neural networks have feedforward and recurrent connections, and can have receptive fields, and many parameters (intelligence systems require sufficient parametric complexity)
- Deep learning can be used for representational models (encoding /decoding) It may be used for group membership prediction, and decoding studies
- Deep learning models provide some of the best current models for internal representations and modeling brain information processing
- Great care should be used when utilizing saliency methods to ensure they are robust to perturbations. (We still lack a ground truth for these methods.)
- To avoid the single model fallacy, multiple models should be tested, and they should be evaluated in terms of the level of generalization they achieve (same data held out, new measurement - same individual, new individuals, new stimuli / tasks, etc)

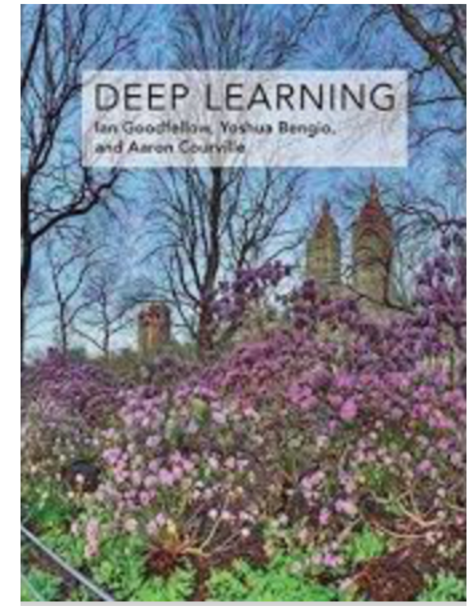
resources











- Deep Learning Book : Freely available online
- https://www.deeplearningbook.org

resources

- OHBM Full Course on Deep Learning (videos, notebooks, slides)

  https://brainhack101.github.io/IntroDL/

- LeCun course on deep learning

  https://cilvr.nyu.edu/doku.php?id=deeplearning2017:schedule

- WEKA MOOC

  https://www.cs.waikato.ac.nz/ml/weka/mooc/dataminingwithweka/

- Reinforcement Learning (D. Silver)

  https://www.youtube.com/watch?v=2pWv7GOvuf0

- Nice primer on deep learning for neuroscience (Kriegeskorte 2015):

  https://www.biorxiv.org/content/biorxiv/early/2015/10/26/029876.full.pdf

thanks

NIMH
National Institute of Mental Health

BRAIN & BEHAVIOR
RESEARCH FOUNDATION
Awarding **NARSAD** Grants

AIR FORCE RESEARCH LABORATORY

Many thanks to :

Niko Kriegeskorte,

Ariana Anderson,

Klaus-Robert Muller,

Gael Varoquaux,

Alex Gramfort

Alex Binder

Leo Christov Moore

Jeiran Chopan

Farzad Vasheghani Farahani

Paul Thompson

Dan Moyer

Susan Bookheimer

@pkdouglas16