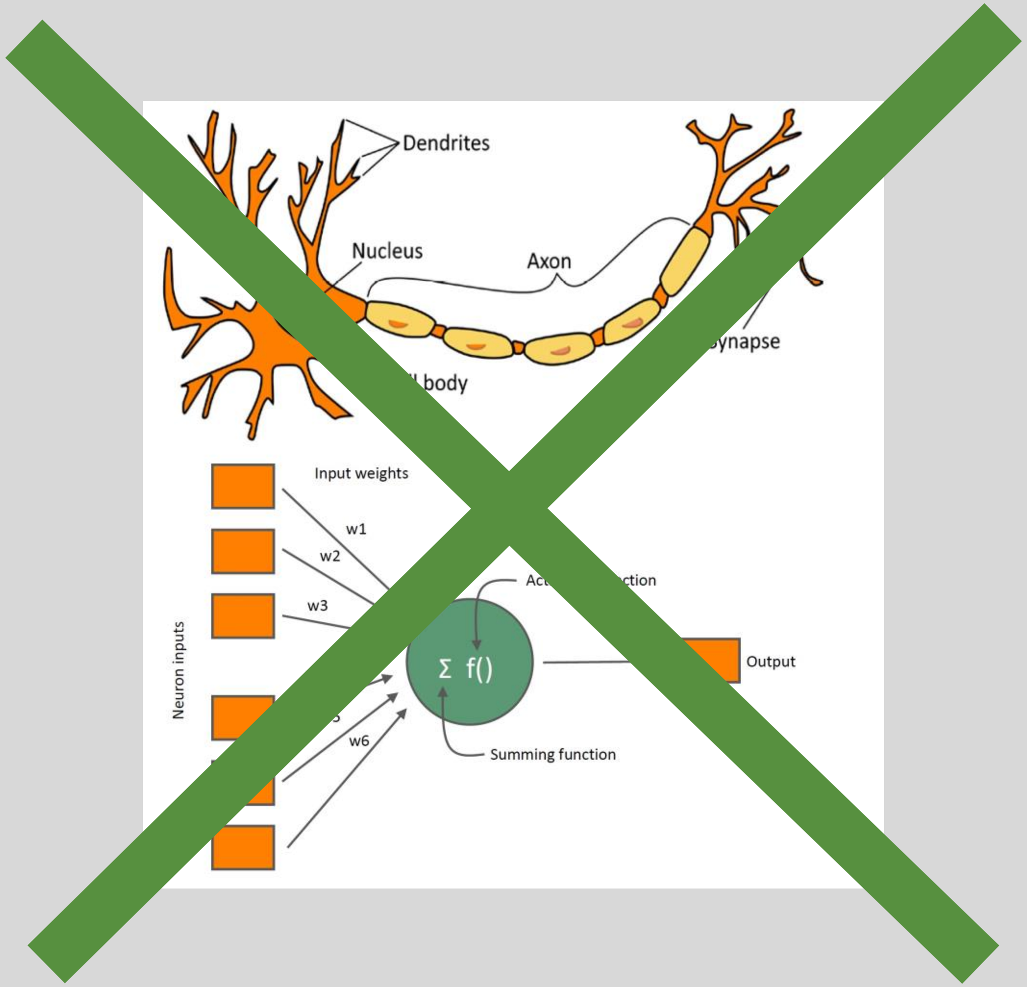
NEURAL NETWORKS INTRODUCTION

Paul Speaker

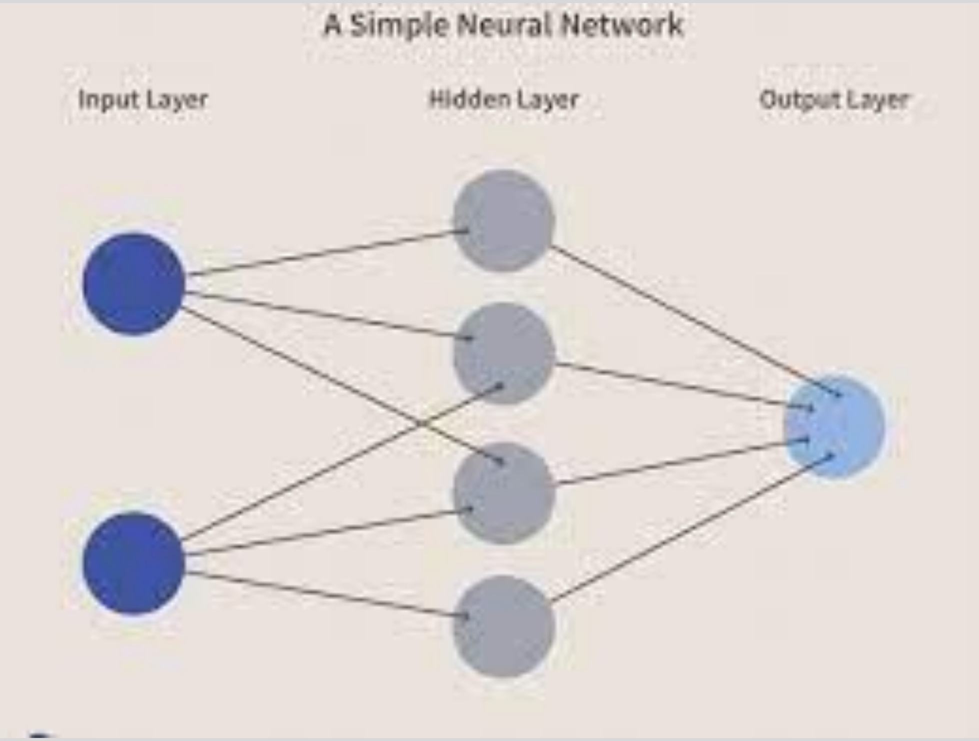
“Neural” Networks

- No area of data science is more subject to hyperbole than neural networks
- Common myths
  - Neural networks are like the human brain
  - Neural networks learn from themselves
  - Neural networks are black boxes



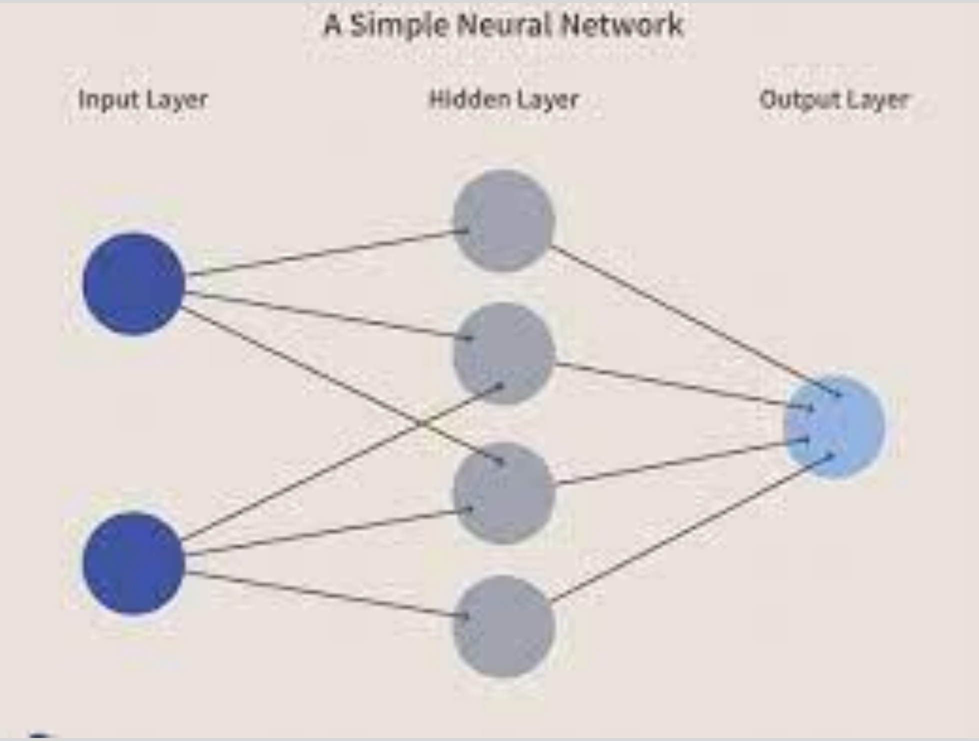
Basic Ideas of Neural Networks

- There are 3 ways that a neural network is typically characterized
  - Network architecture
    - Connections and arrows in diagram
  - Activation functions
    - Functions at each layer of previous layer
    - Since each layer goes into the activation function of the next layer, the primary mathematical concept is **composition**
    - **Composition**: f(g(x))
  - Optimization process
    - Because neural networks are nonlinear, the optimization process is a crucial way to understand the model



Basic Ideas of Neural Networks

- What is missing from this list
- Data preparation—the real "hidden layer" of neural networks
- Some of the data preparation is standard across ML
  - Cleaning (handling missings, etc.)
  - Rebalancing data
- What is often different for NN's
  - Feature extraction
  - Label identification
  - More on these next time!



Activation Functions

- Each layer looks like an activation function applied to a linear combination of the previous layer
  - Linear combination: $L = a + bx_1 + cx_2 + \ldots$
- Most common activation functions
  - ReLU—Rectified Linear Unit:
    - Functional form: max(0, L)
  - Logit: same as the logistic function in logistic regression
    - Function form: $1/(1 + e^{-L})$
  - Softplus
    - Function form: $\log(1 + e^{L})$
  - Hyperbolic tangent
- Structure is true for numerical or categorical target, but….
  - For numeric target, activation function is just linear combination of last layer

Logistic Regression as a Neural Network

- The number of hidden layers can be from 0 to ???
- Think about a neural network with
  - No hidden layers
  - Logit activation function
- The neural network the result will be a linear combination of the raw inputs, put into a logit function.
- The coefficients of the linear combinations are then optimized for
- This is exactly a logistic regression

When to Use Neural Networks

- If neural networks are so great, why aren't they used everywhere?
- There are two main classes of problems that neural networks are particularly good at solving
  - Image classification
  - Text generation
- For other predictive problems, other predictive models perform as well or better
  - Xgboost is typically better
  - Even simpler models can perform as well for less computational and productionization cost
- Why?
  - Hidden layers function to extract better features
  - For data dataframe, data is already in logical structure
  - For example, in image classification intermediate data features are crucial
  - Does not mean that neural networks are not worthwhile for tabular data (more later!)

Optimization Methods For Neural Networks

- Most methods for optimizing a neural network utilize back-propagation
- Basic method for neural networks
  - Initialize weights with initial guess
  - Calculate loss function (usually RMSE)
  - Optimize parameters for 1 layer back, keeping earlier values fixed
  - Repeat until you get to first hidden layer
  - Once layers values are optimized, we determine values through each stage
    - forward propagation
  - Repeat
- The optimization steps are typically done through gradient descent, an iterative approach to optimization

Neural Networks in R

- We will use two different packages
  - neuralnet to get started
  - keras for building neural networks on more complicated data
- Neuralnet syntax similar to most modeling methods in R
- Keras syntax will be more similar to knn (input data matrix, etc.)
- We will only use neuralnet today
- Syntax: neuralnet(<model formula>, data = <dataframe>, act.fct = <activation function, hidden = c(<vector describing hidden layers))
- The activation function can be customized
- hidden option: c(2, 4) will have 2 hidden layers, with 2 nodes in first layer and 4 in second