STUTERN

Artificial Neural Network

SGA07_DATASCI

10th March 2020

Module Overview

- Recap on Data Mining Techniques
- Review of Biological Neural Networks
- Artificial Neural Network Topologies

STUTERN

Book Keeping

- Group task: Reviews by 27th March 2020
- Early Live Sessions: 12 - 2pm

Outcome

After this Module, you will;

- Take a look back at the data mining techniques covered so far…
- Explore biological neural network - connectionist computing approach
- Gain intuition on the architecture and topology of artificial neural networks

DM Techniques

- Statistical
  - Regression (Linear, Multivariate & Logistic)
- Rule-based
  - Concept Learning
  - Decision Tree
- Probabilistic
  - Naive Bayes

Artificial Neural Networks (Def.)

> An artificial neural network is defined as a data processing system consisting of a large number of simple highly interconnected processing elements (artificial neurones) in an architecture inspired by the structure of the cerebral cortex of the brain.
> (Tsoukalas and Uhring, 1997)

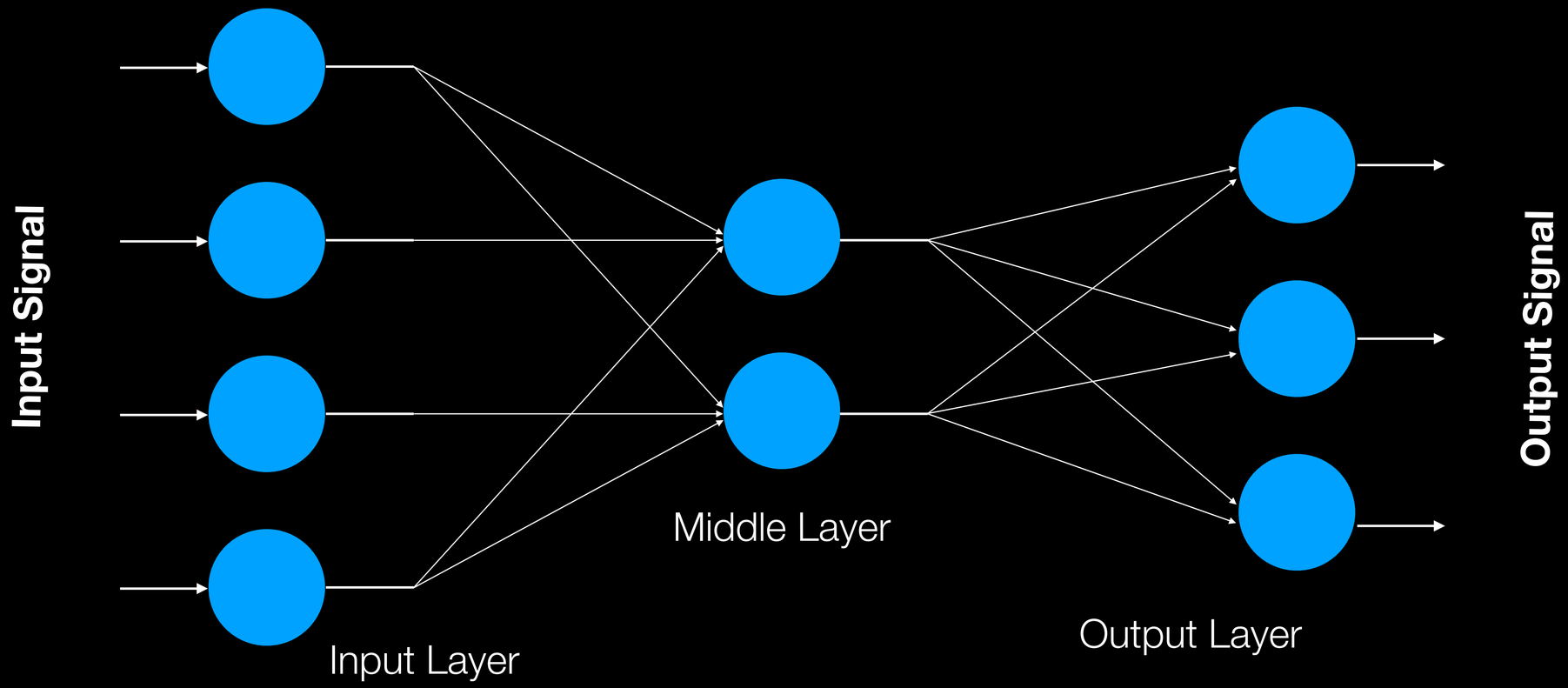
Biological Neural Network

- Brain consists of a densely interconnected set of nerve cells, or basic information-processing units, neurones.
- Human brain incorporates 10 billion neurones and 60 trillion interconnections.
- By using multiple neurones simultaneously, brain can perform its functions much faster than fastest computers.
- Information stored and processed simultaneously throughout whole network, rather than at specific locations
- Learning fundamental characteristic of biological neural networks - Connectionist Computing

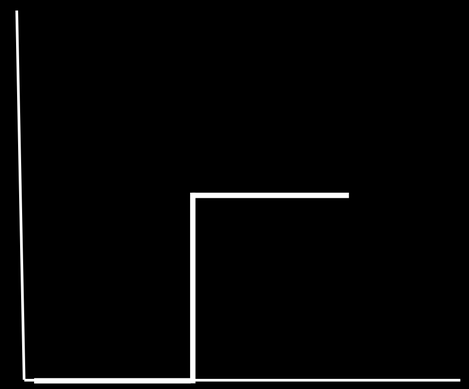
Neural Network Architecture

- Generally, an ANN structure can be represented using a directed graph. A graph G is an ordered 2-tuple (V, E) consisting of a set V of vertices and a set E of edge.
- When each edge is assigned an orientation, the graph is directed and is called a directed graph or digraph.
- There are several classes of NN. Classified according to their learning mechanisms. However we identify 3 fundamentally different classes of Networks.
  - Single layer feedforward network
  - Multilayer feedforward network
  - Recurrent network
- All the three classes employ the digraph structure for their representation.

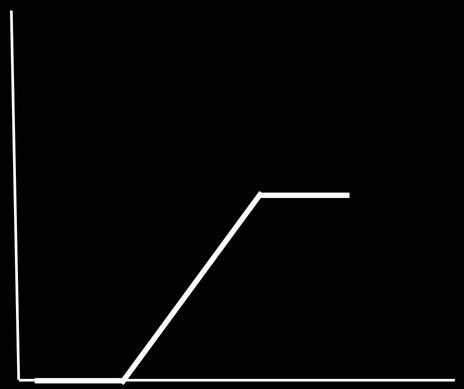
ANN with Hidden Layer



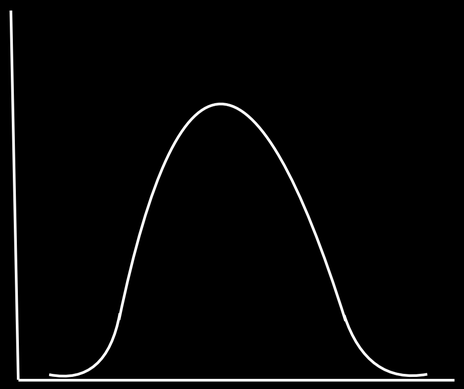
Types of Activation Function



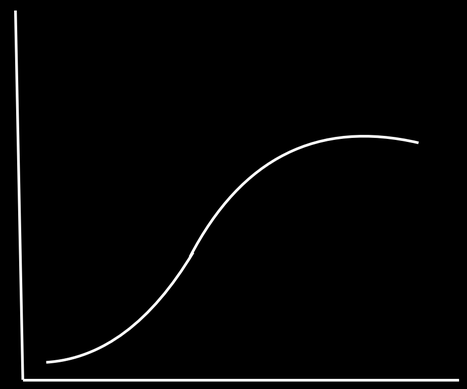
Threshold



Linear

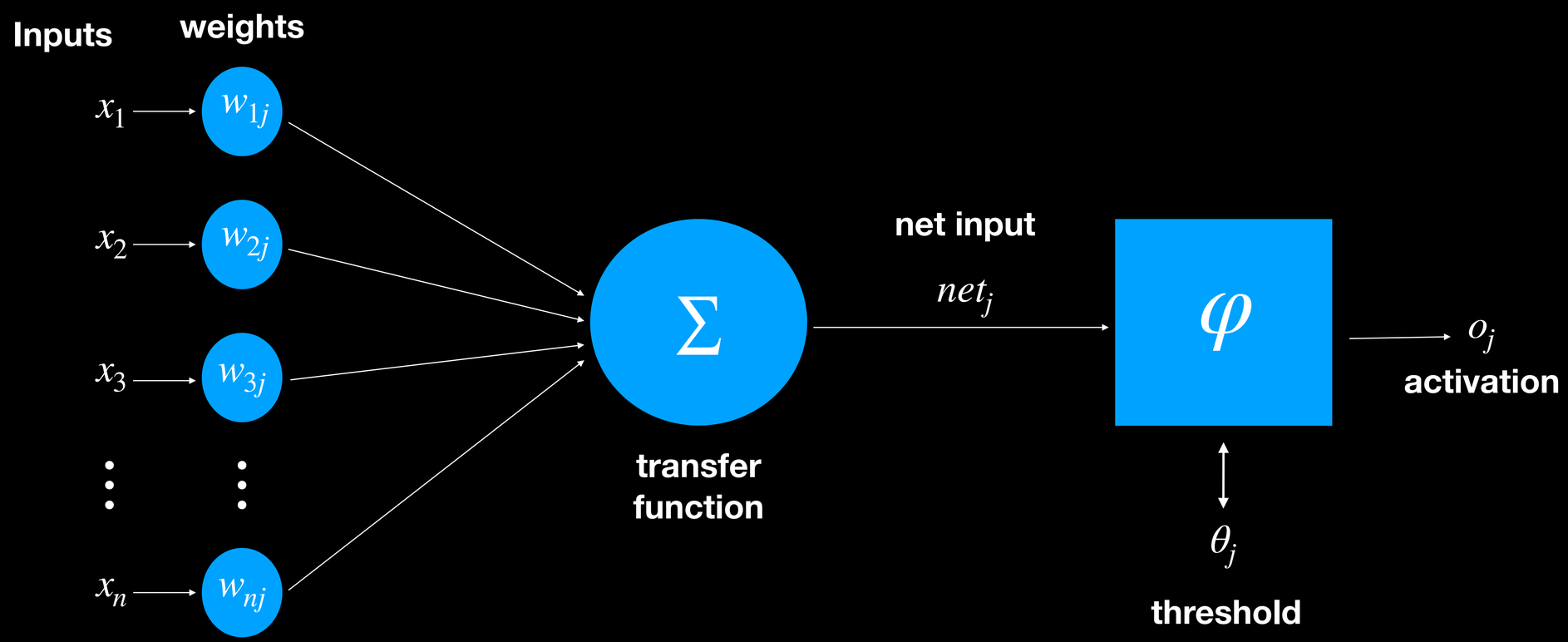


Guasian



Signoid

Single Layer ANN



Single Layer Feedforward Network

- This type of network comprises of two layers, namely the input layer and the output layer.
- The input layer neurones receive the input signals and the output layer neurones receive the output signals.
- The synaptic links carrying the weights connect every input neurone to the output neurone but not vise – versa.
- Such a network is said to be feedforward in type or acyclic in nature.
- Despite the two layers, the network is termed single layer since it is the output layer, alone which performs computation.
- The input layer merely transmits the signals to the output layer.
- Hence the name single layer feedforward network.

Neuron Training Rule

- Initialise weights to random values.
- Making small adjustments in weights to reduce difference between actual and desired outputs.
- Incrementally update weights to obtain output consistent with training examples.
- For training instance d and target output t_d, error given by:

$$E_d = t_d - o_d$$

$$d = x_{1d} + x_{2d} + \cdots + x_{nd}$$

Neuron Training Rule

- If E_d +ve, need to increase neuron output o_d
- If E_d -ve, need to decrease o_d.

$$w_i \longleftarrow w_i + \triangle w_i$$

$$\triangle w_i = \eta(t_d - o_d)x_i$$

- Learning rate, η, +ve constant less than 1.
- Minimise E_d until Convergence

Convergence Theorem

- If there exists set of connection weights w able to perform function i.e. target function linearly separable:

$$o = f(\sum_{i=1}^{2} w_i x_i - \theta)$$

- Then training rule guaranteed to converge to solution in finite number of steps for any initial choice of weights.
- If target function is not linearly separable, then use least Mean Square error and gradient descent to find weight

Gradient Descent Algorithm

- Initialise each w_i to some random value
  - Initialise each Δw_i to 0
  - For each instance $\vec{X}$ and t in training set D
    - For each wi

      $\Delta w_i \leftarrow \Delta w_i + \eta(t - o)x_i$
  - For each w_i

    $w_i \leftarrow w_i + \Delta w_i$
  - Repeat until convergence

Multilayer Feedforward Network

- This network, as its name indicates is made up of multiple layers.
- Architecture of this class besides possessing an input and output layer also have one or more intermediary layers called hidden layers.
- The computational units of the hidden layer are known as hidden neurons or hidden units.
- The hidden layer aids in performing useful intermediary computation before directing the input to the output layer.
- The input layer neurons are linked to the hidden layer neurons and the weights on these links are referred to as input hidden layer weights.
- Again , the hidden layer neurons are linked to the output layer neurons and the corresponding weights are referred to as hidden-output layer weights.

Back-propagation

- Training procedure which allows multi-layer ANNs to be trained.
- Can theoretically perform any input-output function.
- With appropriate choice of units, multi-layer ANNs can learn to solve linearly inseparable problems.

Strength of ANN

- High tolerance to noisy data
- Well-suited for continuous-valued inputs and outputs
- Successful on real-world data e.g. hand- written text recognition
- Inherently parallel – using network after training fast

Recap/Summary

At the end of this Module, you should understand;

- Take a look back at the data mining techniques covered so far…
- Explore biological neural network - connectionist computing approach
- Gain intuition on the architecture and topology of artificial neural networks

Suggested Material

- Machine Learning by Tom Mitchell Chapter 4
- http://colah.github.io/posts/2014-03-NN-Manifolds-Topology/
- https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464
- https://www.youtube.com/watch?v=GqfzCTpCODE