An Introduction to neural network

Motivation

- Brain computation compared to digital computers
  - Neurons are 5-6 orders of magnitude slower than digital logic (ms vs. ns)
  - Brain has a huge number of neurons (10 billion neurons and about 60 000 billions of interconnections)
  - Brain is enormously energy efficient (10^{-16} J per operation per second vs 10^{-6} J per operation per second)
  - Brain is a very complex, nonlinear, parallel computer

Examples of efficacy

Human vision as information processing problem

- Human recognizes faces in 100-200 ms, computes need more time

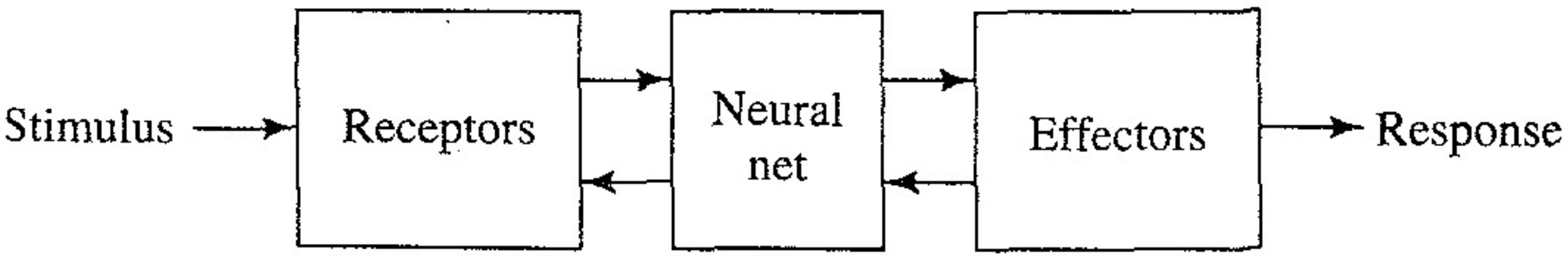
Sonar of a bat

- lt Detects distance to object (e.g. Insect), velocity, size, azimouth and elevation
- All operations are done in a plum-sized brain
- Bat can accurately locate and catch insects

How is this possible?

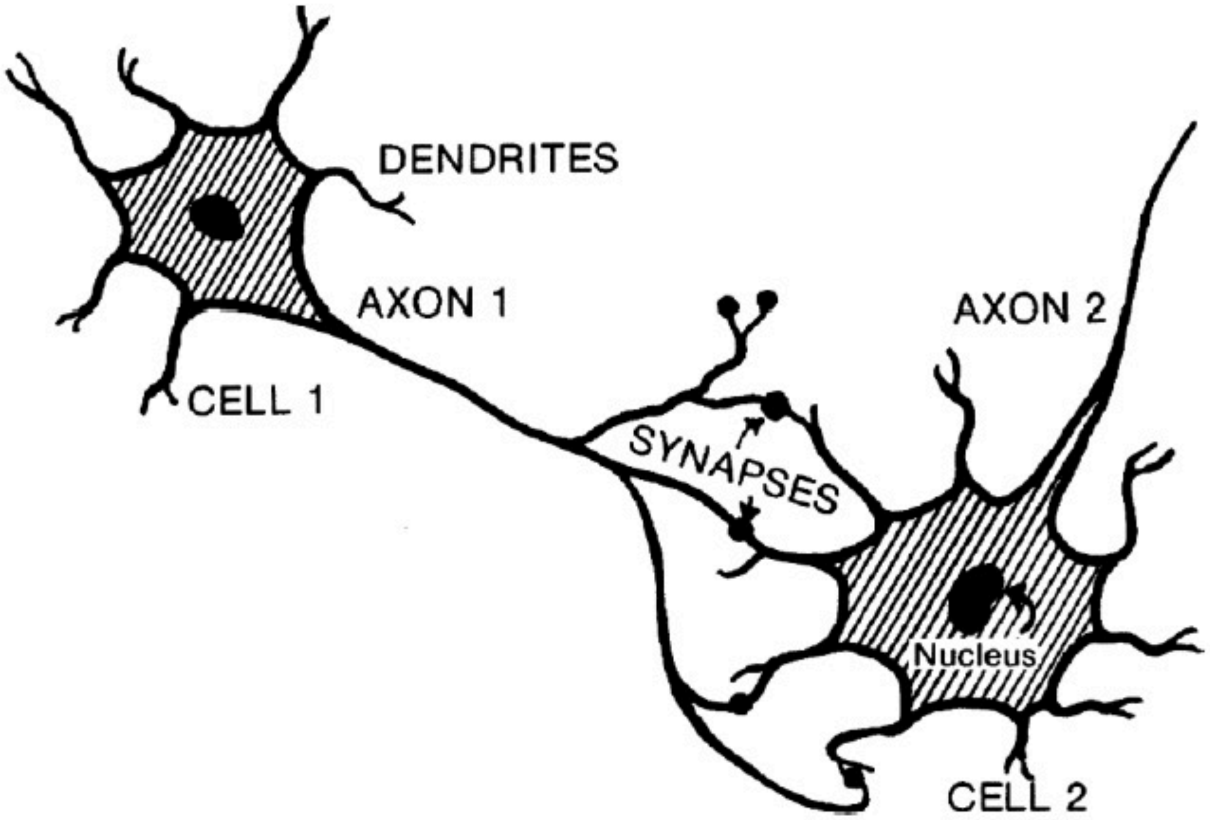
- After birth brain develops very fast through learning
- Knowledge is gained through years of learning
  - Fastest brain development through first two years of life (1 million synapses formed per second)
  - Brain development continues after the initial phase

Block Diagram of Nervous System



Biological Neuron

- Dendrites accept inputs from other neurons
- Axon transmits impulses to other neurons
- Synapses are structures where impulses are transferred from one neuron to another
- Synapses connect neurons to pass electrical signals from one neuron to another
- Biological neuron can have:
- 10000 and more of input synapses
- Its output can be connected to 1000s of other neurons



Neural Networks

Biological neural networks

- Biological organisms
- Human and animal brains
- High complexity and parallelism

Artificial neural networks

- Motivated by biological neural networks
- Much simpler and primitive compared to biological
- networks
- Implementation on general purpose digital computers or using specialized hardware

ANN

- ANN is a massive parallel distributed processor that is good for memorizing of knowledge

- ANN is similar to biological NN in the following aspects:
  - Knowledge is gained through learning process
  - Knowledge is encoded in mutual connections between neurons

ANN properties

- Nonlinearity
- Input to output mapping (supervised learning)
- Adaptivity
- Evidential Response
- Contextual Information
- Fault tolerance
- Possible VLSI implementation
- Neurobiological analogy

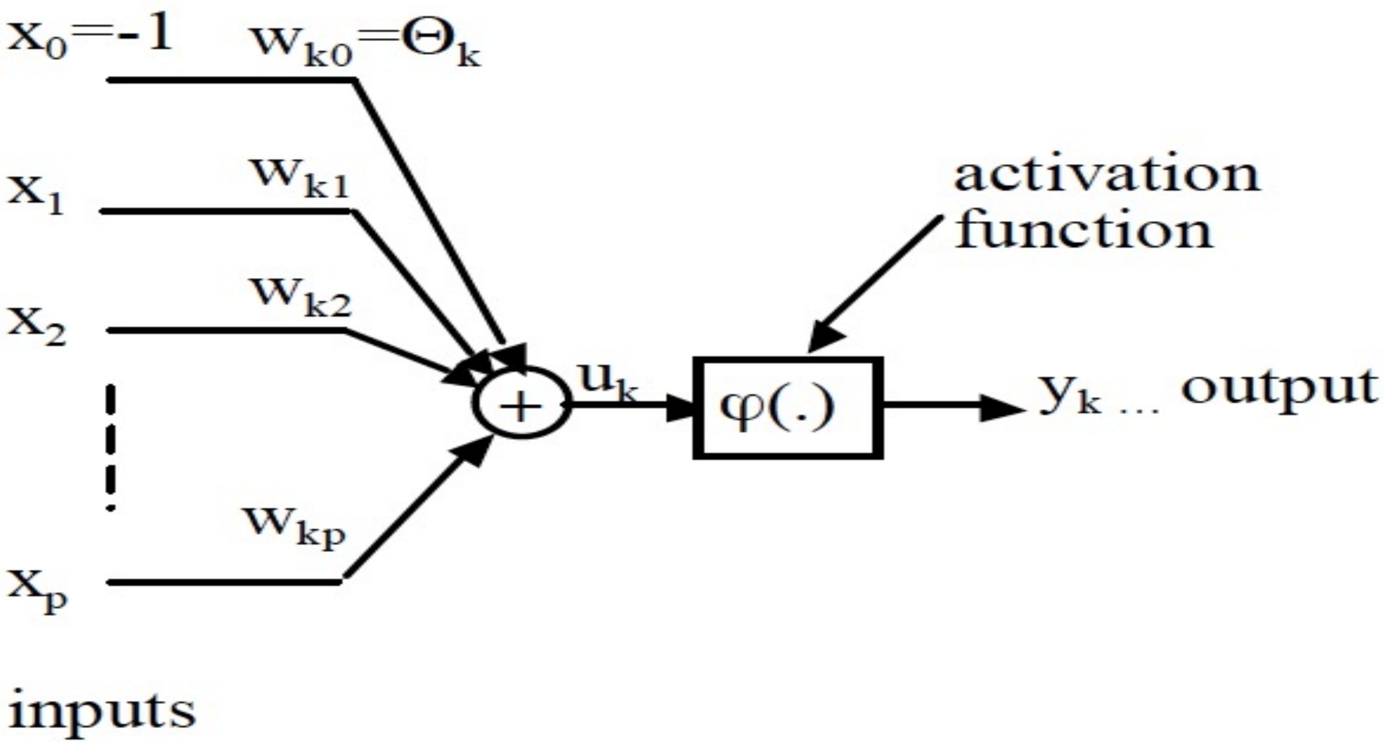
Neuron models

Neuron model elements:

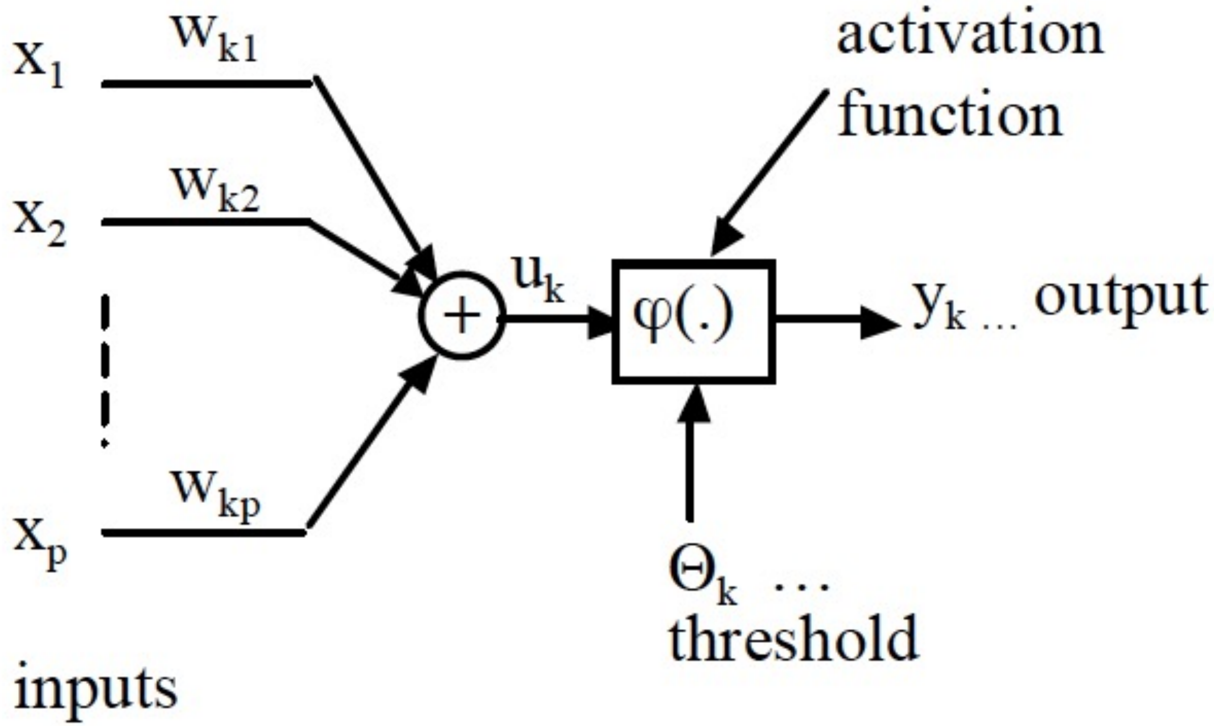
- Set of synapses, i.e. Inputs with respective weights.
- (Notation: Signal x_j at input j of neuron k has weight w_{kj})
- Adder for summation of weighted inputs. These two operations calculate the weighed sum of inputs.
- Non-linear activation function that limits output of neuron to
- interval [0,1]

Neuron models

- Threshold Θ_k can be represented as an additional input of value -1 and weight Θ_k



Neuron models



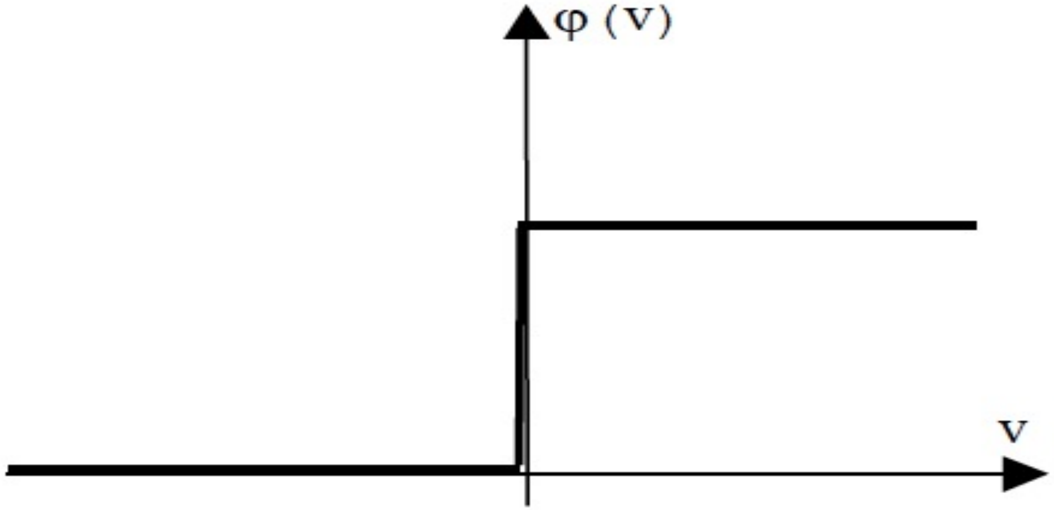
$$u_k = \sum_{j=1}^{p} w_{kj} x_j$$

$$y_k = \varphi\left(u_k - \Theta_k\right)$$

Activation functions

- Type threshold:

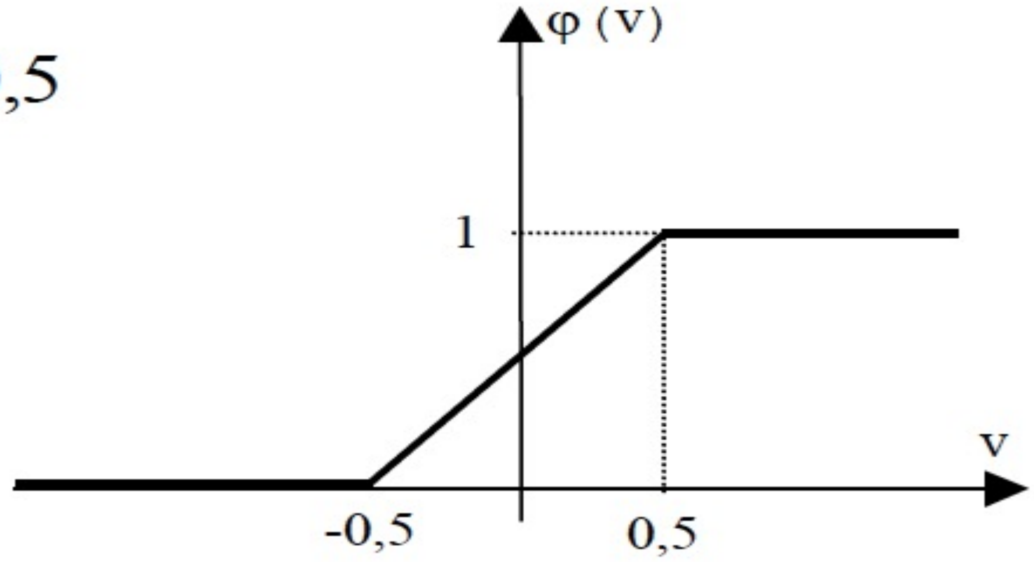
$$\varphi(v) = \begin{cases} 1, & v \geq 0 \\ 0, & v < 0 \end{cases}$$



Activation functions

- Linear :

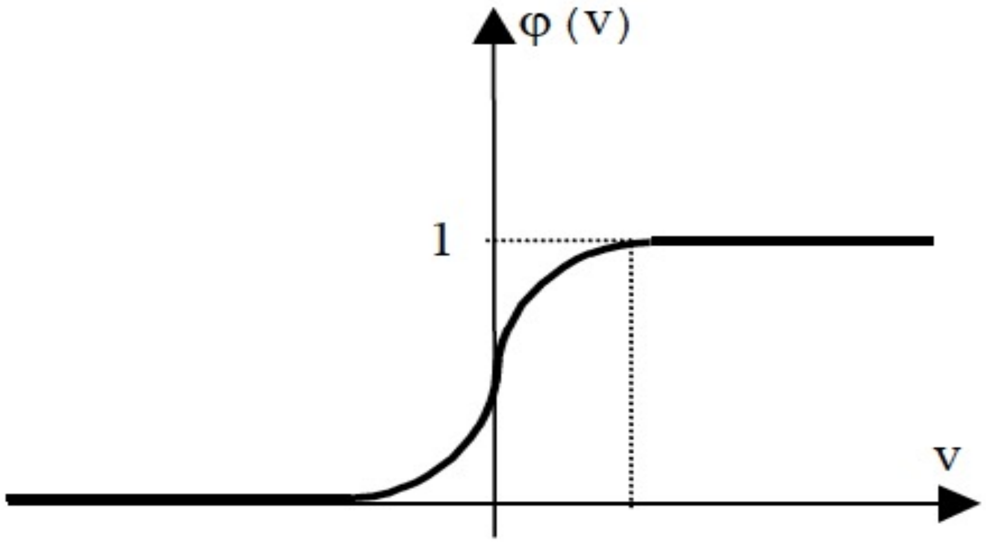
$$\varphi(v)=\begin{cases} 1, & v \geq 0,5 \\ v+0,5, & -0,5 < v < 0,5 \\ 0, & v \leq -0,5 \end{cases}$$



Activation functions

- Sigmoid :

$$\varphi(v) = \frac{1}{1+\exp(-av)}$$



Graphs

- ANNs can be represented using oriented graphs
- Synaptic edge representing linear input-output relation(multiplication by weight)
- Activation circle representing a non linear input-output relation of activation function

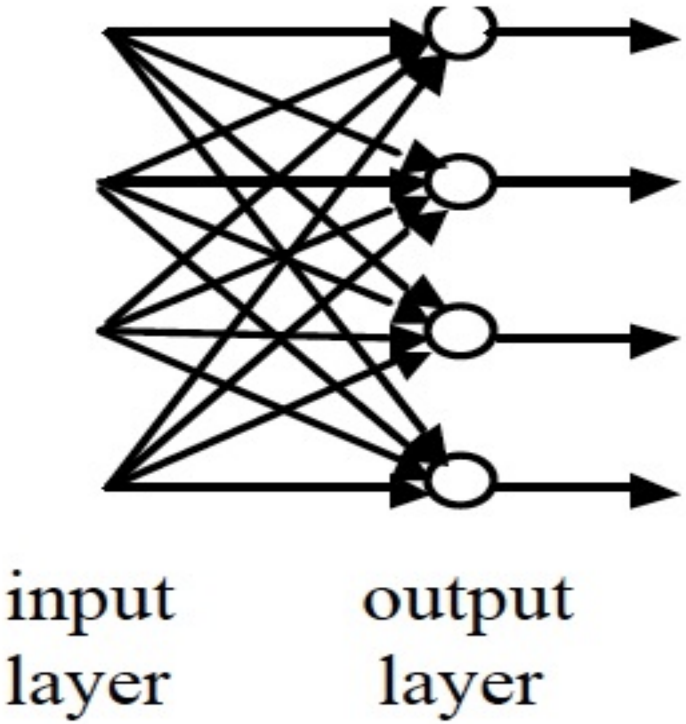
Network architectures

Network architectures (or topologies) define ways how neurons are mutually connected

- Three main architectures:
  - Single-layer feed-forward networks
  - Multi-layer feed-forward networks
  - Recurrent networks

Single-layer networks

- Has a single neuron layer (output layer)
- Input layer does not count due to lack of processing
- Network inputs are connected to neuron inputs
- Neuron outputs are also network outputs
- No feedbacks from outputs to inputs



Multilayer networks

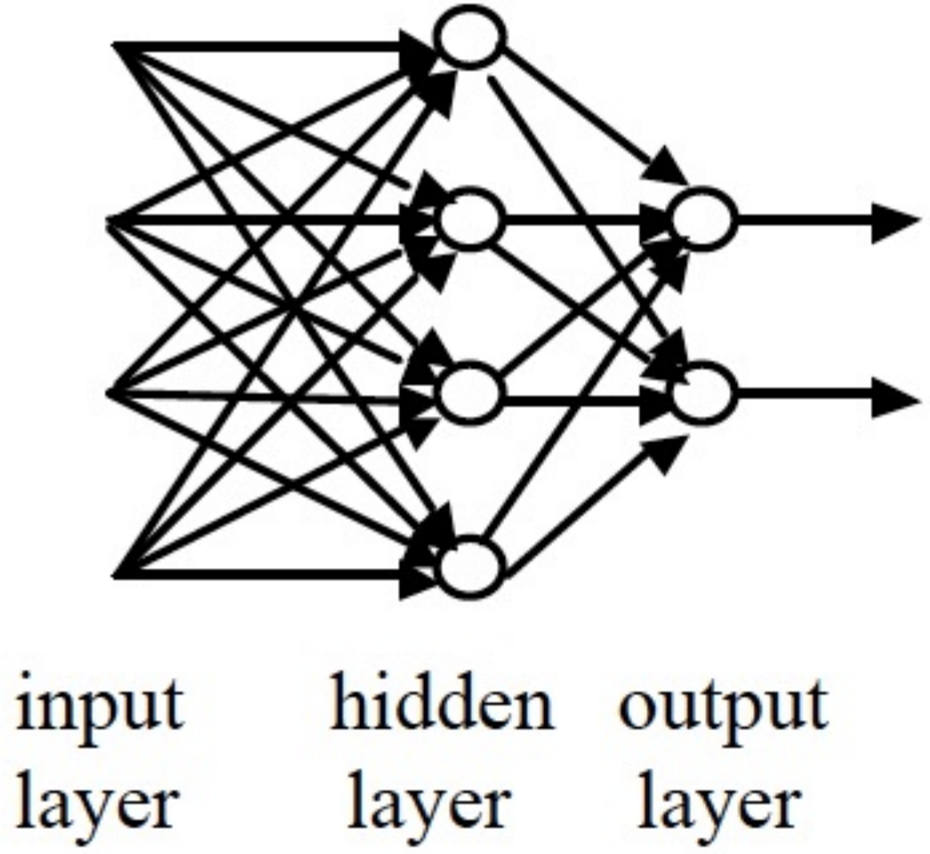
- Multilayer networks have one or more hidden layers, in addition to input and output layers
- Outputs from n-th layer are inputs to n+1-th layer

Connectedness:

- A network is fully connected when each neuron in a layer is connected to all neurons in the next layer
- If some connections are missing the network is partially connected

Multilayer networks

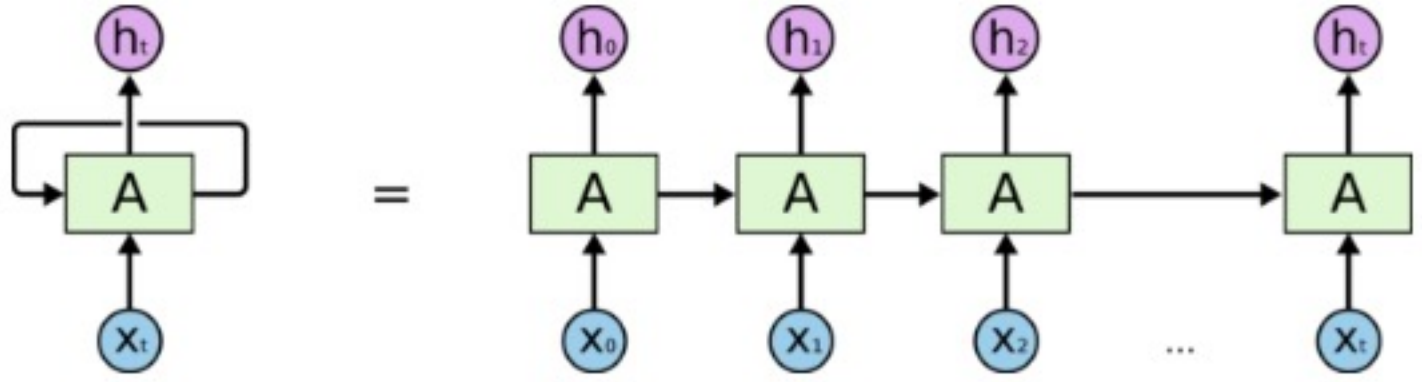
- An example of network with
- one hidden layer with four neurons
- The network has four input neurons
- There are two output neurons



Recurrent networks

- Have at least one feedback
- Can have hidden neurons
- Feedbacks give additional quality to recurrent networks
- Higher complexity for network analysis
- In combination with delay elements we obtain nonlinear dynamic systems which is crucial for ability to memorize patterns

Recurrent networks



An unrolled recurrent neural network.

https://towardsdatascience.com/understanding-rnn-and-lstm-f7cdf6dfc14e

Knowledge representation

- A definition of knowledge says: “Knowledge is stored information or models that is used by a person or a machine to interpret, predict or react to the outside world”
- The main characteristics of knowledge are:
  - Stored information
  - How is information encoded for later use
- Good intelligent systems depend on well chosen knowledge representation

ANN learning

- The main task of ANN is to learn the model of envirnoment in which it will operate and to maintain the model accurate in order to achieve the goals of the system
- The knowledge about the world consists of:
  - A priori information about everything that is known
  - Opservations of the workd that are used as examples for ANN learning
- Every learning example consists of an input-output value pair

Rules for knowledge representation

1. Similar inputs from similar classes should have similar representations and should be classified to the same class
2. Objects belonging to different classes should have sufficiently different representations in the network
3. Important object features must be represented using a larger number of neurons
4. A priori information and invariance should be built into ANN architecture so that ANN does not have to learn them

ANNs for pattern recognition

- Neural networks are often used for pattern recognition
- For pattern recognition a classifier determines a class for each input vector
- ANN is a non-linear classifier that divides the input space into classes that have non-linear borders
- Applications: image analysis, speech analysis, signal analysis, time series analysis, etc.

A priori information - Summary

- A priori knowledge about the problem can be built into ANN in two ways:
  - By restricting network topology through use of local connectedness (local receptive fields)
  - By limiting the choice of synaptic weights - by use of weight sharing
- Methods are very application-dependent

An Example: OCR

- For character recognition, input can be a vector of pixel values (e.g. from a 5x7 pixel matrix), and output could be an identity of a digit (0-9)
- Input layer can have 5x7=35 inputs
- Output layer could have 10 neurons (one for each digit)
- ANN would be trained with pairs of known input and output vectors (aka learning phase)
- After learning is completed, ANN could recognize previously unseen digits

Use of a priori information

- In character recognition example a priori information is
  - Input data (image) is two-dimensional
  - Input data has local structure (characters are localized)
- For this reason we should form ANN so that synaptic connections in input layer are localized (locally connected network)
  - Such local input region for a neuron is called receptive field

Invariance property in pattern recognition

- If an object image is rotated, translated or scaled w can still recognize the object
- If a speaker talks louder or silent, in low or high voice pitch, or if he has a cold we can still recogniz the words spoken
- In general, a pattern recognition system must be invariant to certain transformation of input information (e.g. signal, image)
- Classification result must not depend on such transformations of input information

Realization of invariance

- Some techniques to construct ANNs that will be invariant to certain informations of input information are:
  - Invariance based on ANN architecture (topology)
  - Invariance through learning
  - Invariance based on use of invariant features at ANN input

Invariance by architecture

- ANN architecture can be chosen to be invariant to certain transformations
- Example: Let us assume that we want to achive rotational invariance around the image center:
  - Let w_{ji} be the weight of the neuron *j* linked to the pixel *i* in input image
  - If $w_{ji}=w_{jk}$ for each two pixels *i* and *k* that are located at an equal distance from the image center than the network will be invariant to rotation around the image center

Invariance by learning

- Invariance can be achieved by learning so that ANN is trained with examples corresponding to transformed versions of the same object
- E.g. ANN can be trained with rotated versions of the same object in order to achieve rotational invariance
- Drawbacks of this approach are:
  - Invariance may not be achieved for rotated versions of some other object which was not used in training
  - Increased learning and computational demands due to increased number of training examples

Invariance by features

- Invariance can be achieved by selecting features that are invariant to desired transformations
- If such features are input into ANN than the network will not have to solve the problem of invariance
- Drawback: In this case features have to be selected manually or using some other approach – ANN is not involved in feature extraction/selection

Invariance by features

- Advantages of achieving invariance by features:
  - Input vector dimension can be reduced because only selected features are used for input, not the original data (e.g. image)
  - Demands on ANN are reduced
  - Invariance is achieved for all objects (not only for ones used in learning)

Use of a priori information

- In character recognition example a priori information is
  - Input data (image) is two-dimensional
  - Input data has local structure (characters are localized)
- For this reason we should form ANN so that synaptic connections in input layer are localized (locally connected network)
  - Such local input region for a neuron is called receptive field