Minia University
Faculty of Computers & information

Artificial Neural Networks and Deep Learning

Slides By:

T.A. Sarah Osama Talaat

E-mail: SarahOsama.fci@gmail.com

Slides were prepared based on set of references mentioned in the last slide

Agenda

- ❑Introduction
  - Neural networks to the rescue
  - Where can neural network systems help?
  - Neurons are information processing cells
- ❑Biological neural networks
- ❑Artificial neural networks definition and its component.
- ❑Network topologies

Let’s Start

Introduction

- Humans have a brain that can **learn**. Computers have some processing units and memory. They allow the computer to perform the most **complex numerical calculations** in a very **short time**, but they are not adaptive.

- If we compare **computer and brain**, we will note that, theoretically, the computer should be more powerful than our brain: **It comprises 10^9**

Introduction

	Brain	Computer
No. of processing units	$\approx 10^{11}$	$\approx 10^{9}$
Type of processing units	neurons	transistors
Type of calculation	massively parallel	usually serial
Data storage	associative	address-based
Switching time	$\approx 10^{-3}S$	$\approx 10^{-9}\mathrm{S}$
Possible switching operations	$\approx 10^{13}/_{S}$	$\approx 10^{18}/_{S}$
Actual switching operation	$\approx 10^{12}/_{S}$	$\approx 10^{10}/_{S}$

Introduction

Why are Artificial Neural Networks worth studying?

1. They are **extremely powerful computational** devices (Turing equivalent, universal computers).
2. Massive **parallelism** and decentralized computing make them **very efficient**.
3. They **can learn and generalize from training data** – so there is no need for enormous feats of programming.
4. Reliable: they are **very noise tolerant** – so they can cope with situations where normal symbolic systems would have difficulty.
5. In principle, they can do anything a symbolic / logic system can do.

Introduction

Why we are forced to use Neural Networks?

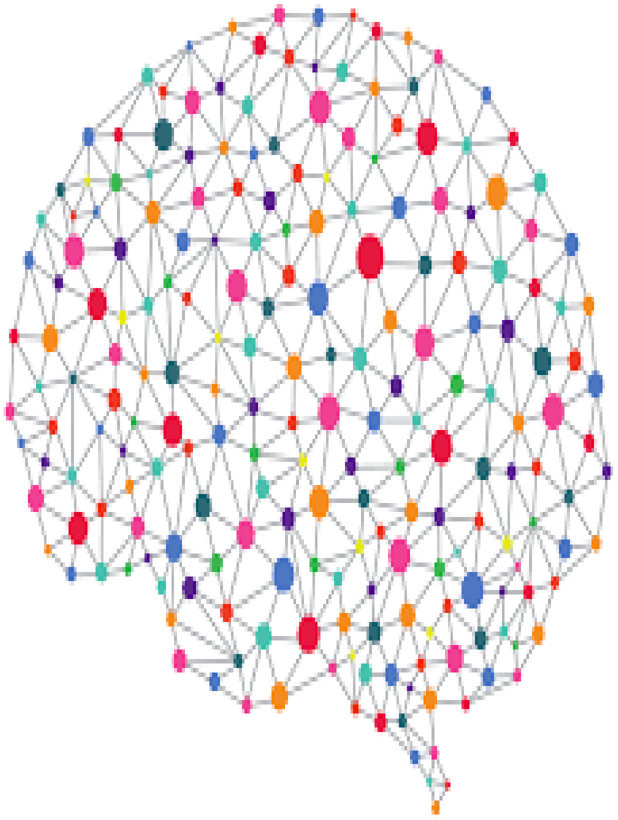
1. Neural networks have **ability to derive meaning from complicated or imprecise data**, can be used **to extract patterns and detect trends** that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer **"what if" questions.**
2. **Adaptive learning**: an ability to learn how to do tasks based on the data given for training or initial experience.
3. **Self-Organization:** an ANN can create its own organization or representation of the information it receives during learning time.
4. **Real Time Operation:** ANN computations may be carried out in **parallel, and special** hardware devices are being designed and manufactured which take advantage of this capability.

Introduction

Neural networks to the rescue

- **Neural networks** are configured for a specific application, such as **pattern recognition or data classification**, through a **learning process**

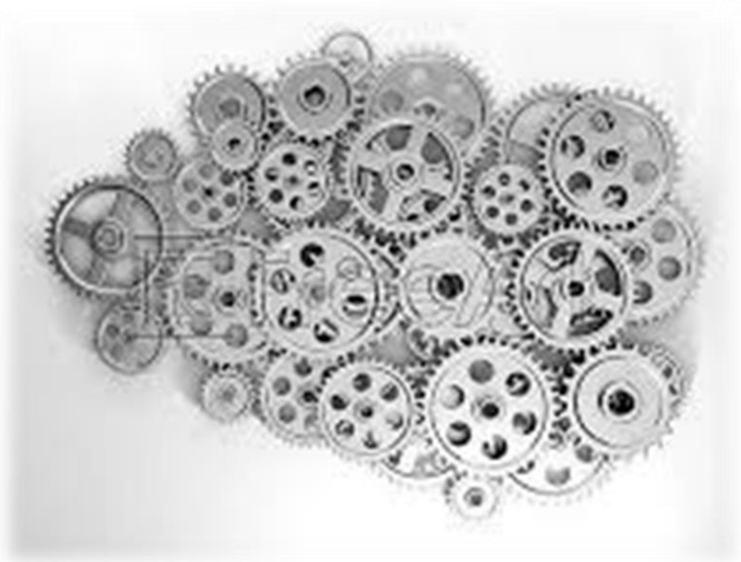
- **In a biological system,** learning involves adjustments to the synaptic connections between neurons that is same for artificial neural networks (ANNs).



Introduction

Where can neural network systems help?

- When we can't formulate an algorithmic solution.
- When we can get lots of examples of the behavior we require. **'learning from experience'**
- When we need to pick out the structure from existing data.



Introduction

Neurons are information processing cells

- A neuron is nothing more than a **switch** with information input and output.
- The switch will be activated if there are enough stimuli of other neurons hitting the information input. Then, at the information output, a pulse is sent to, for example, other neurons.

Biological neural networks
How the Human Brain Learns?

- In the human brain, a typical **neuron** **collects signals** from others through a host of fine structures called **dendrites**. The **neuron sends out spikes of electrical activity** through a long, thin stand known as an **axon,** which splits into thousands of branches. **At the end of each branch**, a structure called a **synapse** **converts the activity from the axon into electrical effects** that inhibit or excite activity in the connected neurons. When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. **Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes.**

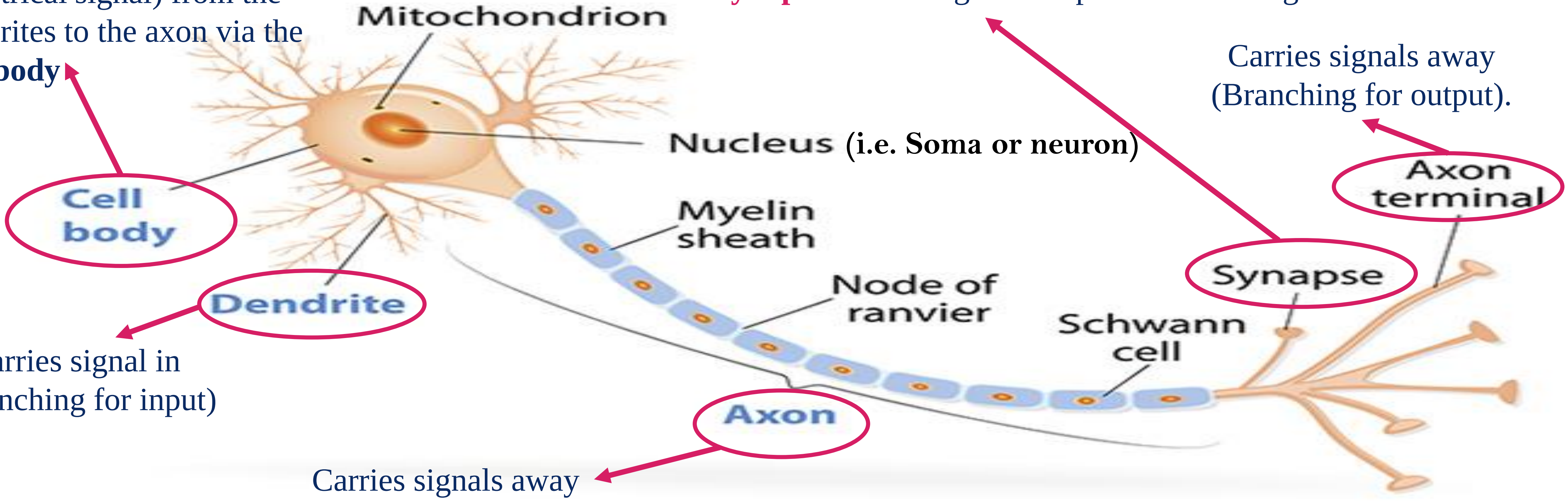
Inspiration From Neurobiology

The information circulates (electrical signal) from the dendrites to the axon via the **cell body**

Electrochemical contact between neurons

synapse size changes in response to learning

Carries signals away (Branching for output).



Carries signal in (Branching for input)

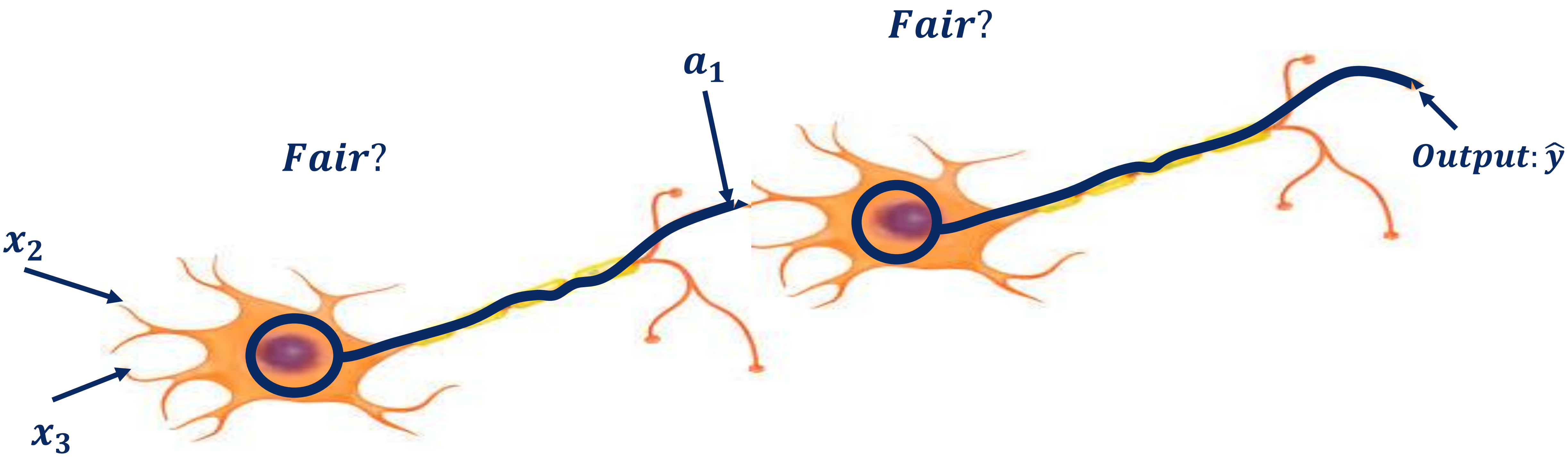
Carries signals away

Figure(3.1): Illustration of a biological neuron with the components discussed in this text.

Inspiration From Neurobiology

Figure(3.2): Illustration of a artificial neuron with the components discussed in the human neurons.

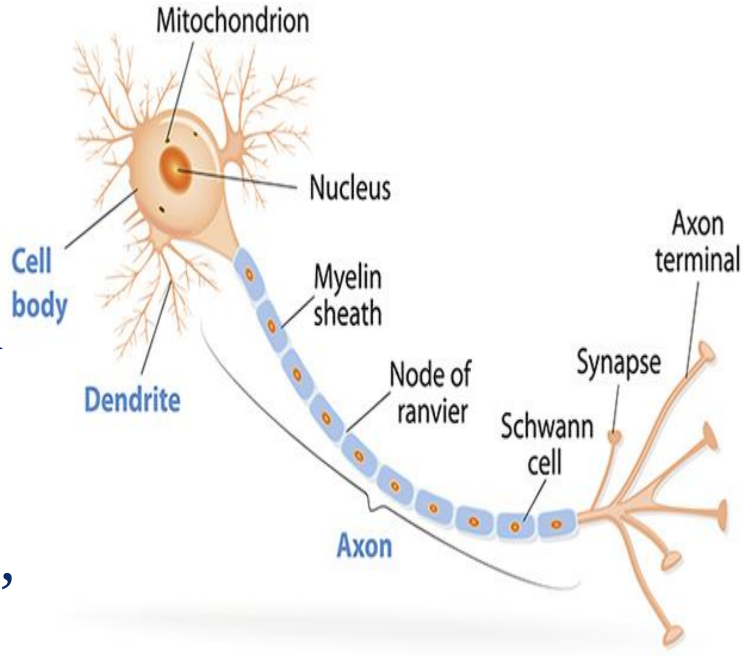
Inspiration From Neurobiology



Figure(3.2): Illustration of a artificial neuron with the components discussed in the human neurons.

Inspiration From Neurobiology

- A neuron has many-inputs and one-output unit
- Output can be ***excited*** or ***not excited (inhibition)***
- Incoming signals from other neurons determine if the neuron shall **excite ("fire")**
- Output subject to attenuation in the ***synapses***, which are junction parts of the neuron
- Basically, a biological neuron receives inputs from other sources, combines them in some way, performs a generally nonlinear operation on the result, and then output the final result.



Inspiration From Neurobiology

Components of a Neuron

- Now we want to take a look at the **components of a neuron** (Figure 3.1). In doing so, we will follow the way the electrical information takes within the neuron.
- The dendrites of a neuron receive the information by special connections called the **synapses.** The synapse resistance to the incoming signal can be changed during a "**learning**" process.

Inspiration From Neurobiology

Synapse concept

❑ Synapses types:

- There are two types of synapse which are **electrical and chemical synapses**.
- The **electrical synapse** is the simpler variant. An electrical signal received by the synapse, i.e. coming from the **presynaptic** side, is directly transferred to the **postsynaptic** nucleus of the cell. Thus, there is a direct, strong, un adjustable connection between the signal transmitter and the signal receiver.
- The **chemical synapse** is the more distinctive variant. Here, the electrical coupling of source and target does not take place, the coupling is interrupted by the **synaptic cleft.** This cleft electrically separates the presynaptic side from the postsynaptic one.

Inspiration From Neurobiology

Dendrites collect all parts of information

Dendrites:

- Dendrites branch like trees from the cell nucleus of the neuron (which is called soma) and **receive electrical signals from many different sources**, which are then transferred into the nucleus of the cell.
- The amount of branching dendrites is also called **dendrite tree.**

Inspiration From Neurobiology

In the soma the weighted information is accumulated

- After the cell nucleus (**soma**) has received a plenty of **activating (=excitation or stimulating)** and **inhibiting (=diminishing)** signals by synapses or dendrites, the soma accumulates these signals. As soon as the accumulated signal exceeds a certain value (called threshold value), the cell nucleus of the neuron activates an electrical **pulse** which then is transmitted to the neurons connected to the current one.

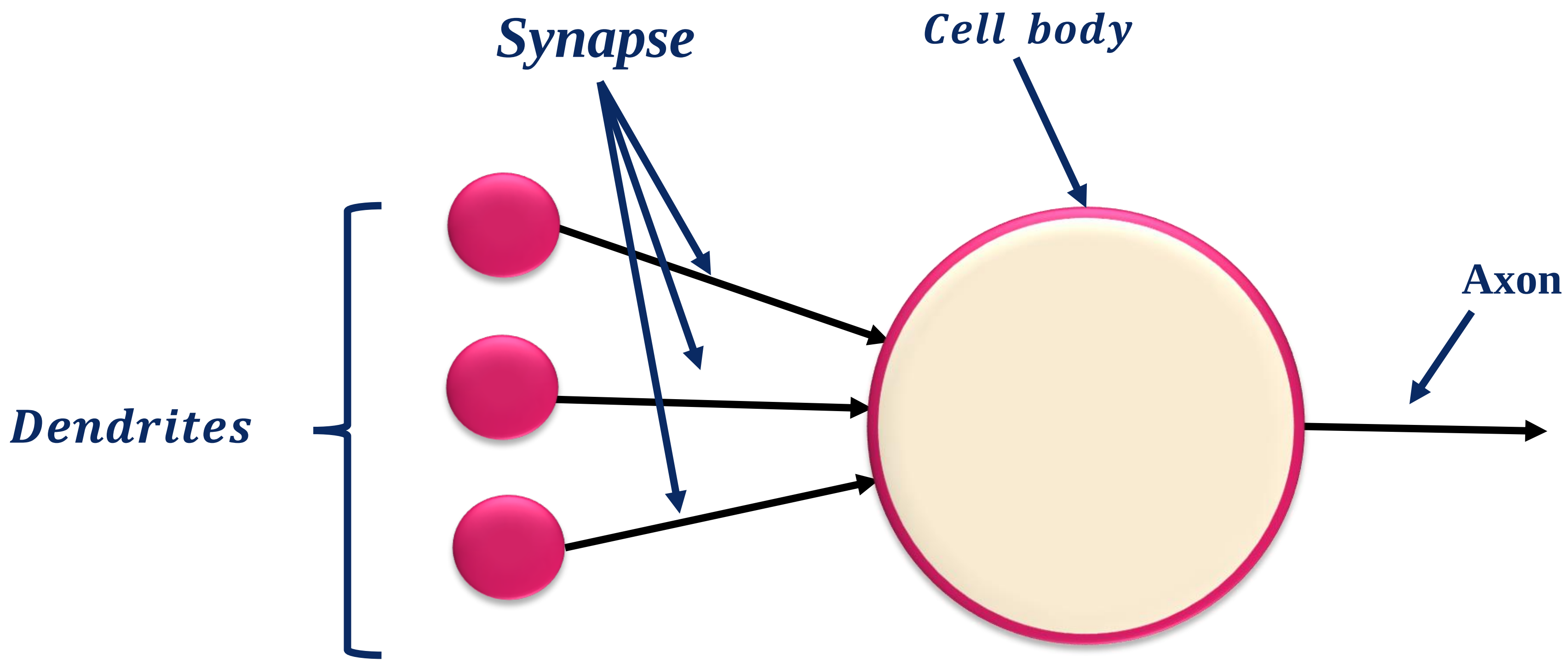
Inspiration From Neurobiology

From Human Neurons to Artificial Neurons

- We conduct these neural networks by first trying to deduce the essential features of neurons and their interconnections. We then typically program a computer to simulate these features. However because our knowledge of neurons is incomplete and our computing power is limited, our models are necessarily gross idealizations of real networks of neurons.

Inspiration From Neurobiology

From Human Neurons to Artificial Neurons



Figure(3.2): Illustration of a artificial neuron with the components discussed in the human neurons.

Artificial Neural Networks

Definition

- **Artificial Neurons(ANs):**
  - Artificial Neurons are crude approximations of the neurons found in brains. They may be physical devices, or purely mathematical constructs.
- **Artificial Neural Networks (ANNs)**
  - ANNs are networks of Artificial Neurons, and hence constitute crude approximations to parts of real brains. They may be physical devices, or simulated on conventional computers.

Components of Artificial Neural Networks

- The ANN consists of simple processing elements called **neurons, units or nodes**.
- Each **neuron** is connected to other nodes with an **associated weight** (strength).
- Each **neuron** has a **single threshold value**.
- **Weighted sum** of all the inputs coming into the neuron is formed and the threshold is subtracted from this value = **activation**.
- **Activation signal** is passed through an activation function to produce the output of the neuron.
- **The next slides contain the formal definitions for most of the neural network components used later in this course.**

Components of Artificial Neural Networks
The concept of time in neural networks

❑ Definition 3.1 (The concept of time)

- **The current time** (present time) is referred to as (t), the next time step as $(t+1)$, the preceding one as $(t-1)$. All other time steps are referred to analogously. When we have several mathematical variables (e.g. net_j or o_i) refer to a certain point in time, the notation will be, for example , $net_j(t\text{-}1)$ or $o_i(t)$

Components of Artificial Neural Networks
Components of neural networks

❑ Definition 3.2 (Neural network).

- **A neural network** is a sorted triple **(N, V, w)** with two sets N, V and a function w, where *N* is the **set of neurons** and *V* a set **{(i, j)|i, j ∈ *N*}** whose elements are called ***connections*** between neuron ***i*** and neuron ***j***. The function ***w : V***→ℝ defines the **weights**, where ***w((i, j)),*** the weight of the connection between neuron ***i*** and neuron ***j***, **is shortened to w**. Depending on the point of view it is either undefined or 0 for connections that do not exist in the network

Components of Artificial Neural Networks

Connections carry information that is processed by neurons

❑ Connections:

- Data are transferred between neurons via connections with the connecting weight being either excitatory or inhibitory. The formal definition of connections has already been included in the definition of the neural network.

Components of Artificial Neural Networks

The propagation function converts vector inputs to scalar network inputs

❑ Propagation function concept:

- Looking at a neuron j, we will usually find a lot of neurons with a connection to j, i.e. which transfer their output to j.
- For a neuron j the **propagation function** receives the outputs ***o*** of other neurons $\boldsymbol{i_1}, \boldsymbol{i_2}, \boldsymbol{i_3}, \ldots, \boldsymbol{i_n}$(which are connected to j), and transforms them in consideration of the connecting weights $\boldsymbol{w_{i,j}}$ into the network input $\boldsymbol{net_j}$ that can be further processed by the ***activation function***. Thus, the **network input** is the result of the propagation function.

Components of Artificial Neural Networks

The propagation function converts vector inputs to scalar network inputs

❑ Definition 3.3 (Propagation function and network input):

- Let $\boldsymbol{I} = \{\boldsymbol{i_1}, \boldsymbol{i_2}, \boldsymbol{i_3}, .., \boldsymbol{i_n}\}$ be the set of neurons, such that $\forall\, \mathbf{z} \in \{\mathbf{1}, .., \boldsymbol{n}\}: \exists\, \boldsymbol{w_{i_z,j}}$. Then the network input of $\boldsymbol{j}$ called $\boldsymbol{net_j}$, is calculated by the propagation function $\boldsymbol{f_{prop}}$ as follows:

$$\boldsymbol{net_j = f_{prop}(o_{ij},\ldots, o_{in}, w_{i1,j}, \ldots, w_{in,j})} \qquad \textbf{(3.1)}$$

- Here the ***weighted sum*** is very popular: The multiplication of the output of each neuron $\boldsymbol{i}$ by $\boldsymbol{w_{i,j}}$, and the summation of the results:

$$\boldsymbol{net_j = \sum_{i=1}^{n} (o_i \cdot w_{ij})} \qquad \textbf{(3.2)}$$

Components of Artificial Neural Networks

The activation is the "switching status" of a neuron

❑ The activation concept:

- Based on the model of nature every neuron is, to a certain extent, at all times active, excited or whatever you will call it. **The reactions of the neurons to the input values depend on this activation state**.
- The **activation state** indicates the extent of a neuron’s activation and is often shortly referred to as **activation**. Its formal definition is included in the following definition of the activation function. But generally, it can be defined as follows:

Components of Artificial Neural Networks

The activation is the "switching status" of a neuron

Definition 3.4 (Activation state / activation in general):

- Let j be a neuron. The activation state a_j, in short activation, is explicitly assigned to j, indicates the extent of the neuron's activity and results from the activation function.

Components of Artificial Neural Networks

Neurons get activated if the network input exceeds their threshold value

❑ Threshold value:

- Near the threshold value, the activation function of a **neuron reacts particularly sensitive**.
- From the biological point of view, the threshold **value represents the threshold at which a neuron starts firing**.

❑ Definition 3.5 (Threshold value in general):

- Let j be a neuron. **The threshold value Θ_j** is uniquely assigned to j and marks the position of the maximum gradient value of the activation function.

Components of Artificial Neural Networks

Activation function

❑ Activation function:

- The activation function is also called ***transfer function.*** The activation function determines the activation of a neuron dependent on **network input and threshold** value. At a certain time the activation a_j of a neuron j depends on the previous activation state of the neuron and the external input.

Components of Artificial Neural Networks

Activation function

Definition 3.6 (Activation function and activation):

- Let j be a neuron. **The activation function** is defined as:

$$a_j(t) = f_{act}(net_j(t), a_j(t-1), \theta_j)$$

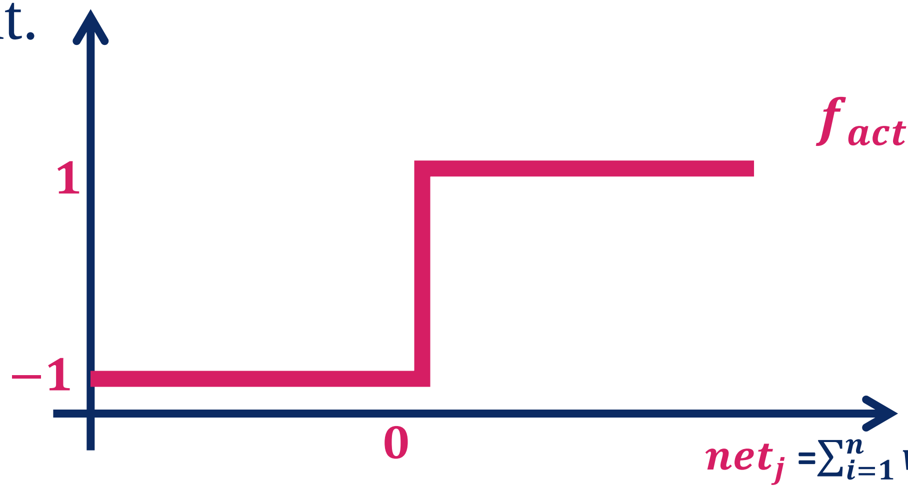
- Activation function transforms the **network input** net_j, as well as the previous activation state $a_j(t-1)$ into a new activation state $a_j(t)$, with the threshold value θ_j playing an important role, as already mentioned.

Components of Artificial Neural Networks

Common activation functions

❑ Binary threshold function:

- The simplest activation function is the **binary threshold function values** (also referred to as Heaviside function), which can only **take on two**. **If the input is above a certain threshold, the function changes from one value to another**, but otherwise remains constant.



$$f_{act}(net_j) = \begin{cases} value, & net_j > \Theta \\ x, & net_j \leq \Theta \end{cases},$$

$$net_j = \sum_{i=1}^{n} w_i x_i + b$$

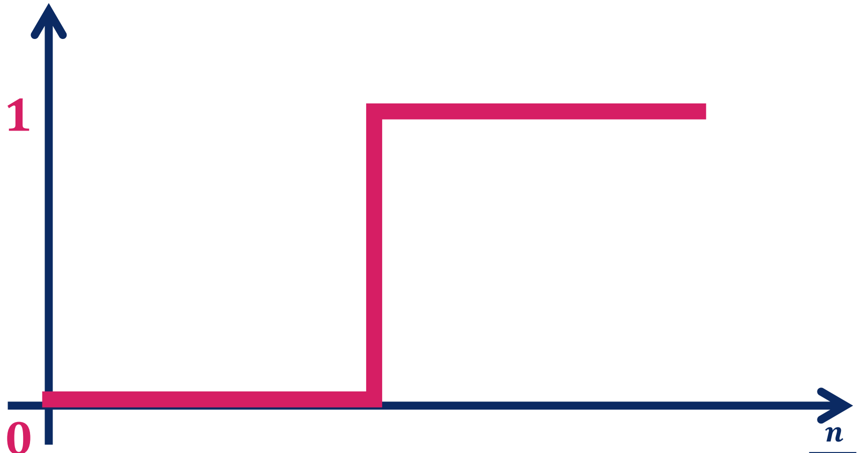
Figure(3.4): Illustration of binary threshold function

Components of Artificial Neural Networks

Common activation functions

❑ Threshold function:

- The simplest activation function is the **threshold function values**, which can only **take on two**. **If the input is above a zero, the function changes from one value to one**, but otherwise it changes zero.



$$f_{act}(net_j) = \begin{Bmatrix} 1\,, net_j \geq 0 \\ 0 net_j \leq 0 \end{Bmatrix}$$

$$net_j = \sum_{i=1}^{n} w_i\, x_i + b$$

Figure(3.4): Illustration of binary threshold function

Components of Artificial Neural Networks

Common activation functions

Fermi function, logistic or sigmoid function:

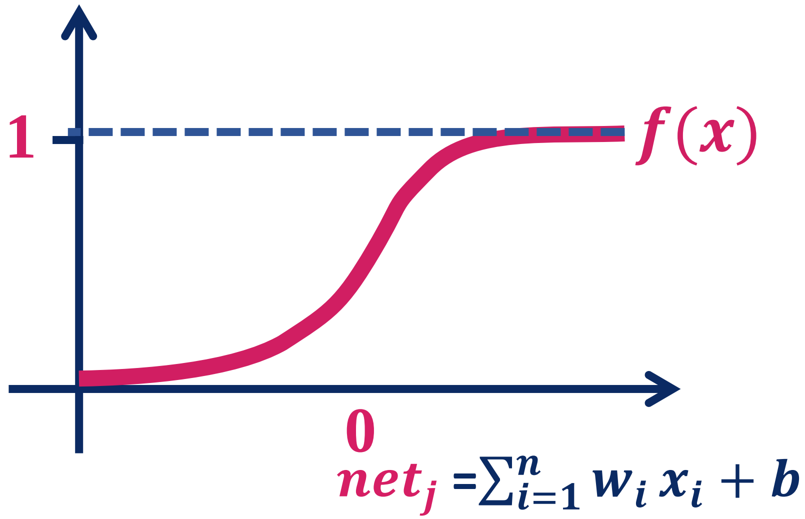
- This function maps to the range of values of (0, 1) .

$$f_{act}(net_j) = f(net_j) = \frac{1}{1+e^{-net_j}}$$

- The Fermi function can be expanded by a temperature parameter T into the form

$$f_{act}(net_j) = f(net_j) = \frac{1}{1+e^{-net_j/T}}$$

- This function is used in the logistic regression algorithm



Figure(3.4): Illustration of fermi function with temperature parameter.

Components of Artificial Neural Networks

Output functions

- **Output function:**
  - An output function may be used to process the activation once again.
  - The output function of a neuron ***j*** calculates the values which are **transferred to the other neurons connected to j**.

Components of Artificial Neural Networks

Output functions

- **Definition 3.7 (Output function):**
  - Let j be a neuron. The output function calculates the output value o_j of the neuron j from its activation state a_j.

$$f_{out}(a_j) = o_j$$

  - Generally, the output function is defined globally, too. Often this function is the identity, i.e. the activation a_j is directly output:

$$f_{out}(a_j) = a_j, \qquad so, o_j = a_j$$

  - Unless explicitly specified differently, we will use the identity as output function within this course.

Components of Artificial Neural Networks

The bias neuron

- **The bias neuron is a technical trick to consider threshold values as connection weights**
  - We know that in many network paradigms neurons have a **threshold** value that indicates when a neuron becomes active. Thus, the **threshold value is an activation function parameter of a neuron**.
  - From the biological point of view this sounds most plausible, but it is complicated to access the activation function at runtime in order to train the threshold value.
  - But threshold values $\Theta_{j1}, \ldots, \Theta_{jn}$ for neurons j can also be realized as **connecting weight of a continuously firing neuron**: For this purpose **an additional bias neuron** whose output **value is always 1** is integrated in the network and connected to the neurons $j_1, \ldots, j_n$. These new connections get the weights $-\Theta_{j1}, \ldots, -\Theta_{jn}$ i.e. they get the negative threshold values.

Components of Artificial Neural Networks

The bias neuron

Definition 3.8 (Bias neuron):

- A bias neuron is a neuron whose output value is always 1 and which is represented by

- It is used to represent neuron biases as connection weights, which enables any weight training algorithm to train the biases at the same time.
- Instead of including the **threshold value in the activation function**, it is now included in the **propagation function**. Or even shorter: The **threshold value** is subtracted from the network input, i.e**. it is part of the network input**.

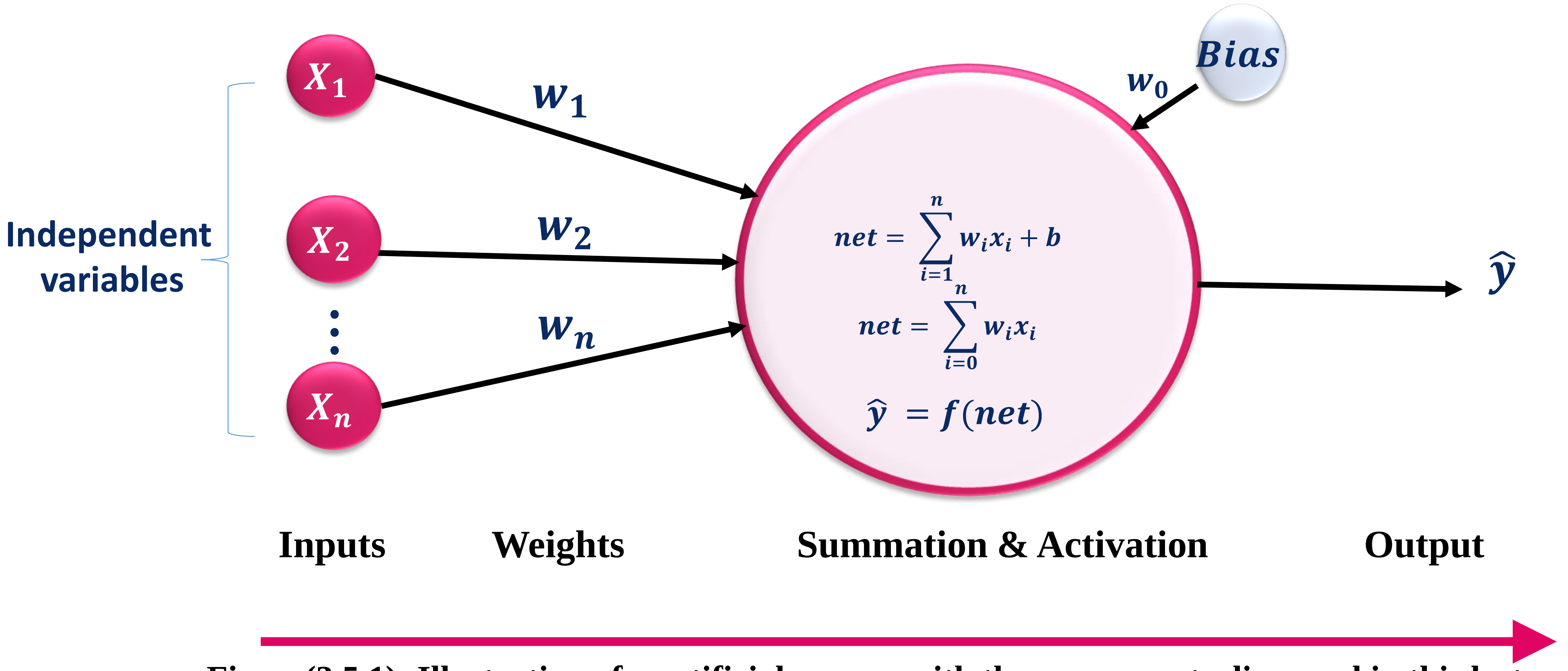
Components of Artificial Neural Networks

The bias neuron

❑ The advantages and disadvantages of bias neuron:

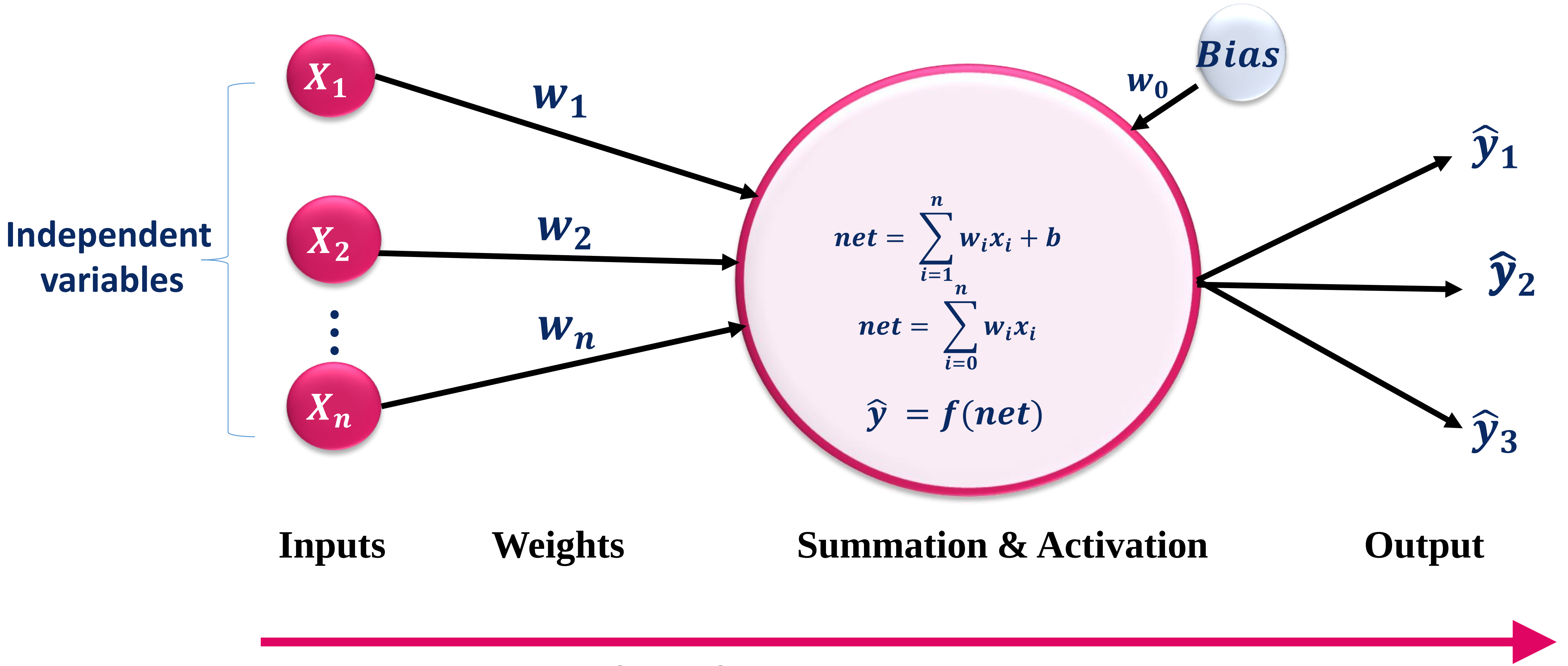
- The **advantage** of the bias neuron is the fact that **it is much easier to implement it in the network.**
- One **disadvantage** is that the representation of the network already becomes quite ugly with only a few neurons, let alone with a great number of them. By the way, a bias neuron is often referred to as on neuron.

Components of Artificial Neural Networks
Mathematical representation



Figure(3.5.1): Illustration of a artificial neuron with the components discussed in this lecture.

Components of Artificial Neural Networks
Mathematical representation



Figure(3.5.2): Illustration of a artificial neuron with the components discussed in this lecture.

Components of Artificial Neural Networks

Learning strategies adjust a network to fit our needs

❑ Definition 3.9 (General learning rule):

- The **learning strategy** is an algorithm that can be used to change and thereby train the neural network, so that the network produces a desired output for a given input. We will address this subject later in detail.

Network topologies

ANN Architectures/ Structures/Topologies

Network Topologies

- Feedforward
  - **Single layer feedforward**
  - **Multi-layer feedforward**
- Recurrent networks
- Completely linked networks

Network topologies
Feedforward networks

❑ Feedforward networks:

- The neurons are arranged in **separate layers**: **one** input layer, ***n*** hidden processing layers (invisible from the outside, that's why the neurons are also referred to as hidden neurons) and **one** output layer.
- In a feedforward network **each neuron in one layer has only directed connections to the neurons of the next layer** (towards the output layer).
- There is **no connection** between the neurons in the same layer.

Network topologies
Feedforward networks

Feedforward networks:

- The neurons in one layer **receive inputs from the previous layer.**
- The neurons in one layer **delivers its output to the next layer**
- We will often be confronted with feedforward networks in which every neuron ***i*** is connected to all neurons of the next layer (these layers are called **completely linked**).
- To prevent naming conflicts the output neurons are often referred to as (omega) Ω .

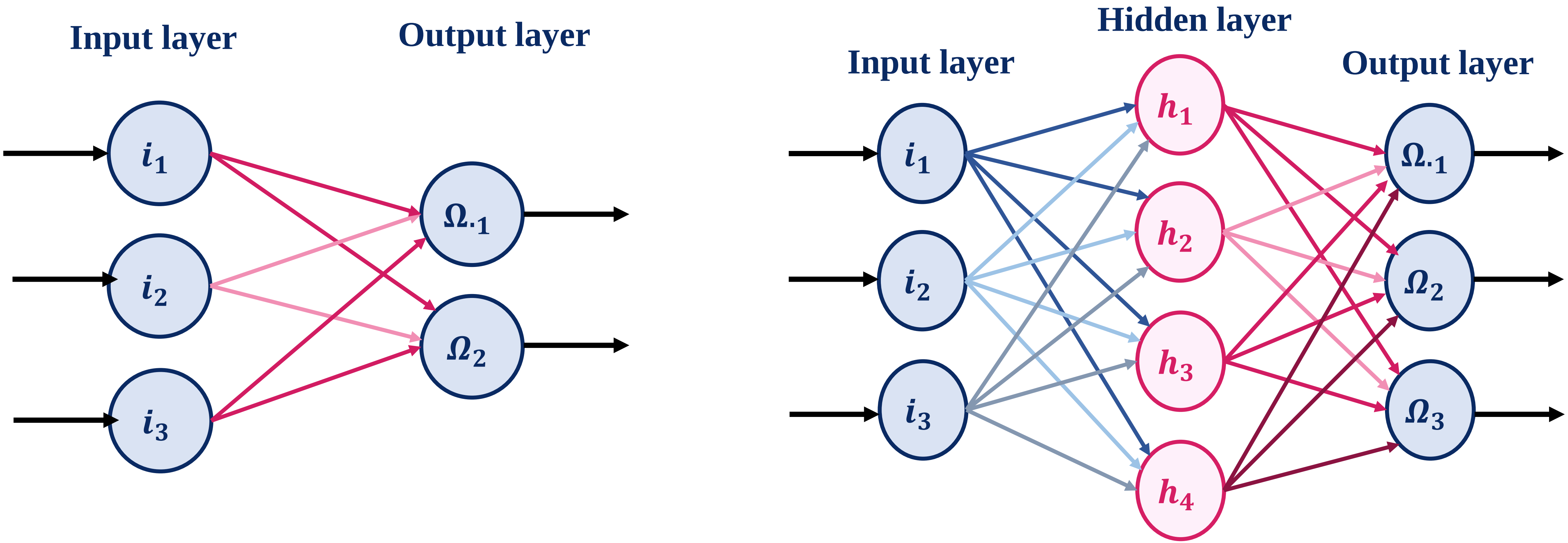
Network topologies
Feedforward networks

- **Feedforward networks:**

1. **Single-Layer Feed-forward:**
   - One input layer and one output layer of processing units.
   - **No feed-back connections**. (For example, **a Single Layer Perceptron**.)
2. **Multi-Layer Feed-forward NNs**
   - **One** input layer, **one** output layer, and **one or more** hidden layers of processing units. **No feed-back connections.**
   - The hidden layers sit in between the input and output layers, and are thus hidden from the outside world. (For example, **a Multi-Layer Perceptron**.).

Network topologies

Feedforward networks



Figure(3.6): Left side: a Single layer feedforward network with two layers: three input neurons and two output neurons
Right side: a Multi-Layer feedforward network with three layers: three input neurons, four hidden neurons and three output neurons.

Network topologies

Feedforward networks

- **Definition 3.10 (Feedforward networks):**
  - The neuron layers of a feedforward network (figure 3.6) are clearly separated: One input layer, one output layer and one or **more processing layers** which are invisible from the outside (also called **hidden layers**). **Connections are only permitted to neurons of the following layer.**

Network topologies

Feedforward networks with shortcut connections

❑ Shortcut connections:

- Some feedforward networks permit the so-called **shortcut connections.** The connection in these networks **skip** one or more levels. These connections may only be directed towards the output layer, too.

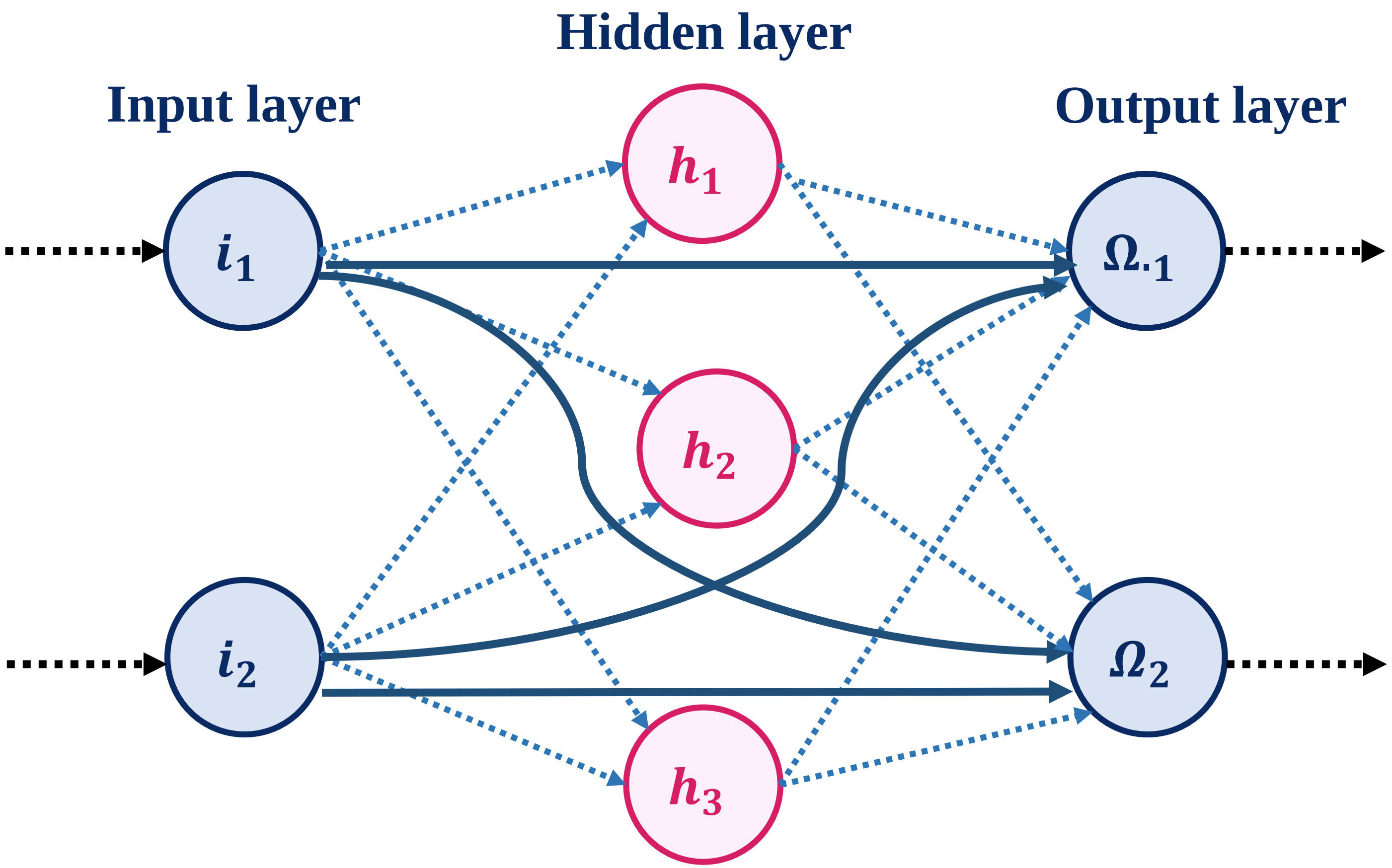
Network topologies

Feedforward networks with shortcut connections

❑ Definition 3.11 (Feedforward network with shortcut connections).

- The neuron layers of a feedforward network (figure 3.7) are clearly separated: One input layer, one output layer and one or more processing layers which are invisible from the outside (also called **hidden layers**). **The connections may not only be directed towards the next layer but also towards any other subsequent layer.**

Network topologies
Feedforward networks



Figure(3.7): A feedforward network with shortcut connections which are represented by solid lines

Network topologies

Hinton diagram

❑ Hinton diagram:

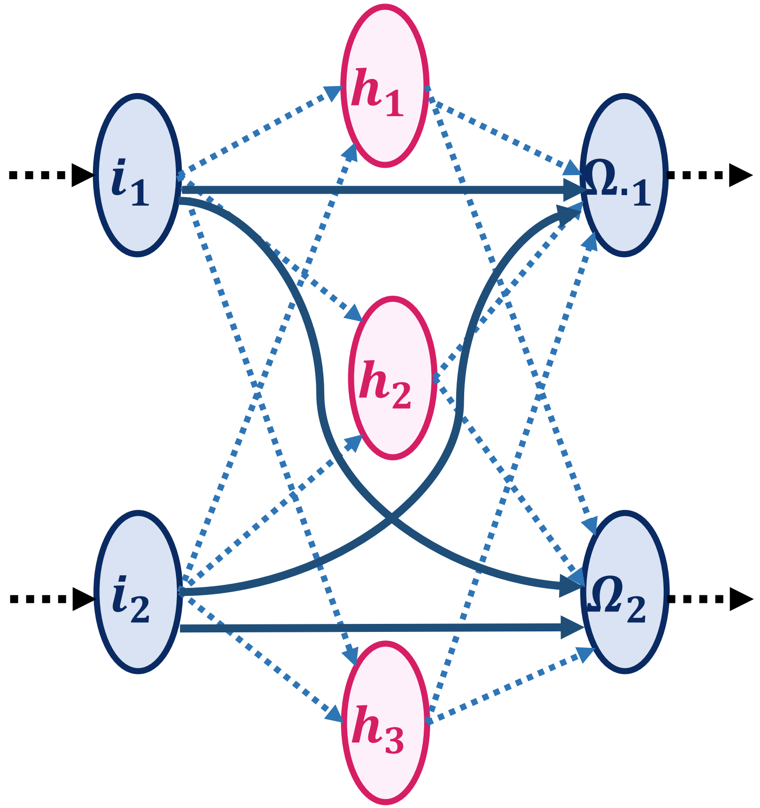
- In the Hinton diagram the **dotted lines (i.e. connection, weights)** are represented by **light blue fields**, the **solid ones** by **dark blue fields**.
- In order to clarify that the connections are between the line neurons and the column neurons, I have inserted the small arrow ↱ in the upper-left cell.
- The input and output arrows, which were added for reasons of clarity, cannot be found in the Hinton diagram.

Network topologies

Feedforward networks

↱	i_1	i_2	h_1	h_2	h_3	$\Omega_{\cdot 1}$	$\Omega_{\cdot 2}$
i_1							
i_2							
h_1							
h_2							
h_3							
$\Omega_{\cdot 1}$							
$\Omega_{\cdot 2}$							

Table (3.2): Hinton diagram of a feedforward network with shortcut connections which are represented by solid lines



Figure(3.7): A feedforward network with shortcut connections which are represented by solid lines

Network topologies
Recurrent networks

❑ Recurrent networks:

- Recurrence is defined as the process of a neuron influencing itself by any means or by any connection. Recurrent networks do not always have explicitly defined input or output neurons.
- Any network with at **least one feed-back connection**.
- It may, or may not, have hidden units. (For example, **a Simple Recurrent Network**.)

Network topologies
Recurrent networks topologies

- **Recurrent networks**
  - Direct recurrences
  - Indirect recurrences
  - Lateral recurrences

Network topologies

Direct Recurrent networks

- **Direct Recurrence networks(start and end at the same neuron):**
  - Some networks allow for neurons to be **connected to themselves**, which is called **direct recurrence** (or sometimes self-recurrence). As a result, neurons inhibit and therefore strengthen themselves in order to reach their activation limits..

Network topologies
Direct Recurrent networks

- **Definition 3.12 (Direct Recurrence):**
  - First, expand the feedforward network by connecting a neuron $\boldsymbol{j}$ to itself, with the weights of these connections being referred to as $\boldsymbol{w_{j,j}}$. In other words: the diagonal of the weight matrix W may be different from 0.

Network topologies

Direct recurrence networks

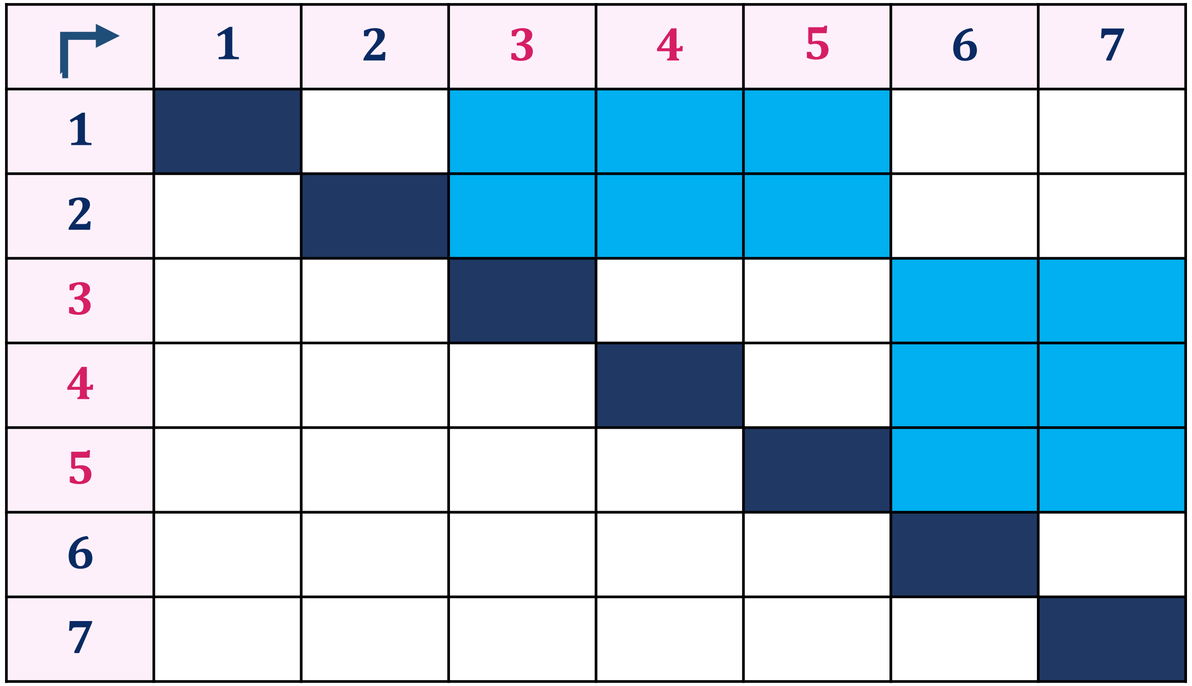
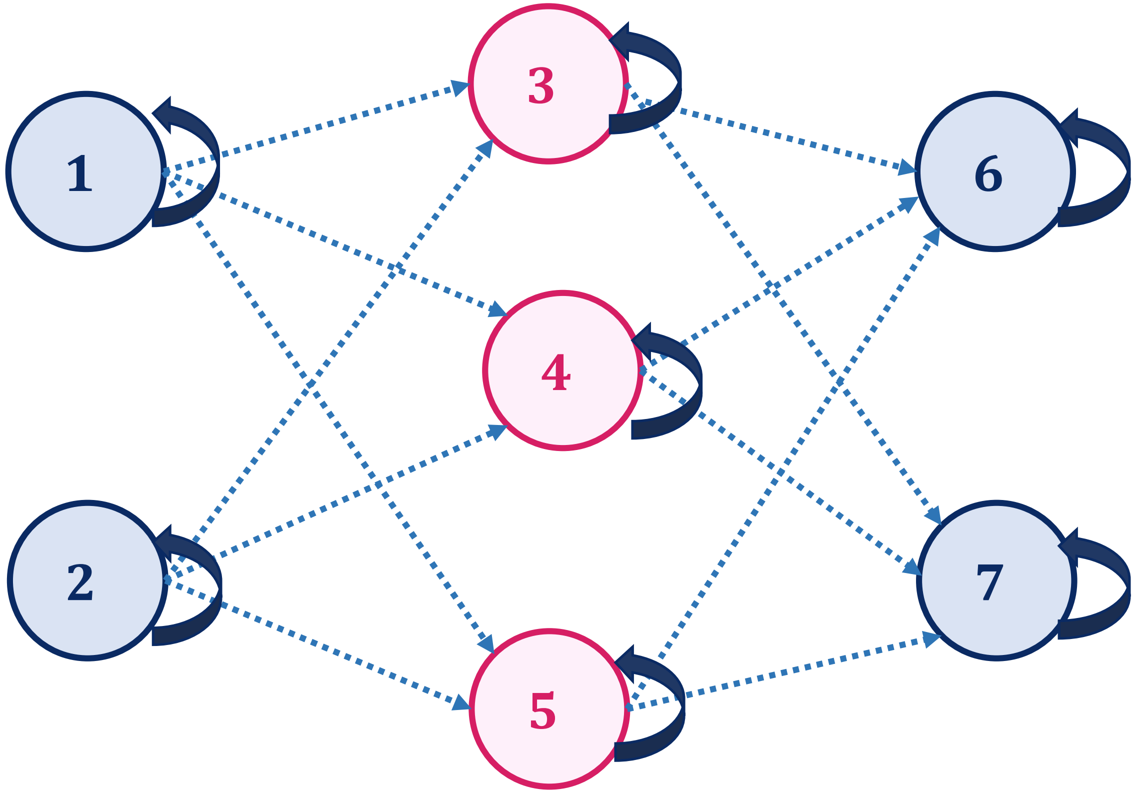


Table (3.3): Hinton diagram of direct recurrences which are represented by diagonal in the Hinton diagram matrix.



Figure(3.8): A network similar to a feedforward network with directly recurrent neurons. The direct recurrences are represented by solid lines

Network topologies
Indirect Recurrences networks

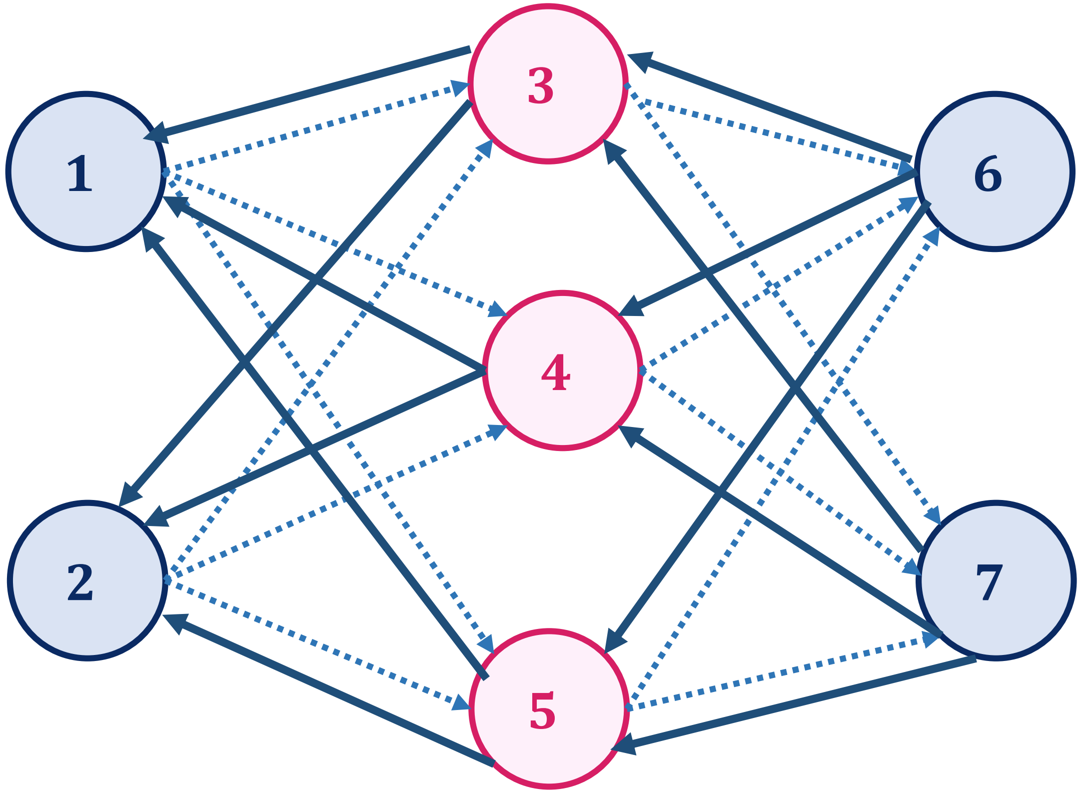
- **Indirect Recurrence networks(start and end at the same neuron):**
  - If connections are allowed towards the input layer, they will be called indirect recurrences. Then a neuron ***j*** can use indirect forwards connections to influence itself.
- **Definition 3.13(Indirect Recurrence):**
  - Indirect recurrence network is like a feedforward network, but with additional connections between neurons and their preceding layer being allowed. Therefore, below the diagonal of **W** is different from 0.

Network topologies

Indirect recurrences networks

↱	1	2	3	4	5	6	7
1							
2							
3							
4							
5							
6							
7							

Table (3.4): Hinton diagram of indirect recurrences.



Figure(3.9): A network similar to a feedforward network with indirectly recurrent neurons. The indirect recurrences are represented by solid lines

Network topologies
Lateral Recurrences networks

- **Lateral Recurrences networks(connect neurons within one layer):**
  - Connections between neurons within one layer are called **lateral recurrences.** Here, each neuron often inhibits the other neurons of the layer and strengthens itself. As a result only the strongest neuron becomes active
- **Definition 3.14 (Lateral Recurrence):**
  - A laterally recurrent network permits connections within one layer.

Network topologies

Laterally recurrence networks

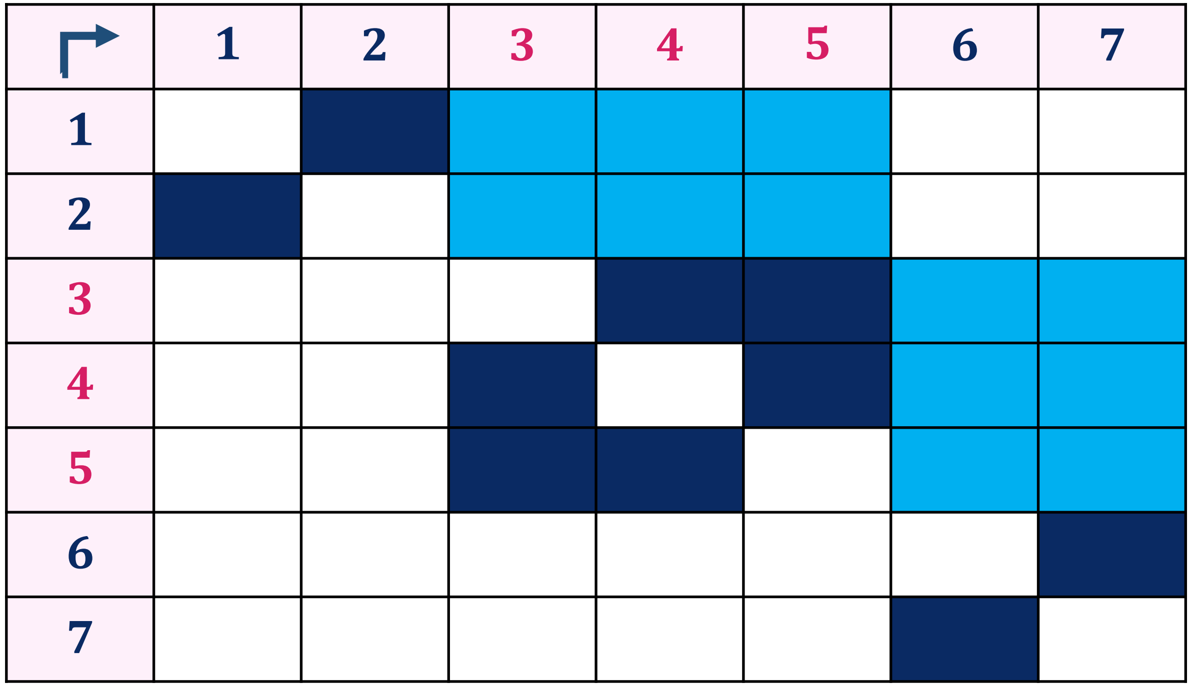
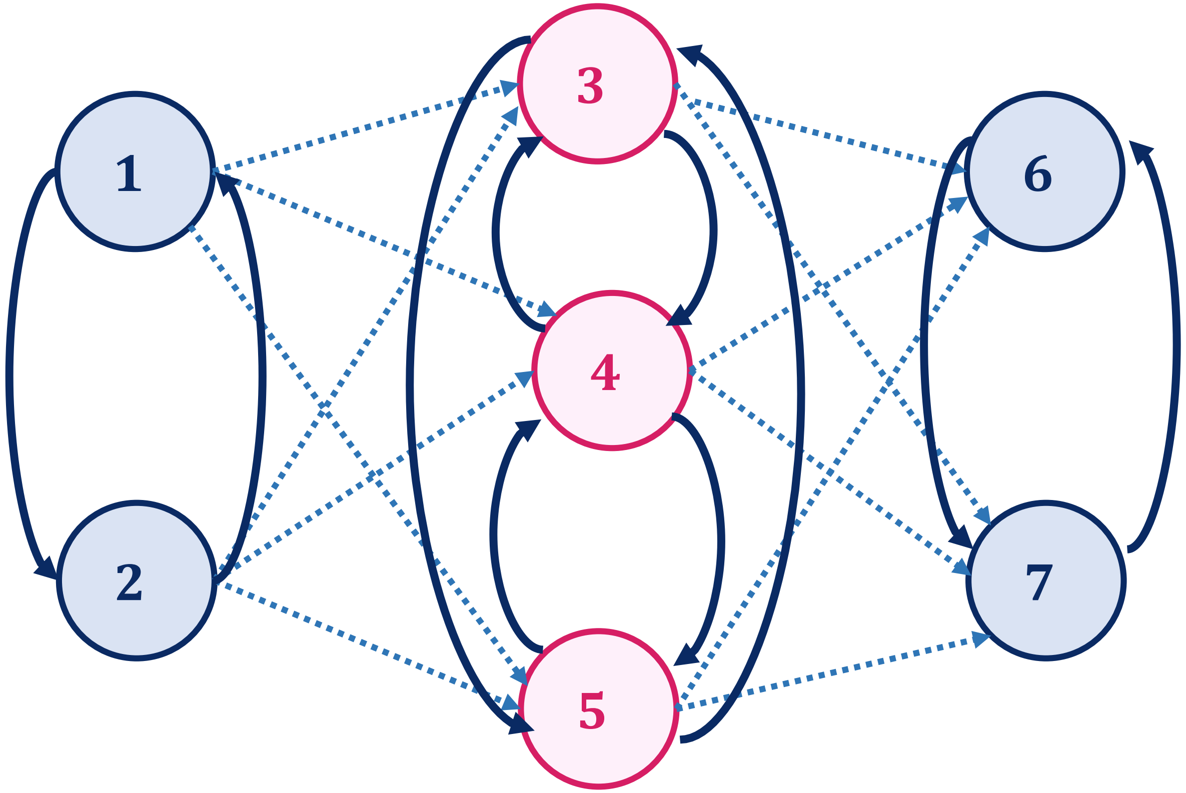


Table (3.4): Hinton diagram of laterally recurrences



Figure(3.10): A network similar to a feedforward network with laterally recurrent neurons. The indirect recurrences are represented by solid lines

Network topologies
Completely linked networks

- **Completely linked networks :**
  - Completely linked networks **permit (allow)** connections between all neurons, **except** for direct recurrences. Furthermore, the connections must be symmetric. A popular example are the self-organizing maps, which will be introduced in one of the next lectures.

Network topologies
Completely linked networks

- **Definition 3.15 (Completely linked networks):**
  - In this case, every neuron is always allowed to be connected to every other neuron – but as a result every neuron can become an input neuron. Therefore, direct recurrences normally cannot be applied here and clearly defined layers do not longer exist. Thus, the matrix **W** may be unequal to 0 everywhere, except along its diagonal.

Network topologies
Completely linked networks

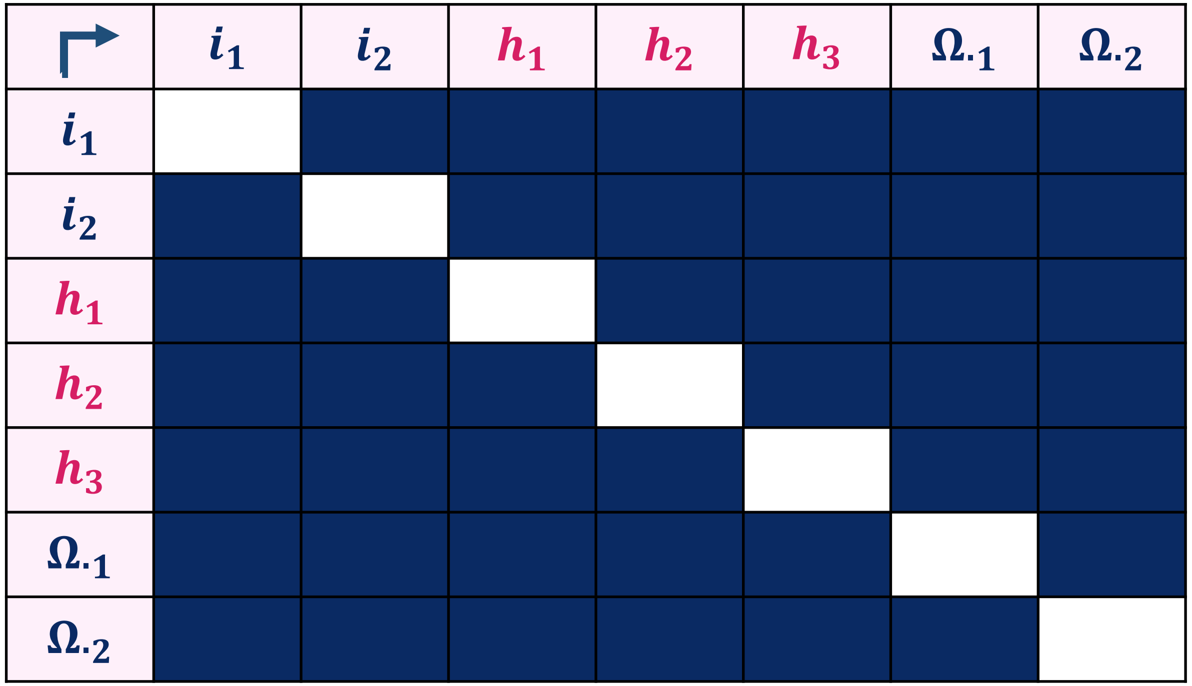
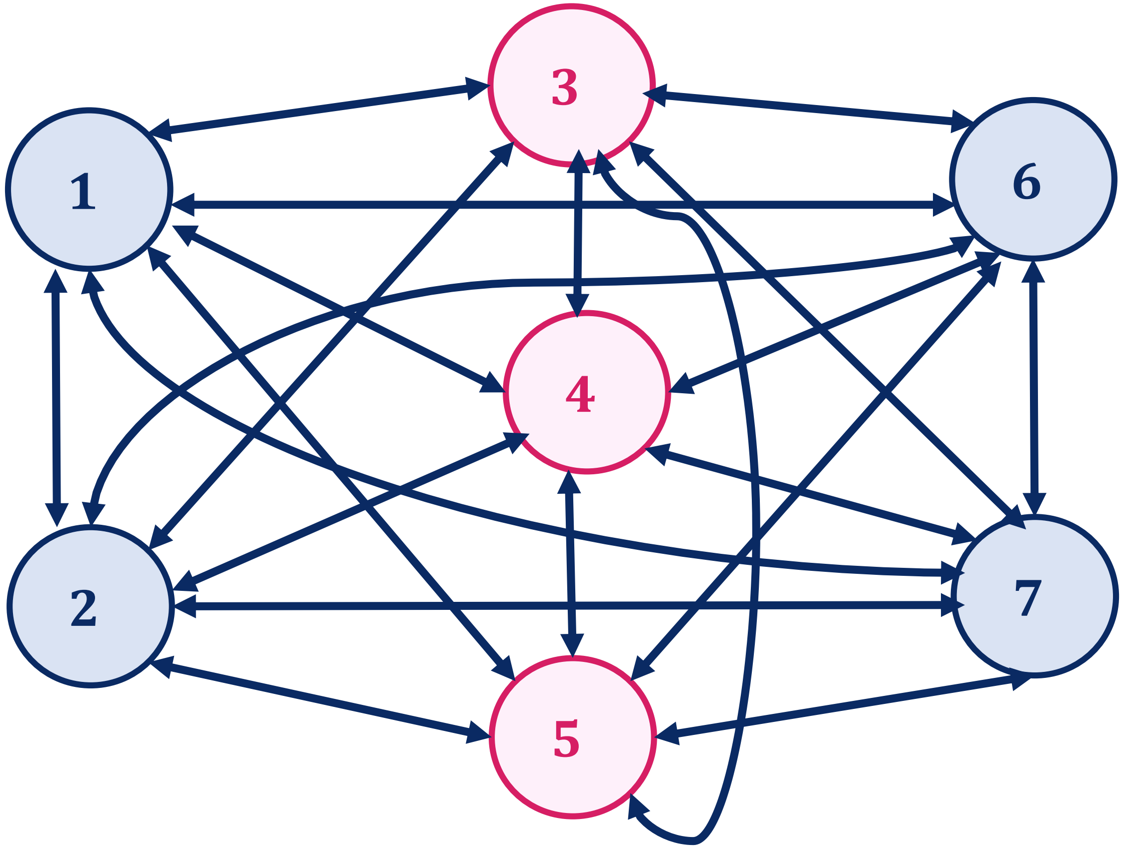


Table (3.2): Hinton diagram of a feedforward network with shortcut connections which are represented by solid lines



Figure(3.12): A completely linked network with symmetric connections and without direct recurrences

How Can The Artificial Neural Networks Learn?

- A learning system changes itself in order to adapt to e.g. environmental changes. A neural network could learn from many things. In principle, a neural network changes when its components are changing, as we have learned above.

How Can The Artificial Neural Networks Learn?

- **Theoretically, a neural network could learn by one of the following**

1. Developing new connections,
2. Deleting existing connections,
3. **Changing connecting weights**,
4. Changing the threshold values of neurons,
5. Varying one or more of the three neuron functions (remember: activation function, propagation function and output function),
6. Developing new neurons, or
7. Deleting existing neurons (and so, of course, existing connections).

How Can The Artificial Neural Networks Learn?

- **In this course,** we let our neural network learn by **modifying the connecting weights** according to rules that can be formulated as algorithms.

- Therefore a **learning procedure is always an algorithm** that can easily be implemented by means of a programming language. Later in this course I will assume that the definition of the term **desired output** which is worth learning is known (and I will define formally what a **training pattern** is) and that we have a training set of learning samples

References

- Kriesel, David. "A Brief Introduction to Neural Networks. 2007." URL http://www. dkriesel. com (2007).

Any Questions!?

Thank you