EENG582
Artificial Neural Networks

Learning Process

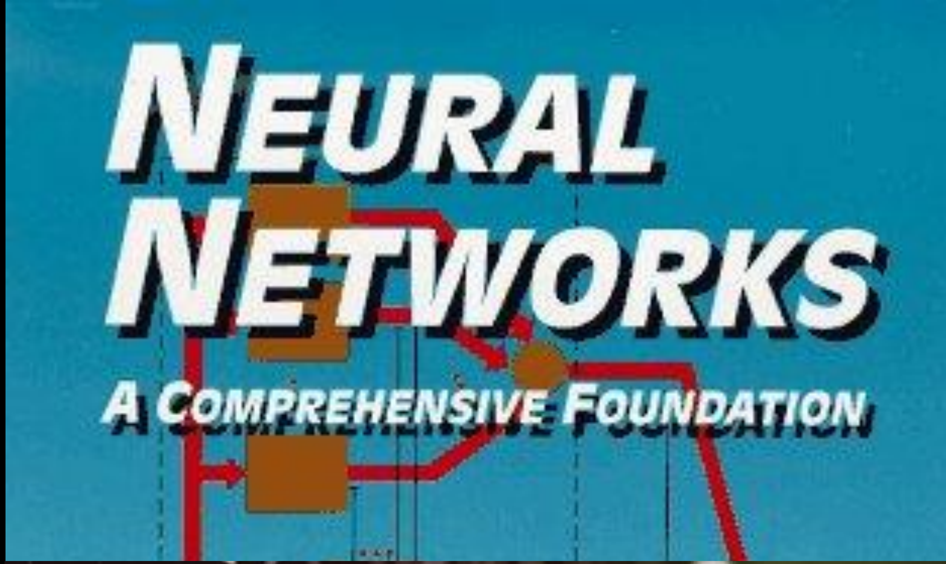
Sections 2.10 – 2.12
Learning Tasks

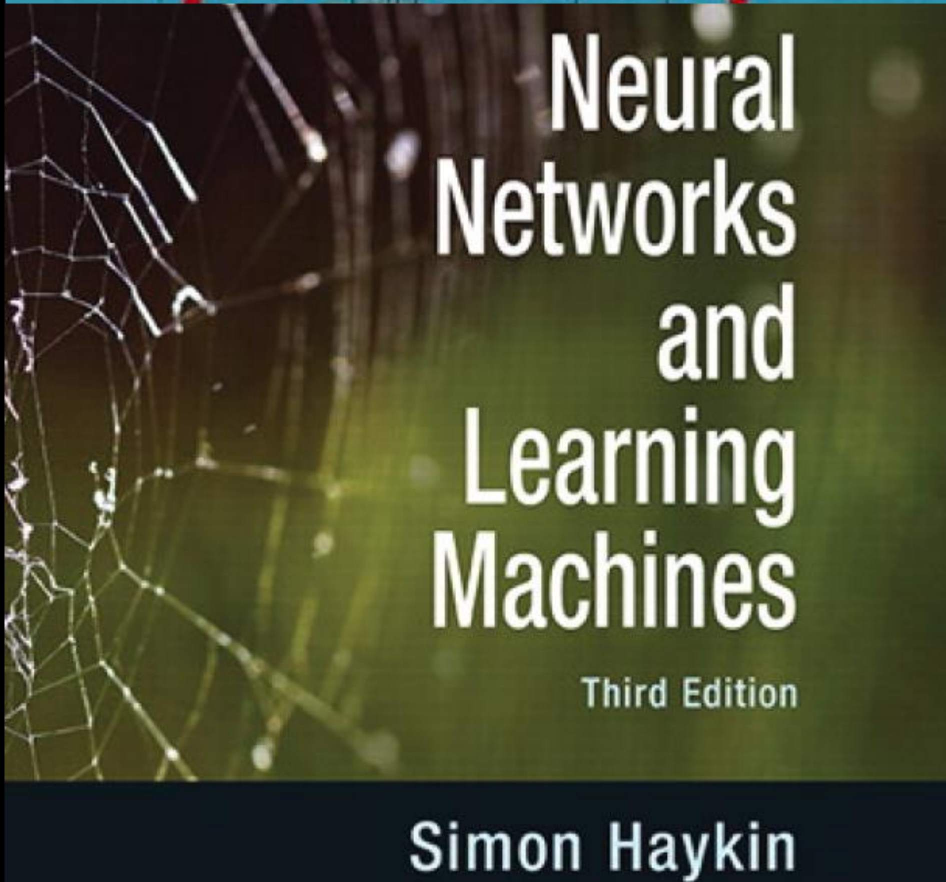
Artificial Neural Networks

Prof. Dr. Hasan AMCA

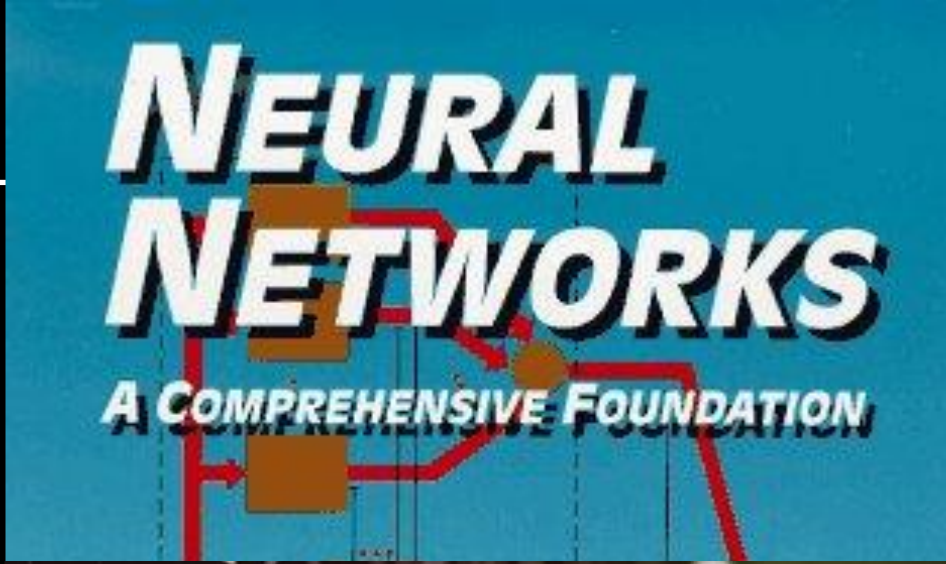
Electrical and Electronic Engineering Department
(ee.emu.edu.tr)

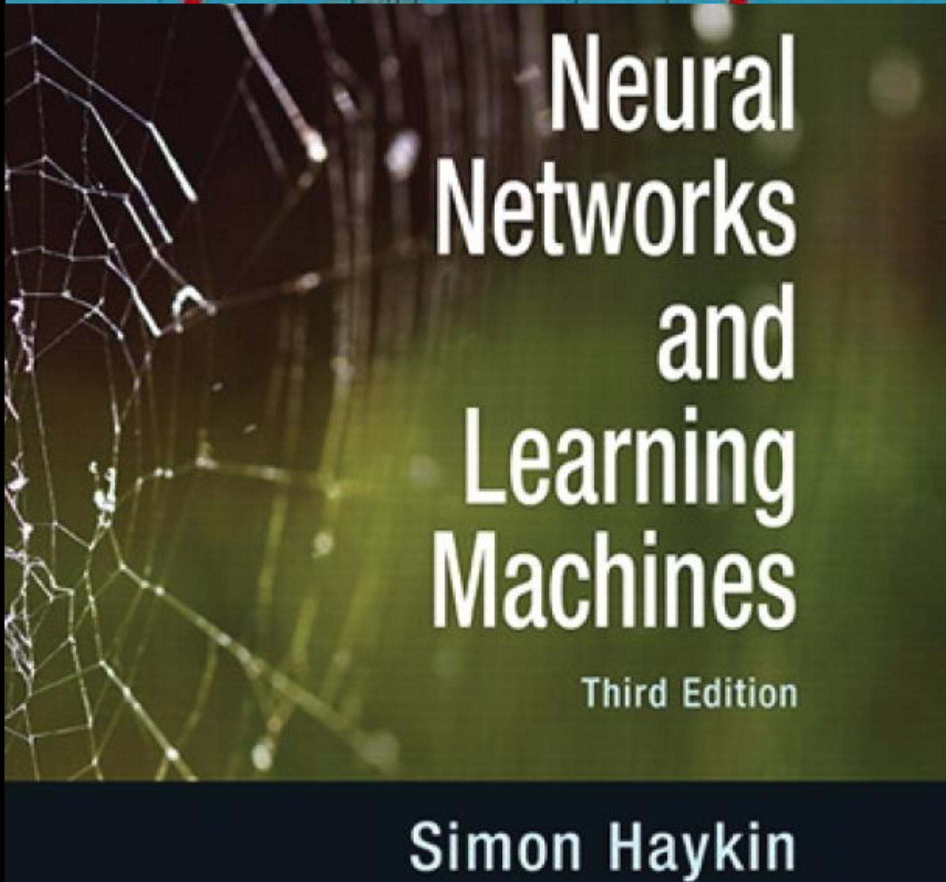
Eastern Mediterranean University
(emu.edu.tr)





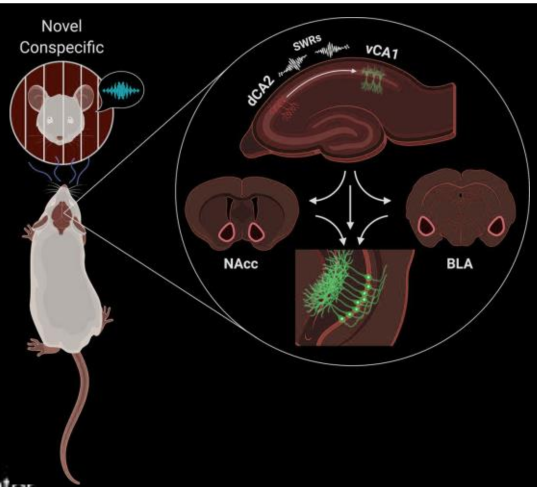
2 Learning Process



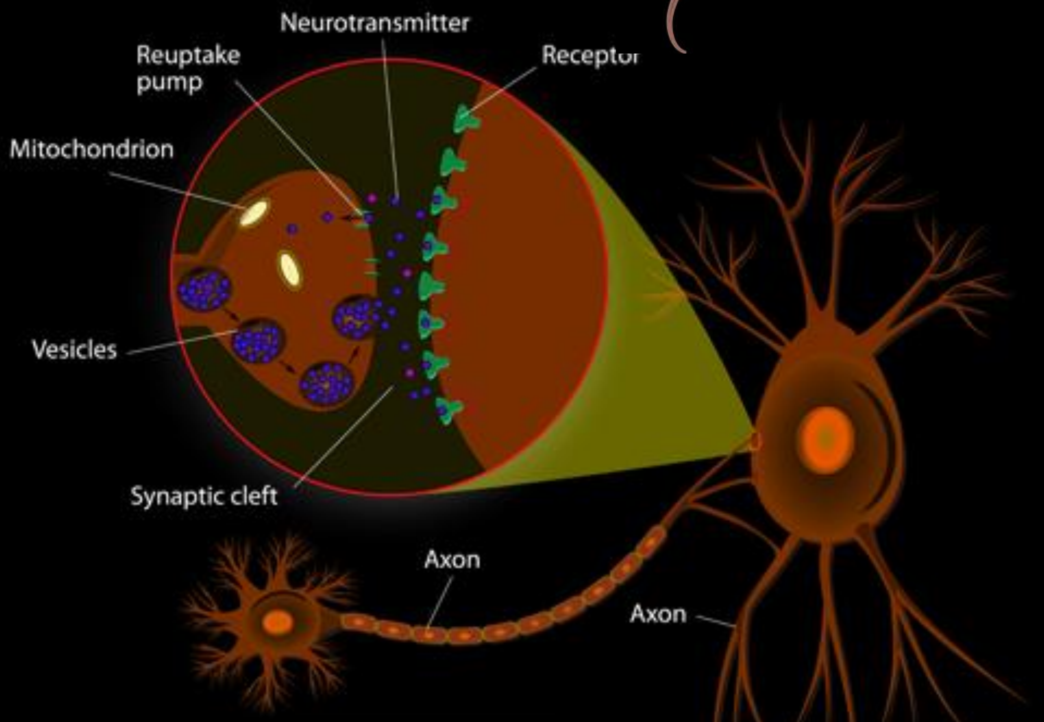


2.11 Learning Tasks: Memory

- In neurobiological context, memory refers to the neural alterations induced by the interaction of an organism with its environment.
- For the memory to be useful, the activity pattern must initially be stored in memory through a learning process.
- When a particular activity pattern is learned, it is stored in the brain where it can be recalled later when required.



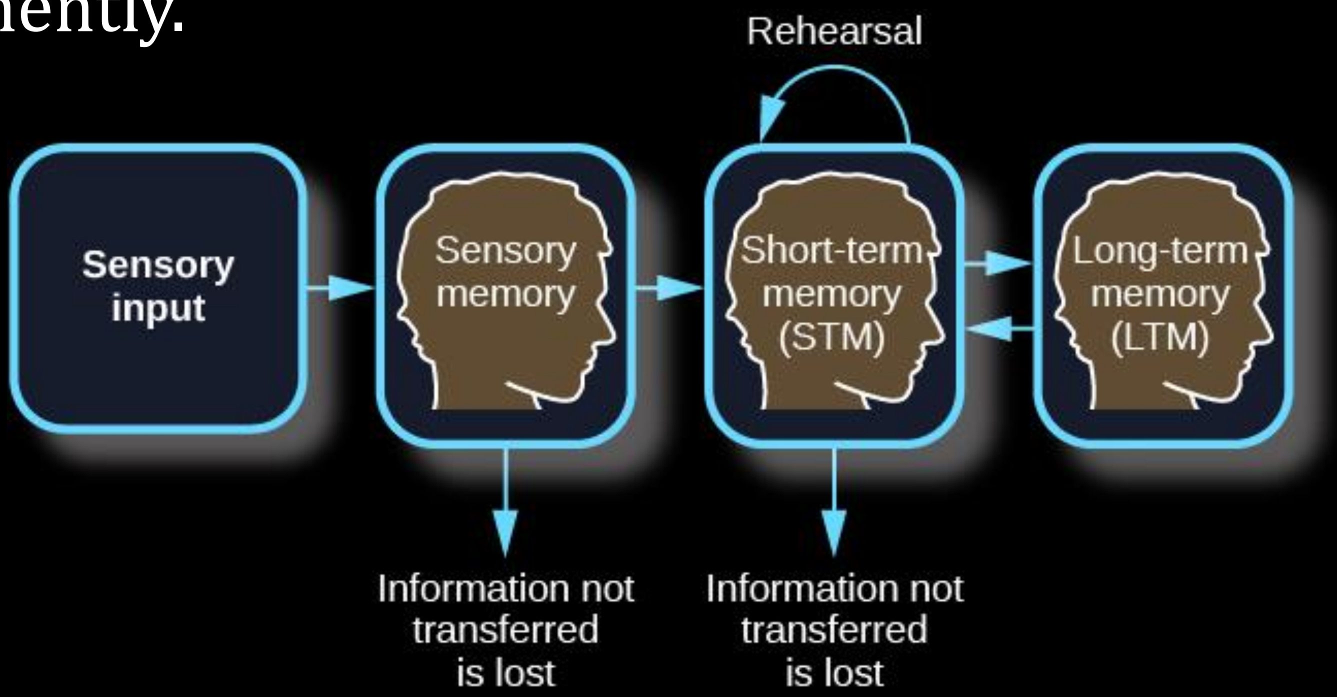
https://www.jneurosci.org/content/40/46/8782



https://www.brainfacts.org/thinking-sensing-and-behaving/learning-and-memory/2018/storing-memories-in-your-synapses-101118

2.11 Learning Tasks: Memory

- Memory may be divided into "short-term" and "long-term" memory, depending on the retention time.
- Short-term memory refers to a compilation of knowledge representing the "current" state of the environment.
- Long-term memory, on the other hand, refers to knowledge stored for a long time or permanently.

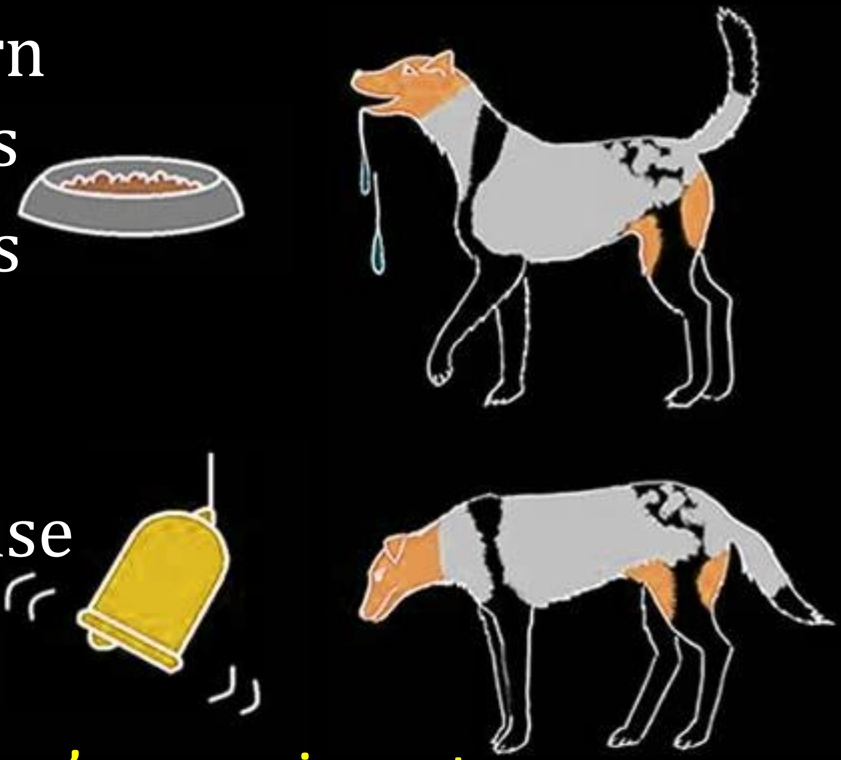


https://courses.lumenlearning.com/wmopen-psychology/chapter/reading-storage/

2.11 Learning Tasks: Memory

Associative Memory

- Associative memory bears the following characteristics:
  - Both the stimulus (key) pattern and the response (stored) pattern of an associative memory consist of data vectors.
  - Information is stored by setting up a spatial pattern of neural activities across large number of neurons
  - Information contained in a stimulus determines its storage location in memory and an address for its retrieval.
  - Memory exhibits a high degree of resistance to noise and damage.



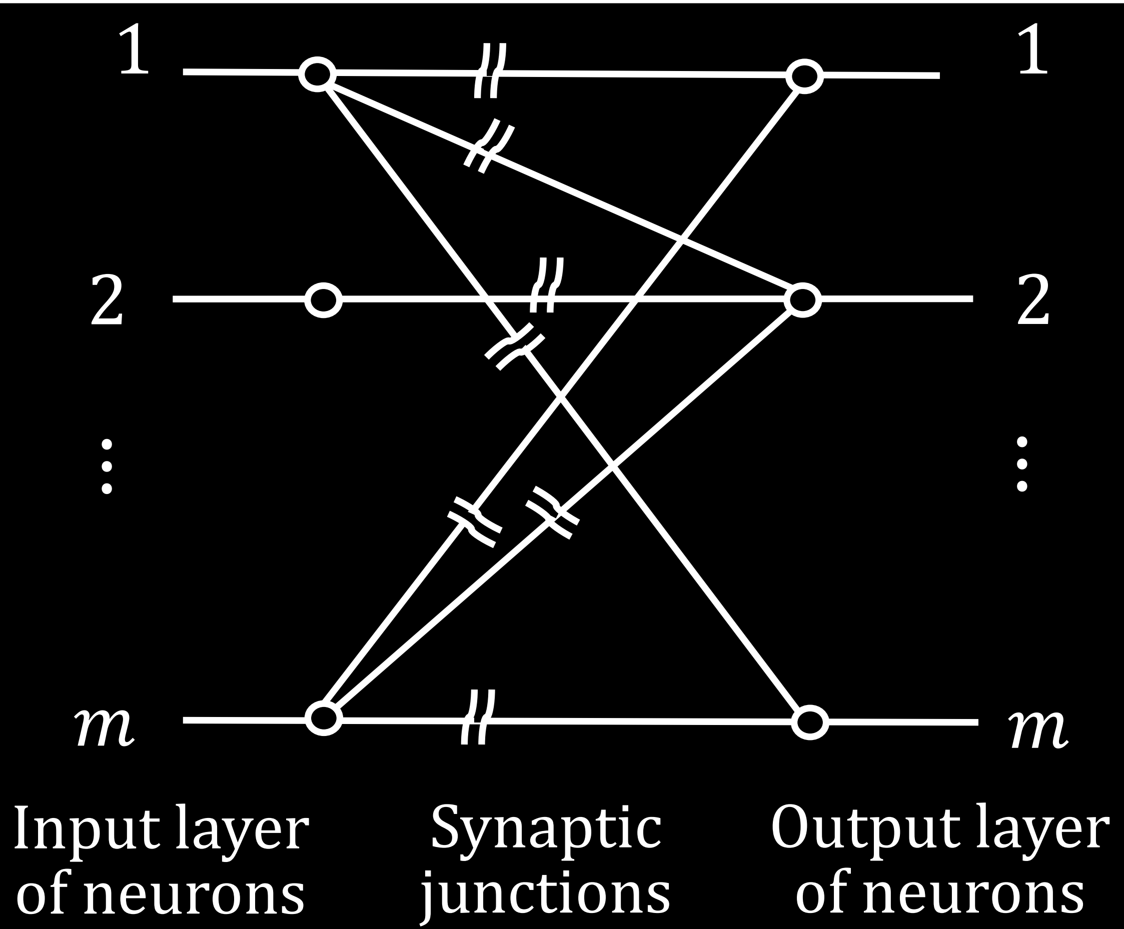
Pavlov’s experiment

2.11 Learning Tasks: Memory

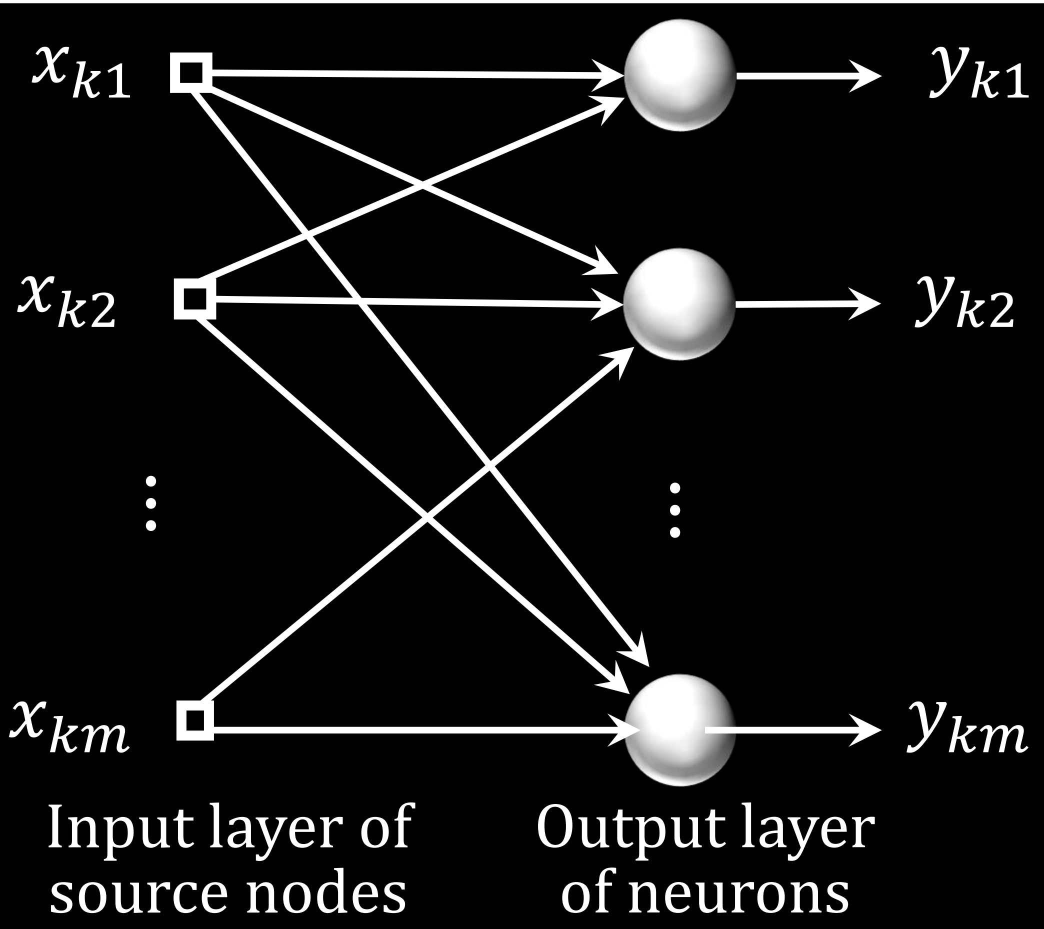
Distributed Memory

- In a distributed memory, the basic issue of interest is the simultaneous or near-simultaneous activities of many different neurons, which are the result of external or internal stimuli.
- Memory performs a distributed mapping that transforms an activity pattern in the input space into another activity pattern in the output space.
- We may illustrate some important properties of a distributed memory mapping by considering an idealized neural network that consists of two layers of neurons as shown in Fig. 2.17.

2.11 Learning Tasks: Memory



(a) Associative memory model component of a nervous system

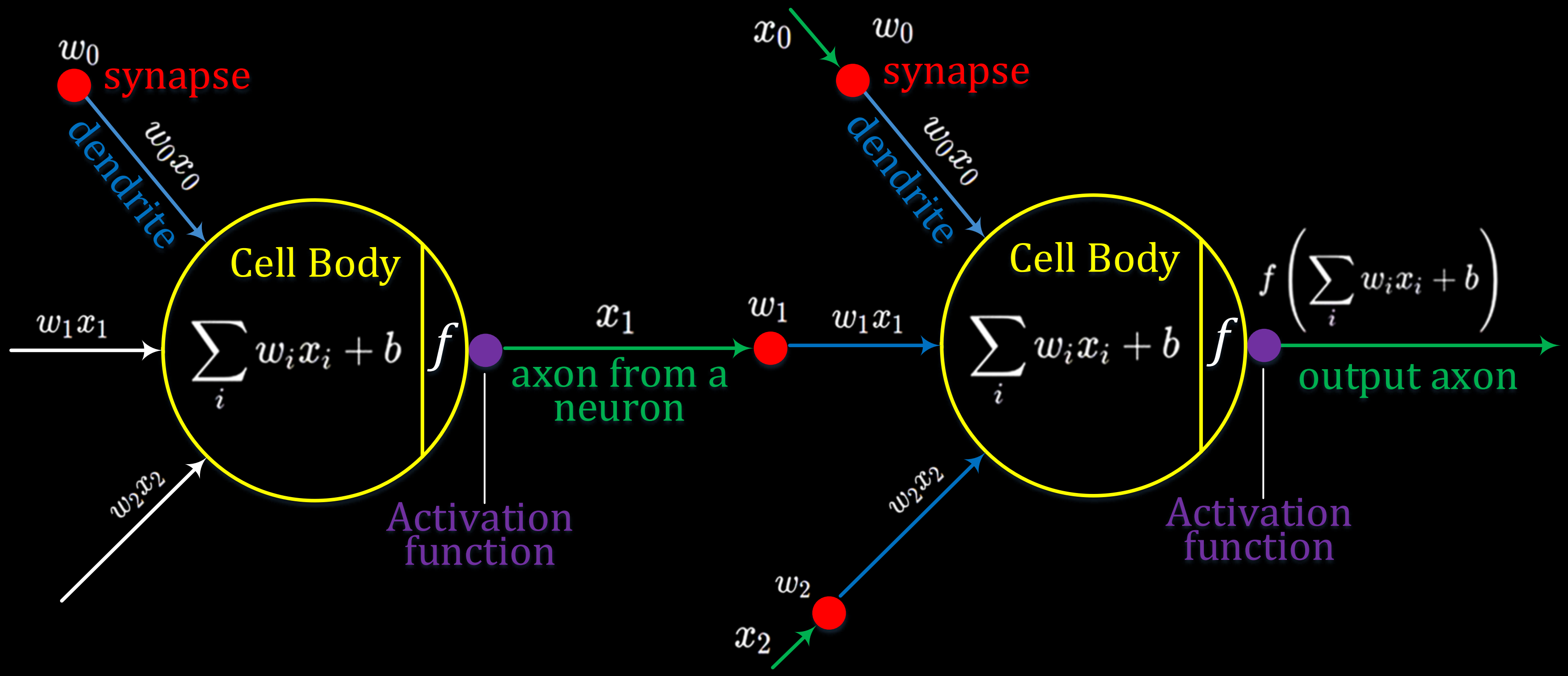


(b) Associative memory model using artificial neurons

Fig. 2.17 Associative memory models.

2.11 Learning Tasks: Memory

- In the *model component of a nervous system,* each neuron in the input layer is connected to every one of the neurons in the output layer.
- In the model of Fig. 2.17a, a single ideal junction is used to represent the integrated effect of all the synaptic contacts between the dendrites of a neuron in the input layer and the axon branches of a neuron in the output layer.



2.11 Learning Tasks: Memory

- In the Fig. 2.17b, an input layer of source nodes and an output layer of neurons act as computation nodes.
- Neuron acts as a linear combiner, as depicted in the signal-flow graph of Fig. 2.18.

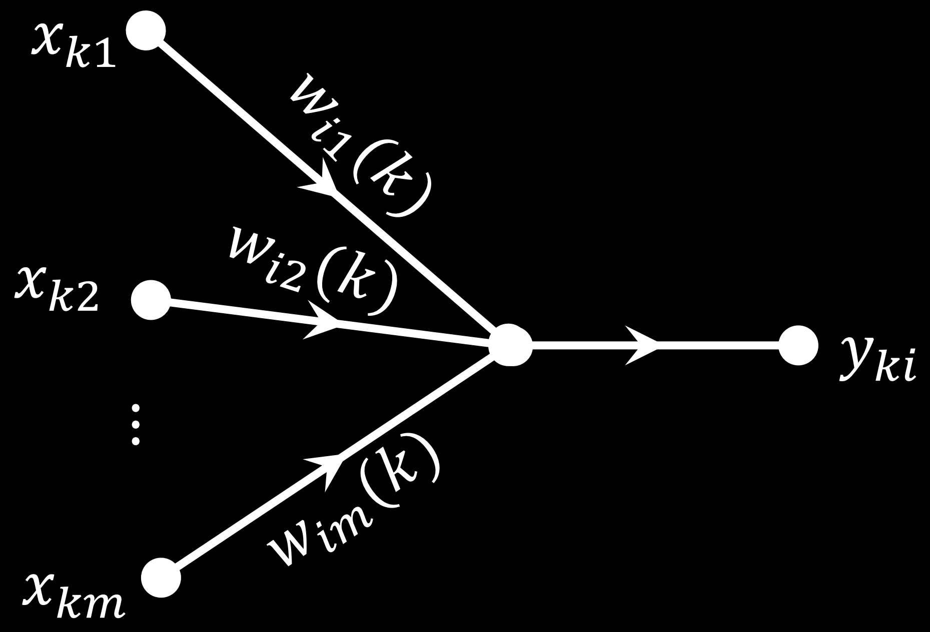


Fig. 2.18 Signal-flow-graph model of a linear neuron labeled *i*.

2.11 Learning Tasks: Memory

- Suppose that an activity pattern x_k occurs in the input layer of the network and that an activity pattern y_k occurs in the output layer.
- For learning from the association between the patterns x_k and y_k, where the patterns x_k and y_k are represented by vectors as:

$$\mathbf{x}_k = [x_{k1}, x_{k2}, \dots, x_{km}]^T$$

and

$$\mathbf{y}_k = [y_{k1}, y_{k2}, \dots, y_{km}]^T$$

Both with *network dimensionality* of *m,* where *m* equals the number of source nodes in the input layer or neurons in the output layer.

- With the networks of Fig. 2.17 assumed to be linear, the association of key vector x_k with memorized vector y_k may be described in matrix form as:

$$\mathbf{y}_k = \mathbf{W}(k)\mathbf{x}_k, \qquad k = 1, 2, \dots, q \tag{2.27}$$

where $\mathbf{W}(k)$ is a weight matrix determined by the input-output pair $(\mathbf{x}_k, \mathbf{y}_k)$.

2.11 Learning Tasks: Memory

- To develop a description of the weight matrix $\mathbf{W}(k)$, consider Fig. 2.18, where the output y_{ki} of neuron i due to the combined action of the elements of the key pattern $\mathbf{x}_k$ applied as stimulus to the input layer is given by

$$y_{ki} = \sum_{j=1}^{m} w_{ij}(k) x\_kj, \qquad i=1, 2, \dots, m \tag{2.28}$$

where the $w_{ij}(k)$, $j = 1, 2, \dots, m$, are the synaptic weights of neuron i corresponding to the kth pair of associated patterns. The output $\mathbf{y}_{ki}$ expressed in matrix form as

$$y_{ki} = [w_{i1}(k), w_{i2}(k), \dots w_{im}(k)] \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{km} \end{bmatrix}, i=1, 2, \dots, m \tag{2.29}$$

2.11 Learning Tasks: Memory

- Column vector on the right-hand side of (2.29) is recognized as key vector $\mathbf{x}_k$. By substituting (2.29) in the definition of the m-by-1 stored vector $\mathbf{y}_k$, we get

$$\begin{bmatrix} y_{k1} \\ y_{k2} \\ \vdots \\ y_{km} \end{bmatrix} = \begin{bmatrix} w_{11}(k) & w_{12}(k) & \dots & w_{1m}(k) \\ w_{21}(k) & w_{22}(k) & \dots & w_{2m}(k) \\ \vdots & \vdots & & \vdots \\ w_{m1}(k) & w_{m2}(k) & \dots & w_{mm}(k) \end{bmatrix} \begin{bmatrix} x_{k1} \\ x_{k2} \\ \vdots \\ x_{km} \end{bmatrix} \tag{2.30}$$

(2.30) is the expanded form of the matrix transformation or mapping described in (2.27). In particular, the m-by-m weight matrix $\mathbf{W}(k)$ is defined by

$$\begin{bmatrix} w_{11}(k) & w_{12}(k) & \dots & w_{1m}(k) \\ w_{21}(k) & w_{22}(k) & \dots & w_{2m}(k) \\ \vdots & \vdots & & \vdots \\ w_{m1}(k) & w_{m2}(k) & \dots & w_{mm}(k) \end{bmatrix} \tag{2.31}$$

where k = 1, 2, …q.

2.11 Learning Tasks: Memory

- The m-by-m memory matrix describes the summation of the weight matrices

$$\mathbf{M} = \sum_{k=1}^{q} \mathbf{W}(k) \qquad (2.32)$$

- The memory matrix **M** defines the connectivity between the input and output layers of the *associative memory* and represents total experience gained by the memory.
- In recursive form

$$\mathbf{M}_k = \mathbf{M}_{k-1} + \mathbf{W}(k), \qquad k = 1, 2, \dots, q \qquad (2.33)$$

where the initial value $\mathbf{M}_0 = 0$ (i.e., the synaptic weights in the memory are all initially zero) and $\mathbf{M}_q = \mathbf{M}$.

2.11 Learning Tasks: Correlation Matrix Memory

- The associative memory of Fig. 2.17b learned the memory matrix **M** through the associations of key and memorized patterns described by $\mathbf{x}_k \rightarrow \mathbf{y}_k$ where $k = 1, 2, \ldots, q$.

- An estimate of the memory matrix **M** in terms of these patterns

$$\hat{\mathbf{M}} = \sum_{k=1}^{q} \mathbf{y}_k \mathbf{x}_k^T \quad m\text{-by-}m \text{ matrix} \qquad (2.34)$$

- This outer product $\mathbf{y}_k \mathbf{x}_k^T$ is an "estimate" of the weight matrix $\mathbf{W}(k)$ that maps the output pattern $\mathbf{y}_k$ onto the input pattern $\mathbf{x}_k$.

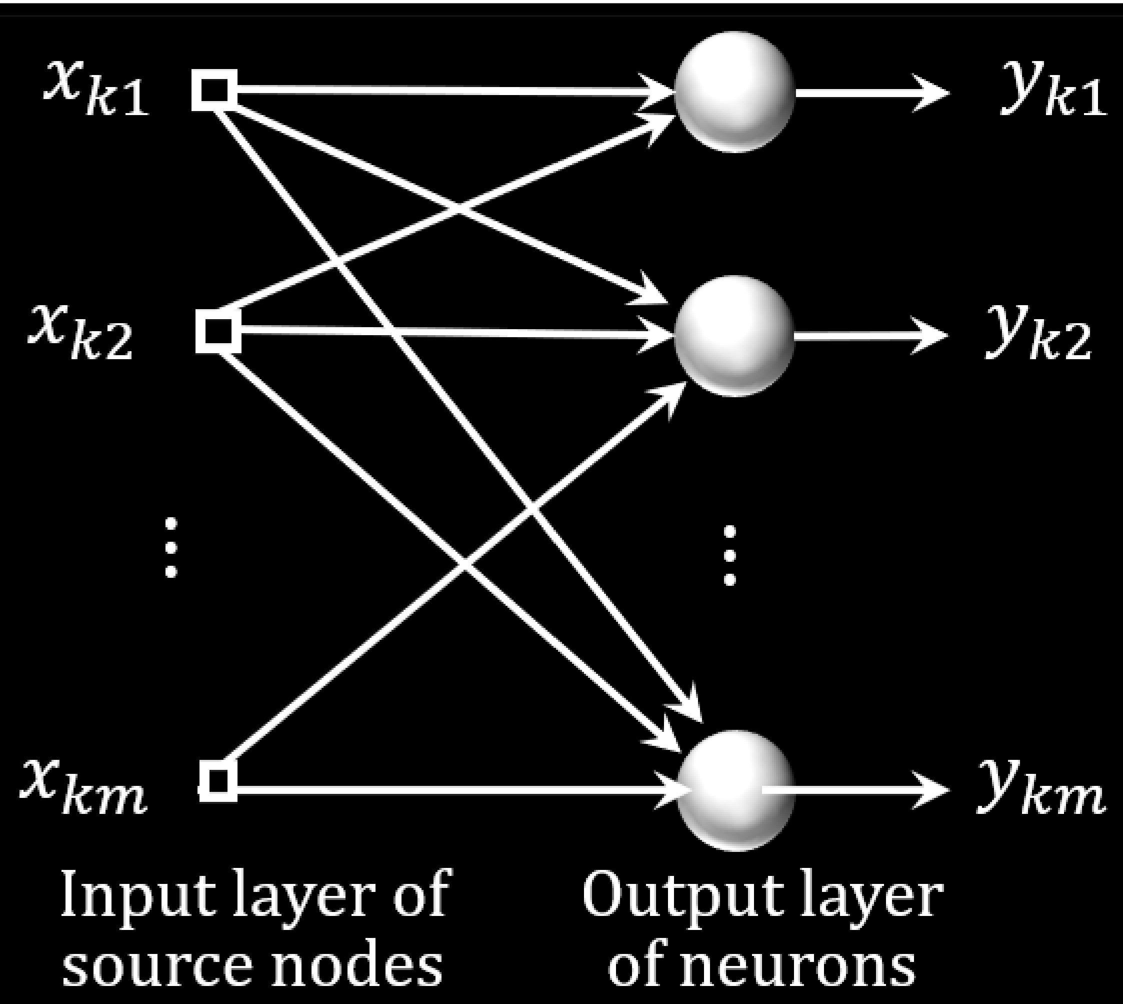


Fig. 2.17(b) Associative memory model using artificial neurons

2.11 Learning Tasks: Correlation Matrix Memory

- Equation (2.34) may be reformulated in the equivalent form

$$\hat{\mathbf{M}} = [\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_q] \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ . \\ \mathbf{x}_q^T \end{bmatrix} = \mathbf{Y}\mathbf{X}^T \qquad (2.35)$$

where,

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_q] \qquad (2.36)$$

and

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_q] \qquad (2.37)$$

The m-by-q matrix $\mathbf{X}$ is composed of the entire set of key patterns is called the *key matrix* and the m-by-q matrix $\mathbf{Y}$ is composed of the corresponding set of memorized patterns and it is called the *memorized matrix*.

2.11 Learning Tasks: Correlation Matrix Memory

- Equation (2.35) may also be restructured in the form of a recursion as:

$$\hat{\mathbf{M}}_k = \hat{\mathbf{M}}_{k-1} + \mathbf{y}_k \mathbf{x}_k^T, \quad k = 1, 2, \dots, q \qquad (2.38)$$

- A signal-flow graph representation of this recursion is depicted in Fig. 2.19.

- According to this signal-flow graph and the recursive formula of Eq. (2.38),
- The matrix $\hat{\mathbf{M}}_k$ is updated in the light of a new association performed by the memory on the patterns $\mathbf{x}_k$ and $\mathbf{y}_k$.

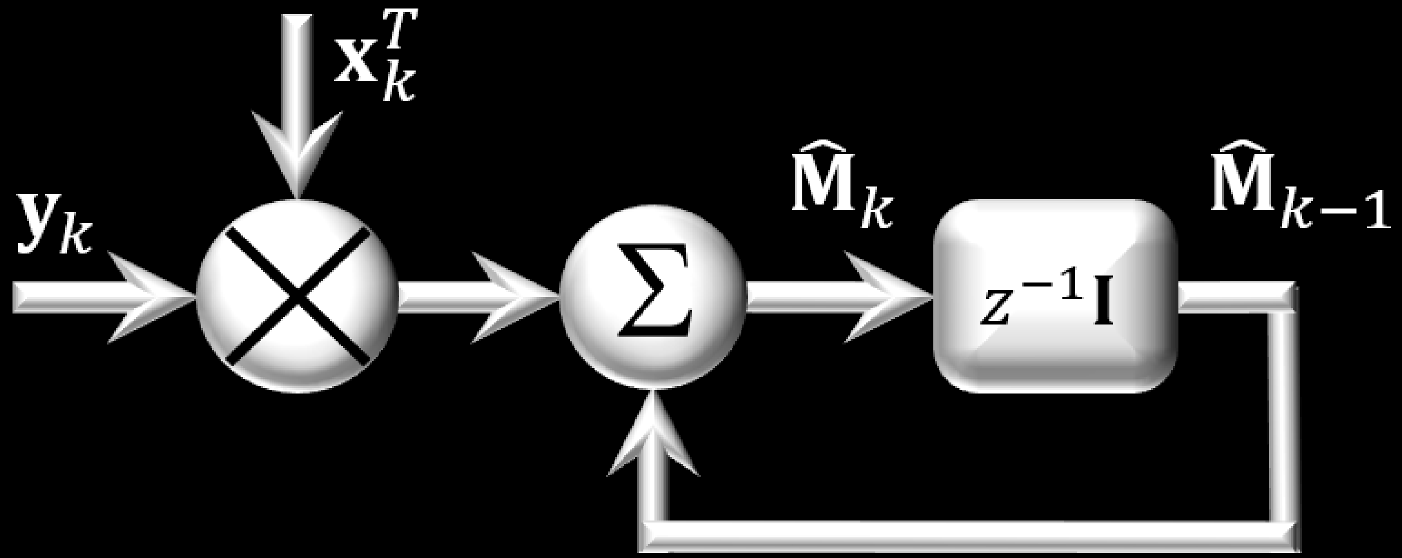


Fig. 2.19 Signal-flow graph representation of Eq. (2.38).

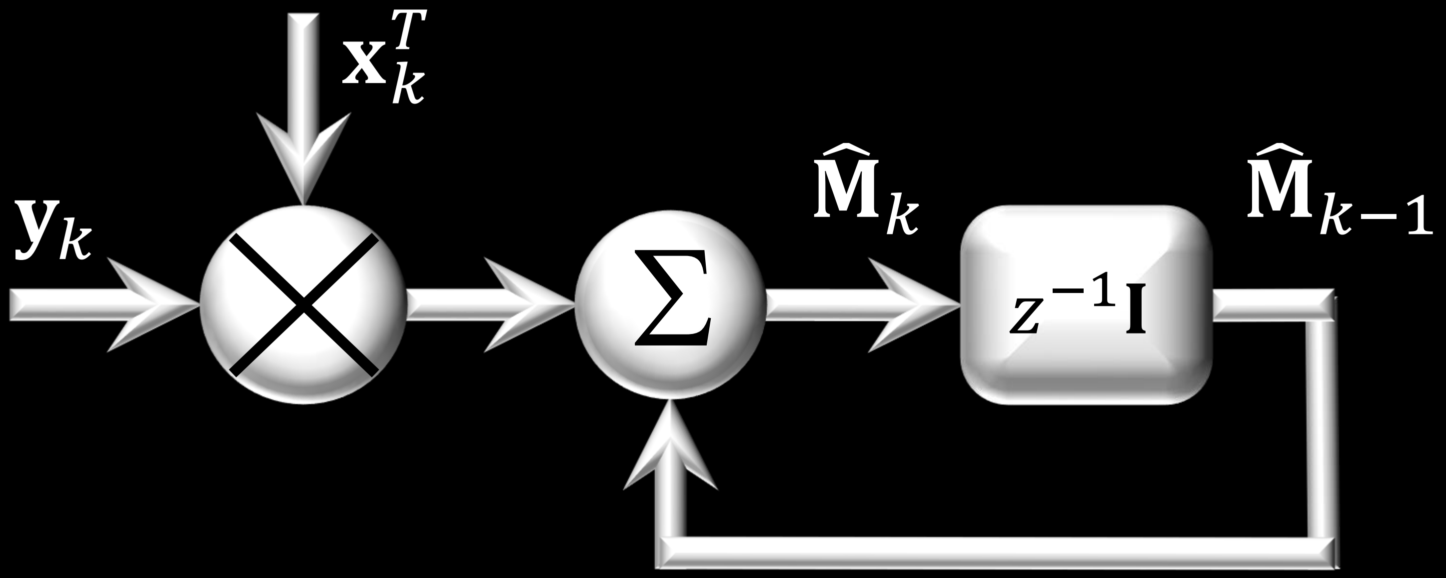


Fig. 2.19 Signal-flow graph representation of Eq. (2.38).

2.11 Learning Tasks: Memory - Recall

- The fundamental problem posed by the use of an *associative memory* is the address and recall of patterns stored in memory.
- To explain, let the memory matrix of an associative memory denoted by M, which has been completely learned through its exposure to q pattern associations as shown in (2.34).
- Let the stimulus $\mathbf{x}_j$ yield the random response

$$\mathbf{y} = \widehat{M}\mathbf{x}_j \tag{2.39}$$

Substituting Eq. (2.34) in (2.39),

$$\mathbf{y} = \sum_{k=1}^{m} \mathbf{y}_k \mathbf{x}_k^T \mathbf{x}_j = \sum_{k=1}^{m} (\mathbf{x}_k^T \mathbf{x}_j)\mathbf{y}_k \tag{2.40}$$

Here, $\mathbf{x}_k^T\mathbf{x}_j$ is the inner product

2.11 Learning Tasks: Memory - Recall

- Rewriting (2.40) as

$$\mathbf{y} = \mathbf{x}_j^T \mathbf{x}_j \mathbf{y}_j + \sum_{\substack{k=1 \\ k \neq j}}^{m} (\mathbf{x}_k^T \mathbf{x}_j)\, \mathbf{y}_k \tag{2.41}$$

The unit energy

$$E_k = \sum_{l=1}^{m} x_{kl}^2 = \mathbf{x}_k^T \mathbf{x}_k = 1, \qquad k = 1, 2, \dots, q \tag{2.42}$$

Then the response of the memory to stimulus (key pattern) $\mathbf{x}_j$ is

$$\mathbf{y} = \mathbf{x}_j^T \mathbf{x}_j \mathbf{y}_j + \sum_{\substack{k=1 \\ k \neq j}}^{m} (\mathbf{x}_k^T \mathbf{x}_j)\, \mathbf{y}_k = \mathbf{y}_j + \sum_{\substack{k=1 \\ k \neq j}}^{m} (\mathbf{x}_k^T \mathbf{x}_j)\, \mathbf{y}_k = \mathbf{y}_j + \mathbf{v}_j \tag{2.43}$$

$$\mathbf{v}_j = \sum_{\substack{k=1 \\ k \neq j}}^{m} (\mathbf{x}_k^T \mathbf{x}_j)\, \mathbf{y}_k \tag{2.44}$$

Here, $\boldsymbol{y} = \mathbf{y}_j + \mathbf{v}_j$ = desired response vector + noise vector)

Recall errors are due to the noise vector $\mathbf{v}_j$

2.11 Learning Tasks: Memory - Recall

- The cosine of the angle between $\mathbf{x}_j$ and $\mathbf{x}_k$

$$\cos(\mathbf{x}_k, \mathbf{x}_j) = \frac{\mathbf{x}_k^T \mathbf{x}_j}{\|\mathbf{x}_k\| \|\mathbf{x}_j\|} \tag{2.45}$$

The Euclidean distance

$$\|\mathbf{x}_j\| = \left(\mathbf{x}_k^T \mathbf{x}_j\right)^{1/2} = (E_k)^{1/2} \tag{2.46}$$

(2.45) cvan be reduced to

$$\cos(\mathbf{x}_k, \mathbf{x}_j) = \mathbf{x}_k^T \mathbf{x}_j \tag{2.47}$$

Then the noise vector is

$$\mathbf{v}_j = \sum_{\substack{k=1 \\ k \neq j}}^{m} \cos(\mathbf{x}_k, \mathbf{x}_j)\, \mathbf{y}_k \tag{2.48}$$

2.11 Learning Tasks: Memory - Recall

- When $\mathbf{x}_k$ and $\mathbf{x}_j$ are orthogonal

$$\cos(\mathbf{x}_k, \mathbf{x}_j) = 0, \quad k \neq j \tag{2.49}$$

And also, the noise vector $\mathbf{v}_j = 0$ and the response $\mathbf{y} = \mathbf{y}_j$

- The memory associates perfectly if the key vectors from an orthonormal set satisfy the following pair of conditions

$$\mathbf{x}_k^T \mathbf{x}_j = \begin{cases} 1 & k = j \\ 0 & k \neq j \end{cases} \tag{2.50}$$

2.11 Learning Tasks: Memory - Recall

Exercise 1: What is then the limit on the storage capacity of the associative memory? (Answer: pp. 104, par 2)

2.11 Learning Tasks: Memory - Recall

- The memory occasionally recognizes and associates patterns never seen or associated before. To illustrate this property of an associative memory, consider a set of key patterns

$$\{\mathbf{x}_{key}\}:\ \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_q$$

and a corresponding set of memorized patterns

$$\{\boldsymbol{y}_{mem}\}:\ \mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_q$$

- The community is defined as the closeness of the key patterns in a linear signal space give by $\{\mathbf{x}_{key}\}$ as the lower bound on the inner products

$$\mathbf{x}_k^T\mathbf{x}_j$$

2.11 Learning Tasks: Memory - Recall

- The response of the memory $\mathbf{y}$ to a stimulus $\mathbf{x}_j$ selected from the set $\{\mathbf{x}_{key}\}$ is given by Eq. (2.39), where it is assumed that each pattern in the set $\{\mathbf{x}_{key}\}$ is a unit vector of unit energy, assuming that

$$\mathbf{x}_k^T \mathbf{x}_j > \gamma \tag{2.51}$$

- If the lower bound γ is large enough, the memory may fail to distinguish the response $\mathbf{y}$ from that of any other key pattern contained in the set $\{\mathbf{x}_{key}\}$

2.11 Learning Tasks: Memory - Recall

- If the key patterns in this set have the form

$$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{v} \tag{2.52}$$

- where $\mathbf{v}$ is a stochastic vector, it is likely that the memory will recognize $\mathbf{x}_0$ and associate with it a vector $\mathbf{y}_0$ rather than any of the actual pattern pairs used to train it in the first place;
- $\mathbf{x}_0$ and $\mathbf{y}_0$ denote a pair of patterns never seen before. This phenomenon may be termed *animal logic*.

2.12 Adaptation

- Space and time are two fundamental dimensions of the learning process.
- Animal’s inherent capacity to represent the temporal structure of experience makes it possible to adapt their behavior to the temporal structure of an event in its behavioral space.

https://www.sciencenewsforstudents.org/article/what-bats-see-when-they-probe-the-world-with-sound

2.12 Adaptation

- When a neural network operates in a stationary environment (i.e., an environment whose statistical characteristics do not change with time),
  - the essential statistics of the environment can be learned by the network under the supervision of a teacher.
- The synaptic weights of the network can be computed by having the network undergo a training session with a set of data that is representative of the environment.
- Once the training process has completed, the synaptic weights of the network should capture the underlying statistical structure of the environment, which would justify "freezing" their values thereafter.
- Thus, a learning system relies on memory, in one form or another, to recall and exploit past experiences.

2.12 Adaptation

- Frequently, however, the environment of interest is nonstationary, which means that the statistical parameters of the information-bearing signals generated by the environment vary with time.
- It is desirable for a neural network to continually adapt its free parameters to variations in the incoming signals in a real-time fashion.
- Thus, an adaptive system responds to every distinct input as a novel one.
- This form of learning is called **continuous learning** or **learning-on-the-fly.**

2.12 Adaptation

- Linear adaptive filters, built around a linear combiner are designed to perform continuous learning.
- They are widely used in such diverse applications as radar, sonar, communications, seismology, and biomedical signal processing.
- A neural network adapts its behavior to the varying temporal structure of the incoming signals in its behavioral space since
  - statistical characteristics of a nonstationary process usually change slowly enough for the process to be considered pseudo-stationary over a window of time.

2.12 Adaptation

- Adaptation examples include:
  - The mechanism responsible for the production of a speech signal may be considered essentially stationary over a period of 10 to 30 milliseconds,
  - Radar returns from an ocean surface remain essentially stationary over a period of several seconds,
  - weather related data may be viewed as essentially stationary over a period of minutes,
  - stock market data may be considered as essentially stationary over a period of days.

Consider skipping this section

?

2.13 Statistical Nature of the Learning Process

- This part of the chapter deals with statistical aspects of learning
  - which focuses on the deviation between a "target" function $f(\mathbf{x})$ and the "actual" function $F(x, w)$ realized by the neural network.

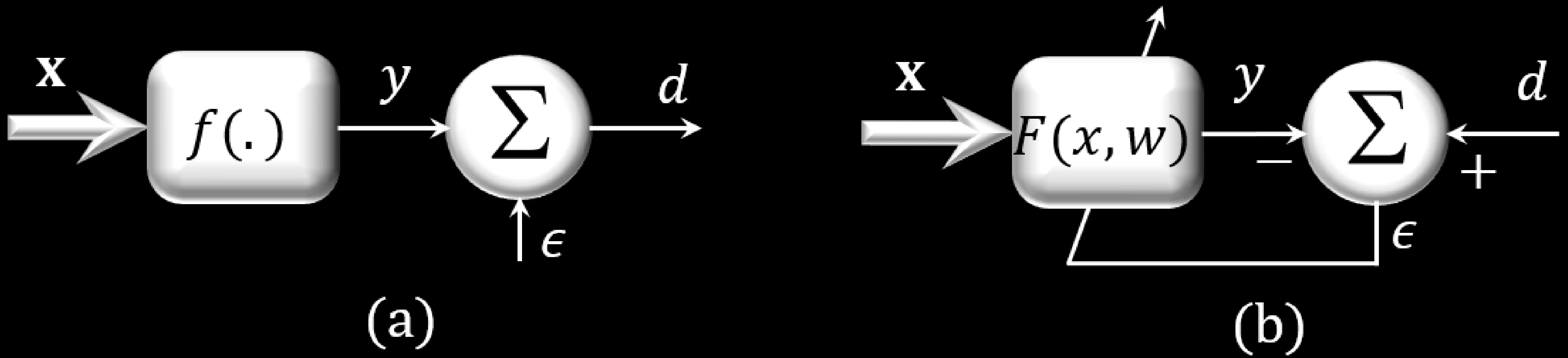


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

- Suppose we have N realizations of random vector $\mathbf{X}$ denoted by $\{\mathbf{x}_i\}_{i=1}^{N}$ and a corresponding set of realizations of the random scalar D denoted by $\{d_i\}_{i=1}^{N}$.
- These realizations (measurements) constitute training sample denoted by

$$\mathcal{T}\{\mathbf{x}_i, d_i\}_{i=1}^{N} \tag{2.53}$$

- Since there is no knowledge of the exact functional relationship between $\mathbf{X}$ and D, we proceed by proposing the model

$$D = f(\mathbf{X}) + \epsilon \tag{2.54}$$

where $f(\cdot)$ is a deterministic function and ϵ is a random expectational error that represents our "ignorance" about the dependence of D and $\mathbf{X}$.

2.13 Statistical Nature of the Learning Process

- The statistical model described by Eq. (2.54) is called a *regressive model*; it is depicted in Fig. 2.20a.
- The expectational error ϵ is a random variable with zero mean and positive probability of occurrence.

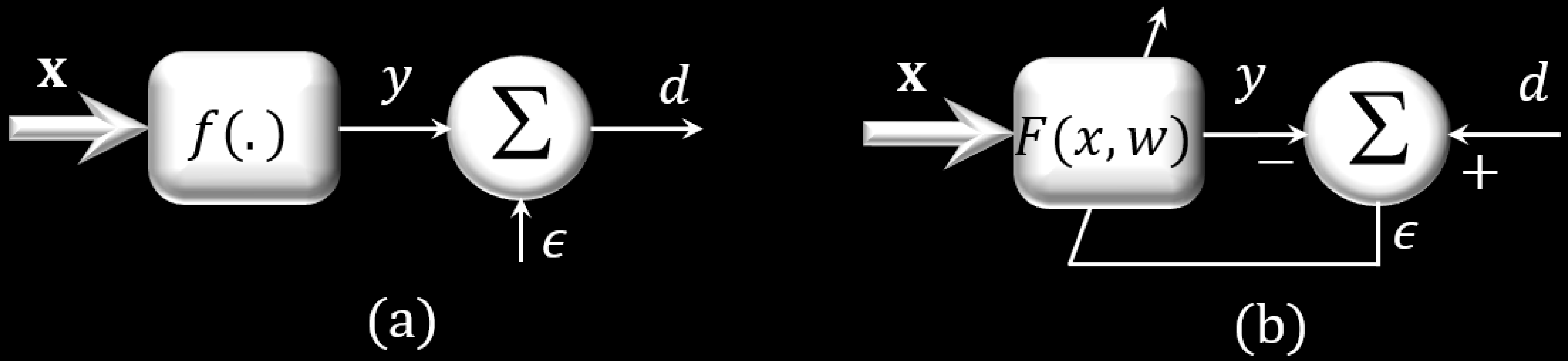


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

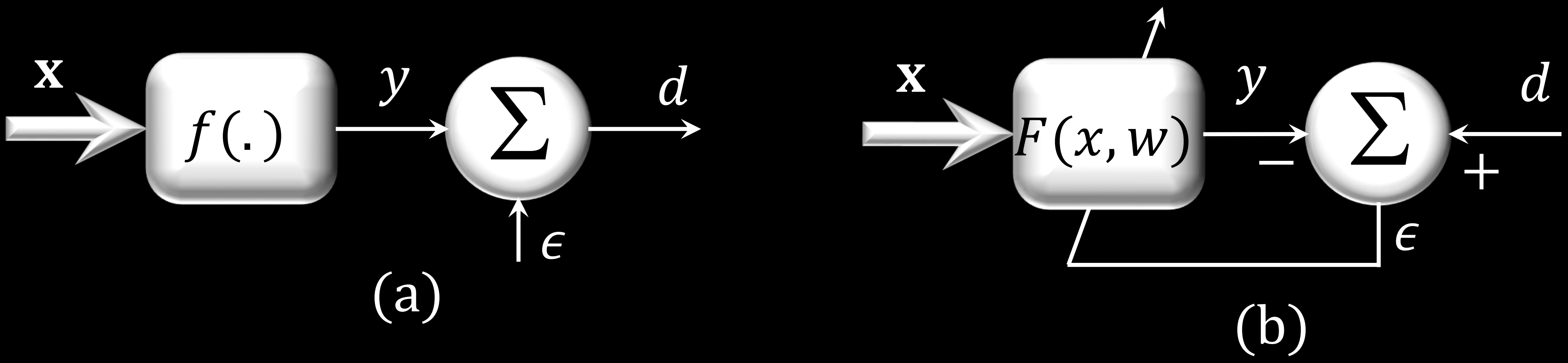


Fig. 2.20 (a) Regressive model (mathematical).
(b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

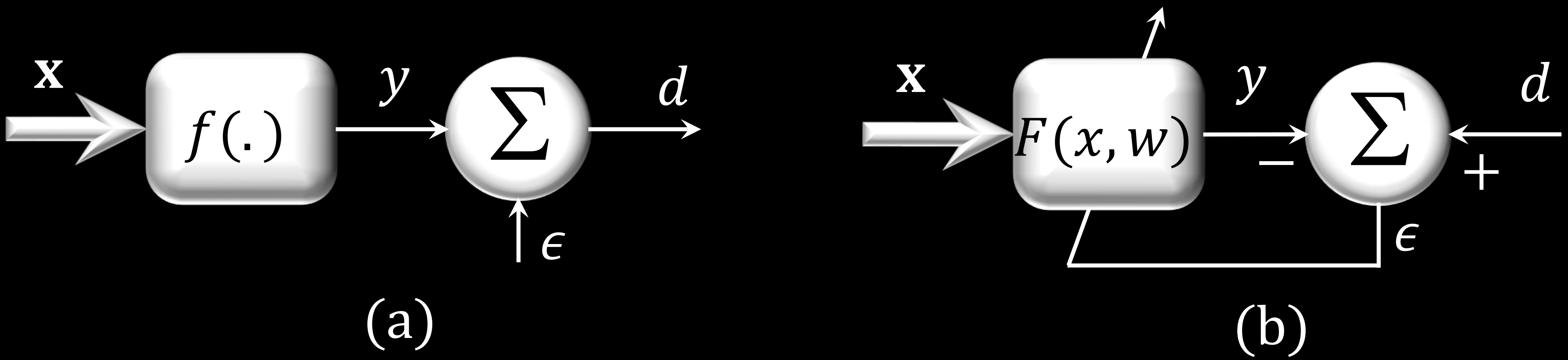


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

- On this basis, the regressive model of Fig. 2.20a has two useful properties:

1. Mean value of expectational error ϵ, given any realization $\mathbf{x}$, is zero; that is,

$$E[\epsilon|\mathbf{x}] = 0 \qquad (2.55)$$

2. Expectational error ϵ is uncorrelated with the regression function $f(\mathbf{X})$, i.e.

$$E[\epsilon f(\mathbf{x})] = 0 \qquad (2.56)$$

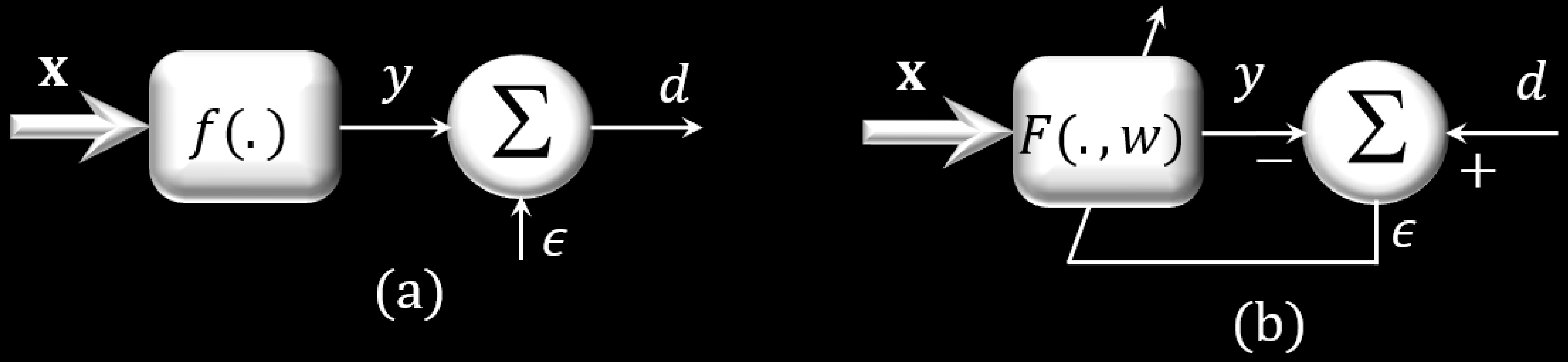


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

- The purpose of the *physical* model of the environment shown in Fig. 2.20b is to encode the empirical knowledge represented by the training sample $\mathcal{T}$ into a corresponding set of synaptic weight vectors, **w**, as shown by

$$\mathcal{T} \rightarrow \mathbf{w} \tag{2.58}$$

- In effect, the neural network provides an "approximation" to the regressive model of Fig. 2.20a.

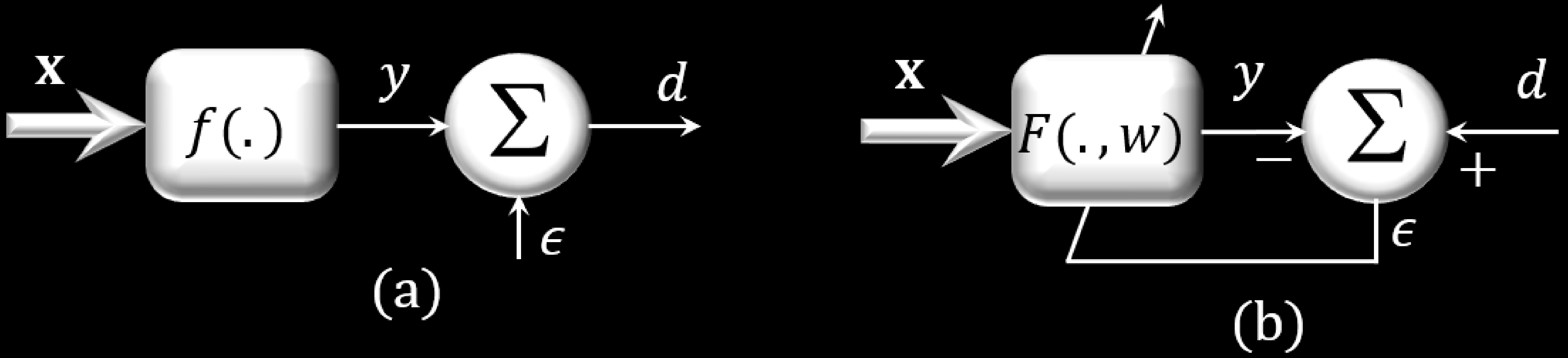


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

- Let the actual response of the neural network, produced in response to the input vector **x**, be denoted by the random variable

$$Y = F(\mathbf{X}, \mathbf{w}) \tag{2.59}$$

- Weight vector **w** is obtained by minimizing the cost function $\mathcal{C}(\mathbf{w})$ defined as squared difference between the desired response d and the actual response y of the neural network, averaged over the entire training data set $\mathcal{T}$:

$$\mathcal{C}(w) = \frac{1}{2}\sum_{i=1}^{N}\left(d_i - F(\mathbf{x}_i, \mathbf{w})\right)^2 \tag{2.60}$$

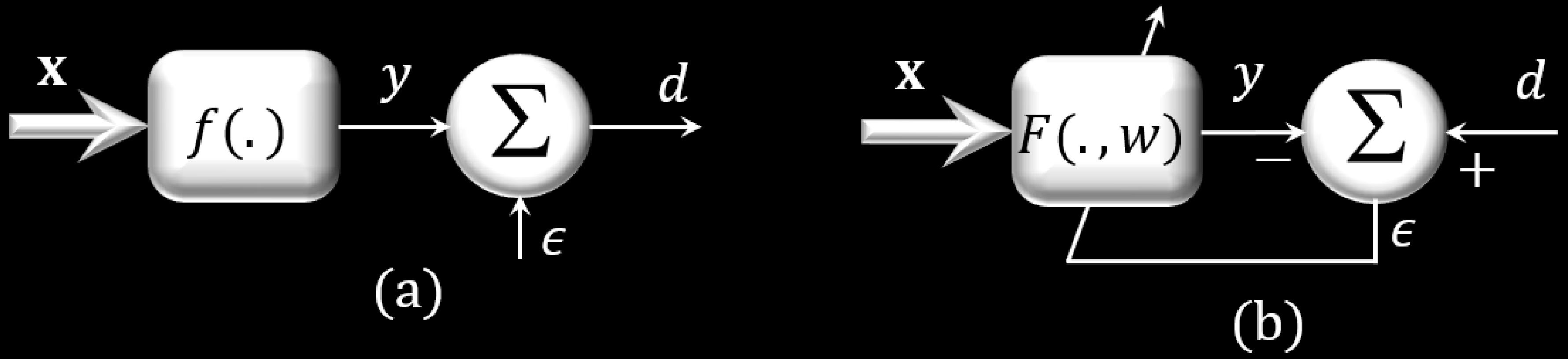


Fig. 2.20 (a) Regressive model (mathematical). (b) Neural network model (physical).

2.13 Statistical Nature of the Learning Process

- Rewriting Eq. (2.60) in the equivalent form

$$C(w) = \frac{1}{2} E_{\mathcal{T}}\left[\left(d - F(\mathbf{x}, \mathcal{T})\right)^2\right] \tag{2.61}$$

- By adding and subtracting $f(\mathbf{x})$ to the argument $\left(d - F(\mathbf{x}, \mathcal{T})\right)$ and rearranging, we get the cost function written as

$$C(w) = \frac{1}{2} E_{\mathcal{T}}[\epsilon^2] + \frac{1}{2} E_{\mathcal{T}}\left[\left(f(\mathbf{x}) - F(\mathbf{x}, \mathcal{T})\right)^2\right] \tag{2.63}$$

- The first term on the right-hand side of Eq. (2.63) is the variance of the expectational (regressive modeling) error ϵ, evaluated over the training sample $\mathcal{T}$.
- The intrinsic error may be ignored as far as the minimization of the cost function $C(w)$ with respect to $\mathbf{w}$ is concerned.

Bias / Variance Dilemma

- Invoking the use of Eq. (2.56), we may redefine the squared distance between $f(\mathbf{x})$ and $F(\mathbf{x}, \mathbf{w})$
  - in other words, the effectiveness of $F(\mathbf{x}, \mathbf{w})$ as a predictor of the desired response d is defined by:

$$L_{av}\big(f(\mathbf{x}), F(\mathbf{x}, \mathbf{w})\big) = E_{\mathcal{T}}[\big(E[D|\mathbf{X} = \mathbf{x}] - F(\mathbf{x}, \mathcal{T})\big)^2] \qquad (2.65)$$

- This expression may also be viewed as the average value of the estimation error between the regression function $f(\mathbf{x}) E[D|\mathbf{X} = \mathbf{x}]$ and the approximating function $F(\mathbf{x}, \mathbf{w})$, evaluated over the entire training sample $\mathcal{T}$.

Bias / Variance Dilemma

- we may reformulate Eq. (2.65) as the sum of two terms:

$$L_{av}\big(f(\mathbf{x}), F(\mathbf{x},\mathbf{w})\big) = B^2(\mathbf{w}) + V(\mathbf{w}) \tag{2.66}$$

We now make two important observations:

1. The term $B(\mathbf{w})$ viewed as an approximation error is the bias of the average value of the approximating function $F(\mathbf{x}, \mathcal{T})$ measured with respect to the regression function $f(\mathbf{x}) = E[D|\mathbf{X} = \mathbf{x}]$.
2. The term $V(\mathbf{w})$, viewed as the estimation error, is the variance of the approximating function $F(\mathbf{x}, \mathbf{w})$, measured over entire training sample $\mathcal{T}$.

- To achieve good overall performance, the bias $B(\mathbf{w})$ and the variance $V(\mathbf{w})$ of the approximating function $F(\mathbf{x}, \mathbf{w}) = F(\mathbf{x}, \mathcal{T})$ would both have to be small.

Bias / Variance Dilemma

- Figure 2.21 illustrates the relations between the target and approximating functions, and shows how the estimation errors, namely bias and variance, accumulate.

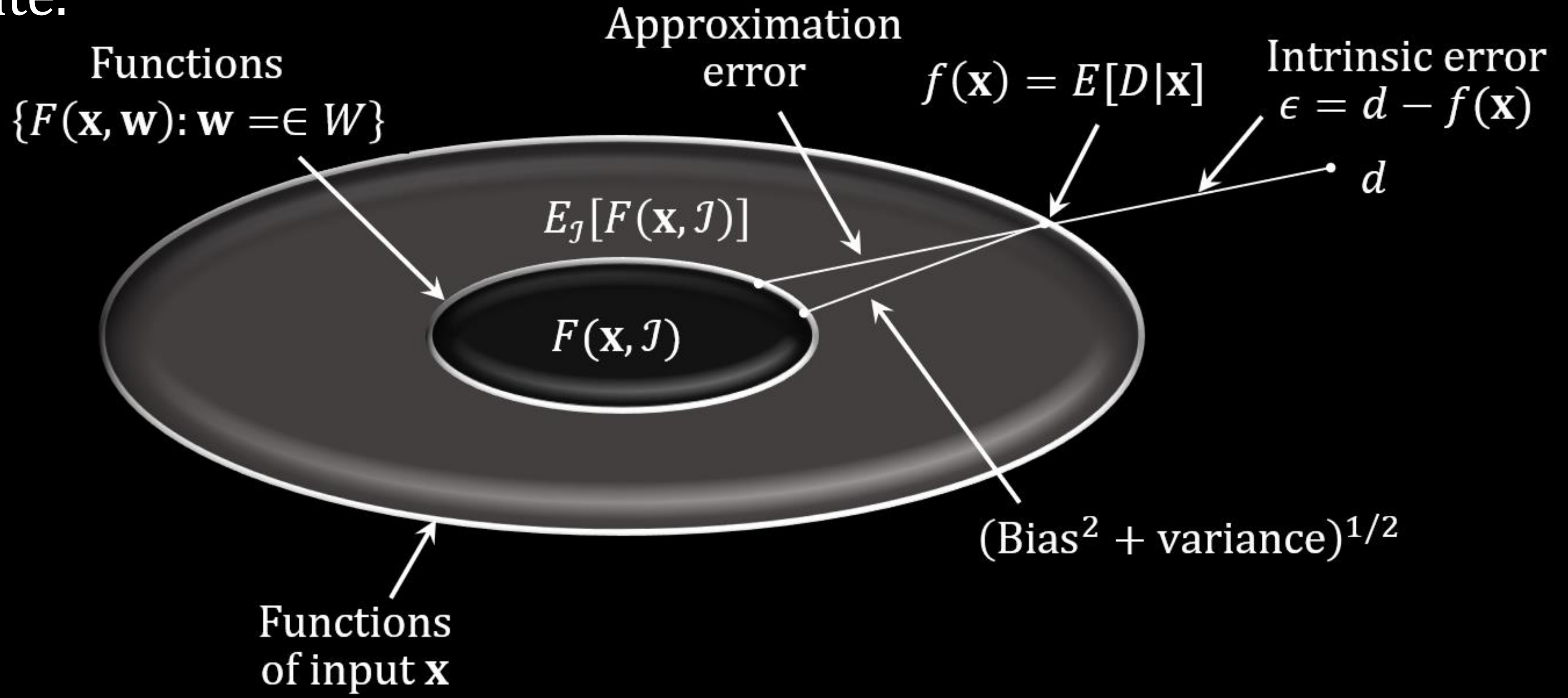


Fig. 2.21 Illustration of the various sources of error in solving the regression problem.

Bias / Variance Dilemma

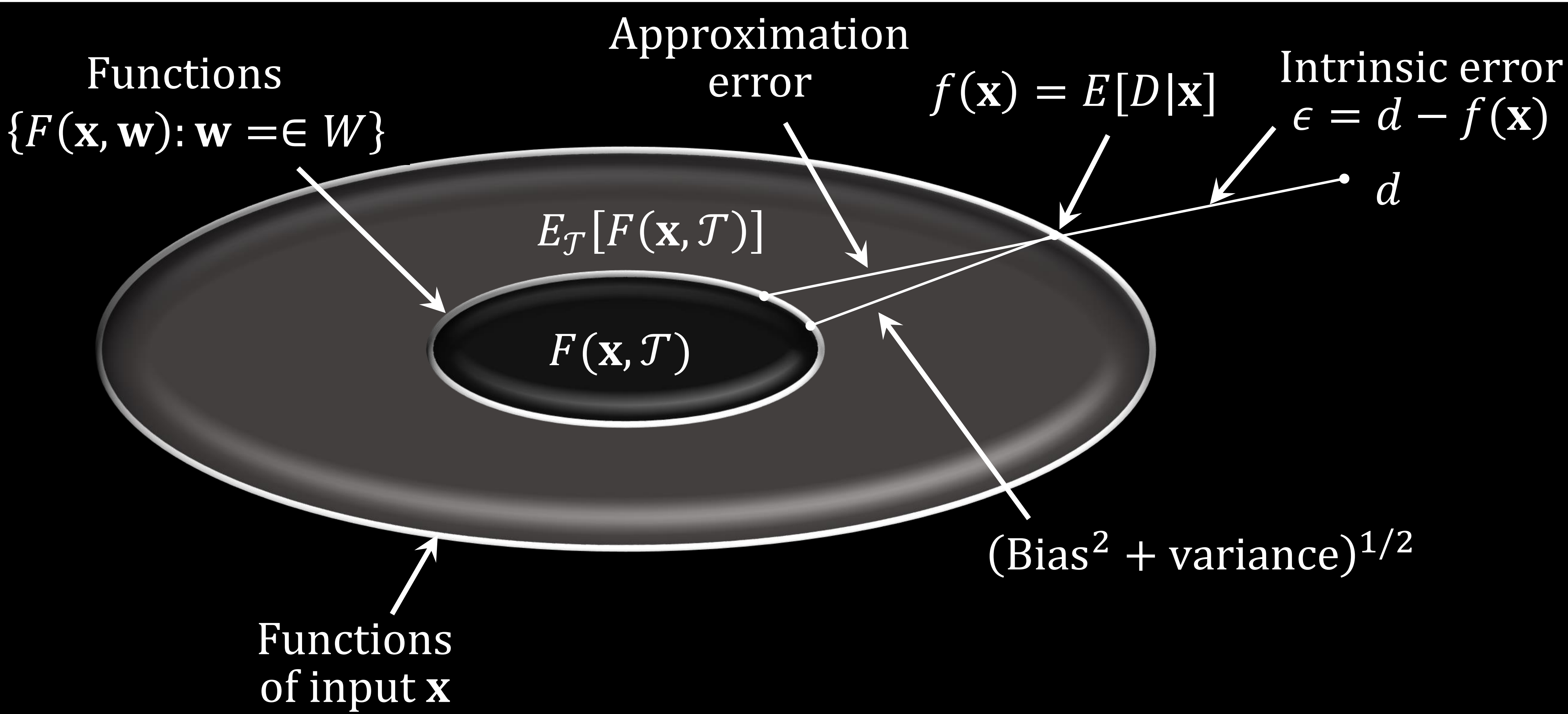


Fig. 2.21 Illustration of the various sources of error in solving the regression problem.