



شبکه‌های عصبی مصنوعی

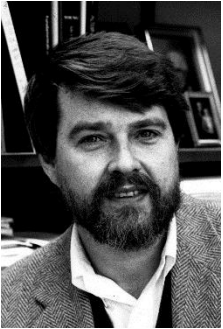
جلسه پنجم:

پرسپترون چند لایه (۱)

(Multi-Layer Perceptron = MLP)

پرسپترون چندلایه (MLP)

پرسپترون چندلایه (MLP)



Rumelhart



Hinton



Williams

- پرسپترون چندلایه (Multi-Layer Perceptron)
توسط روملهارت، هینتون و ویلیامز در سال ۱۹۸۶
ابداع شد.

D.E. Rumelhart, G. E. Hinton, and R. J. Williams,
Learning representations by back-propagating error,
Nature, vol. 323, pp. 533-536, 1986.

پرسپترون چندلایه (MLP)



Rumelhart



Hinton



Williams

- پرسپترون چندلایه (Multi-Layer Perceptron) توسط روملهارت، هینتون و ویلیامز در سال ۱۹۸۶ ابداع شد.

D.E. Rumelhart, G. E. Hinton, and R. J. Williams,
Learning representations by back-propagating error,
Nature, vol. 323, pp. 533-536, 1986.

- این شبکه عصبی در ابتدا با نام شبکه پیشخورد چندلایه (Multi-layer feedforward network) معرفی شد.

پرسپترون چندلایه (MLP)



Rumelhart



Hinton



Williams

- پرسپترون چندلایه (Multi-Layer Perceptron) توسط روملهارت، هینتون و ویلیامز در سال ۱۹۸۶ ابداع شد.

D.E. Rumelhart, G. E. Hinton, and R. J. Williams,
Learning representations by back-propagating error,
Nature, vol. 323, pp. 533-536, 1986.

- این شبکه عصبی در ابتدا با نام شبکه پیشخورد چندلایه (Multi-layer feedforward network) معرفی شد.

- نوع یادگیری در این نوع شبکه از نوع یادگیری با نظارت است.

پرسپترون چندلایه (MLP)



Rumelhart



Hinton



Williams

- پرسپترون چندلایه (Multi-Layer Perceptron) توسط روملهارت، هینتون و ویلیامز در سال ۱۹۸۶ ابداع شد.

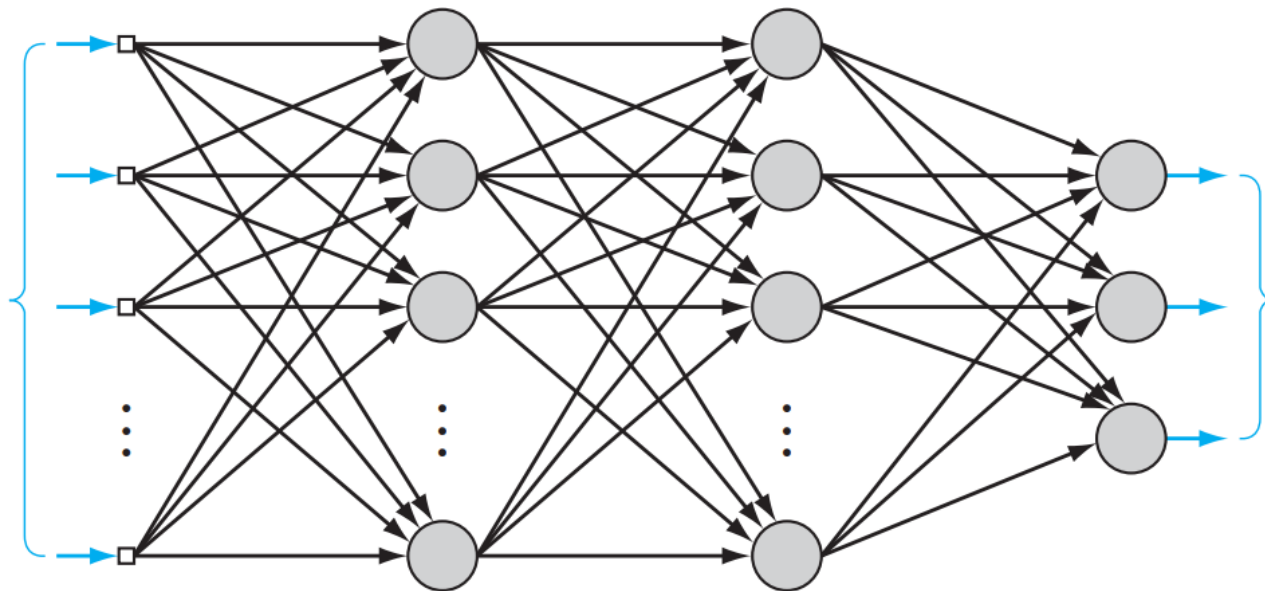
D.E. Rumelhart, G. E. Hinton, and R. J. Williams,
Learning representations by back-propagating error,
Nature, vol. 323, pp. 533-536, 1986.

- این شبکه عصبی در ابتدا با نام شبکه پیشخورد چندلایه (Multi-layer feedforward network) معرفی شد.

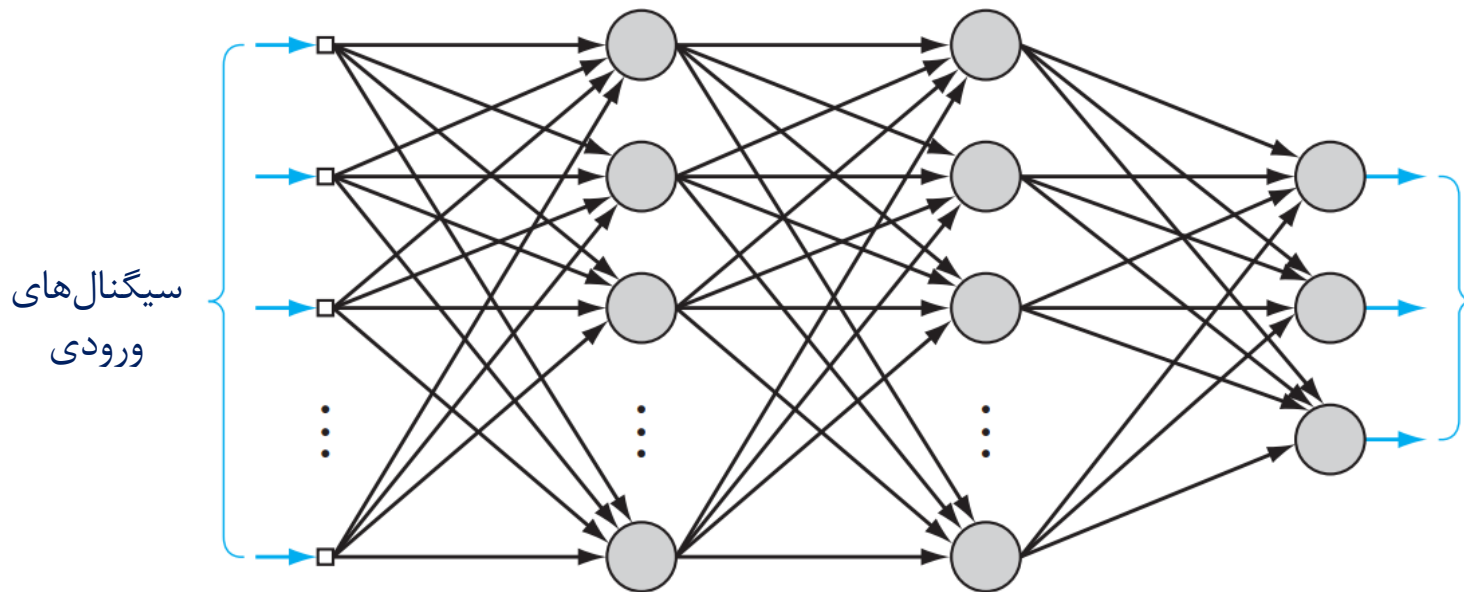
- نوع یادگیری در این نوع شبکه از نوع یادگیری با نظارت است.

- الگوریتم یادگیری در شبکه از نوع یادگیری تصحیح خطا است که در این جا به آن «الگوریتم پس-انتشار خطا» (Error back-propagation) می گویند.

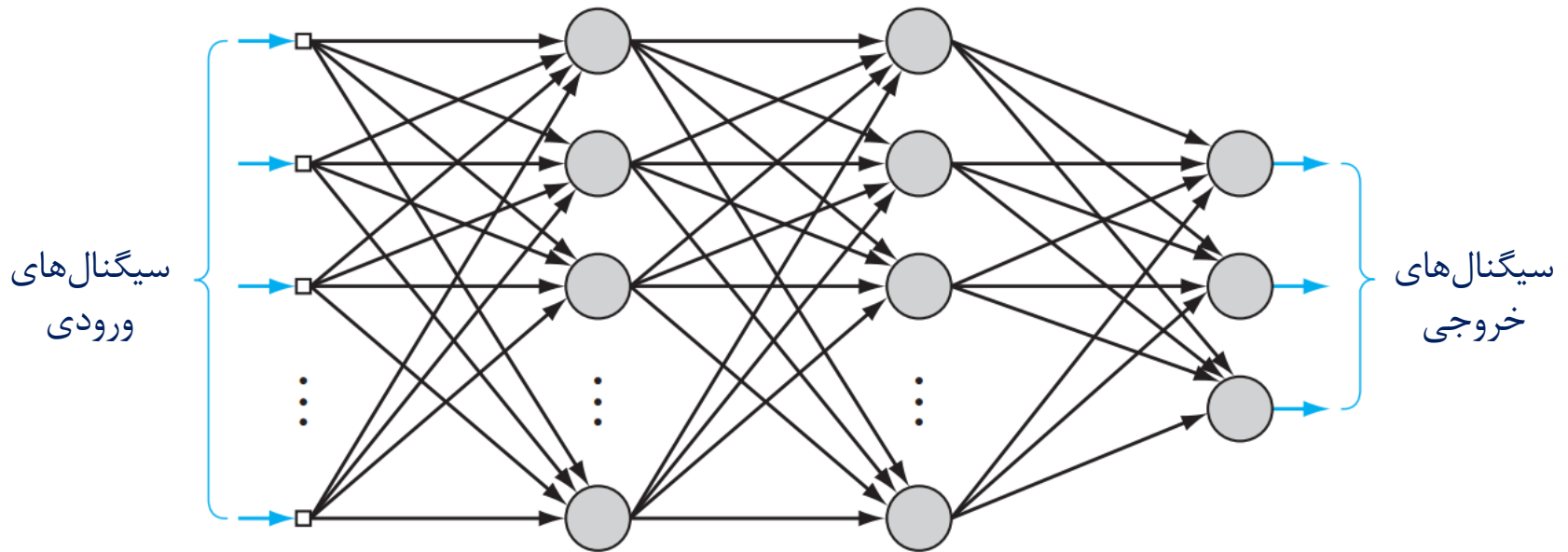
پرسپترون چندلایه (MLP)



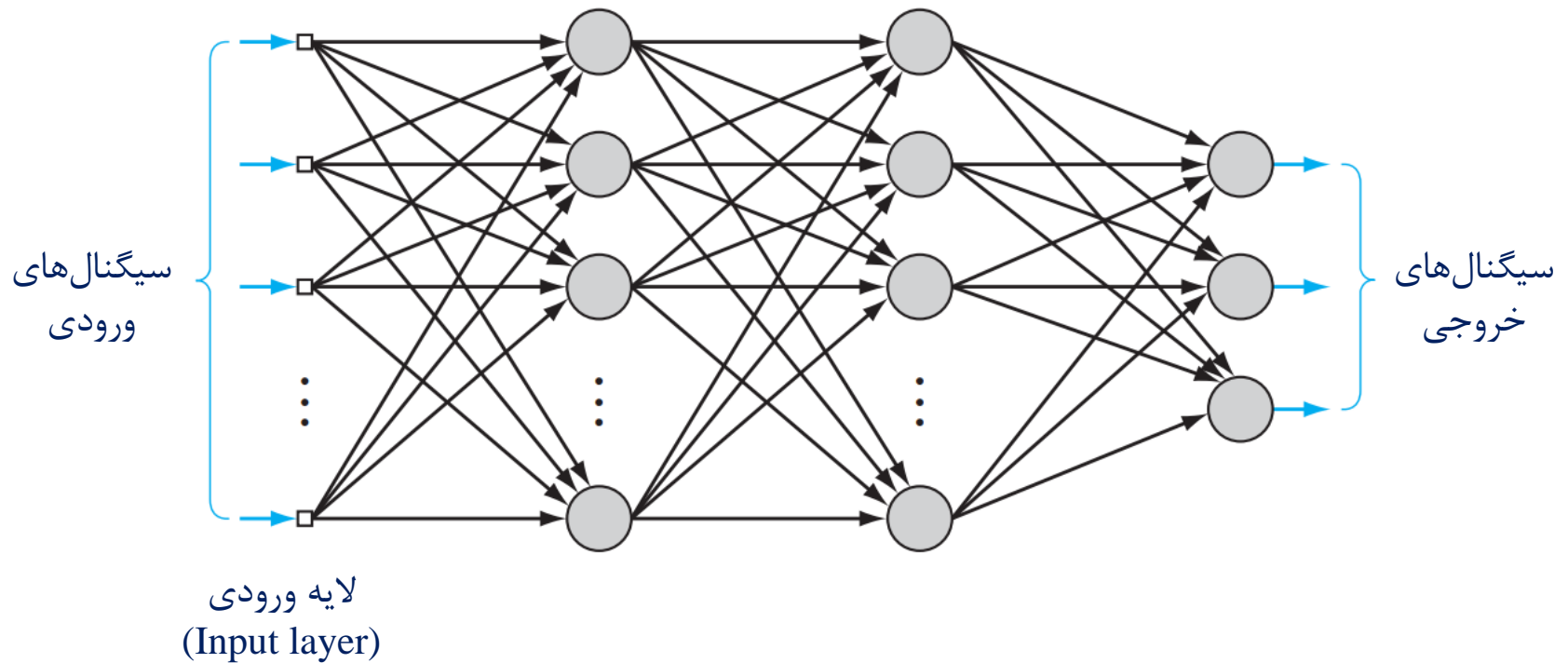
پرسپترون چندلایه (MLP)



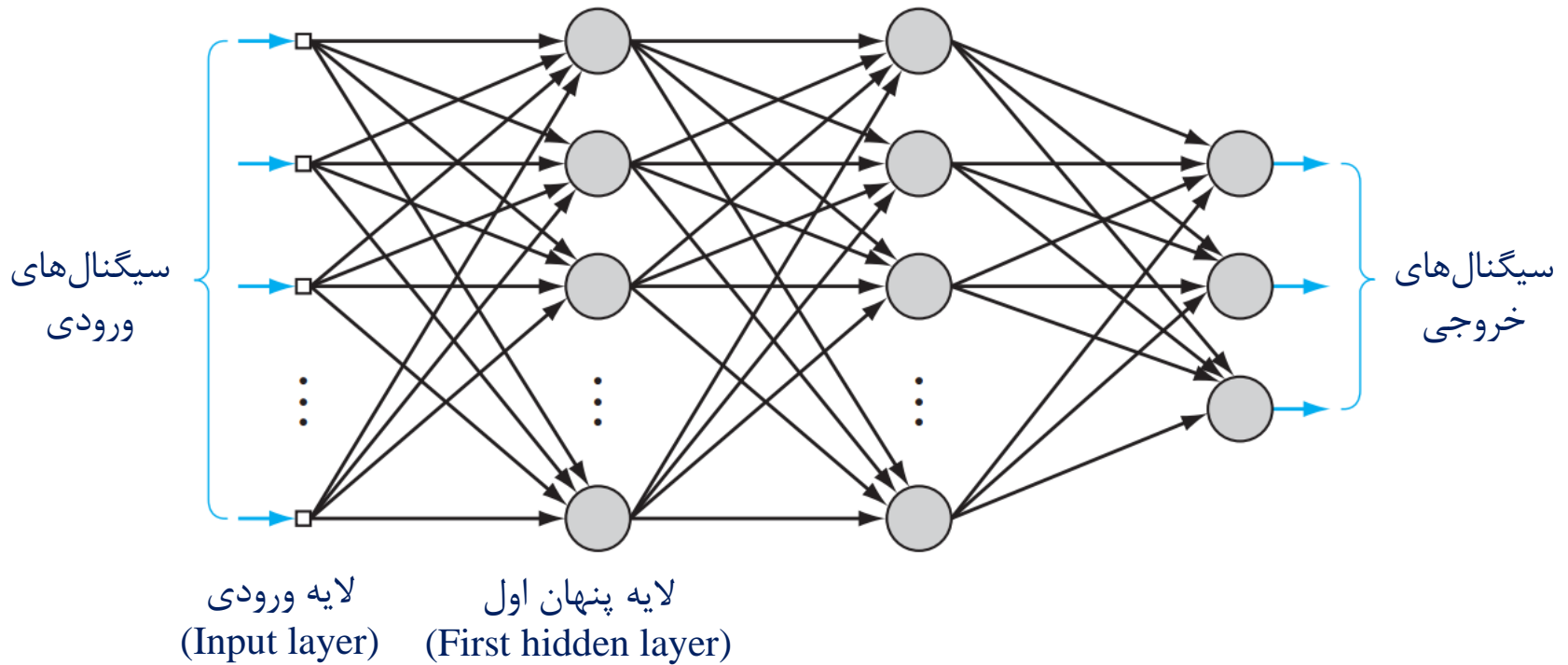
پرسپترون چندلایه (MLP)



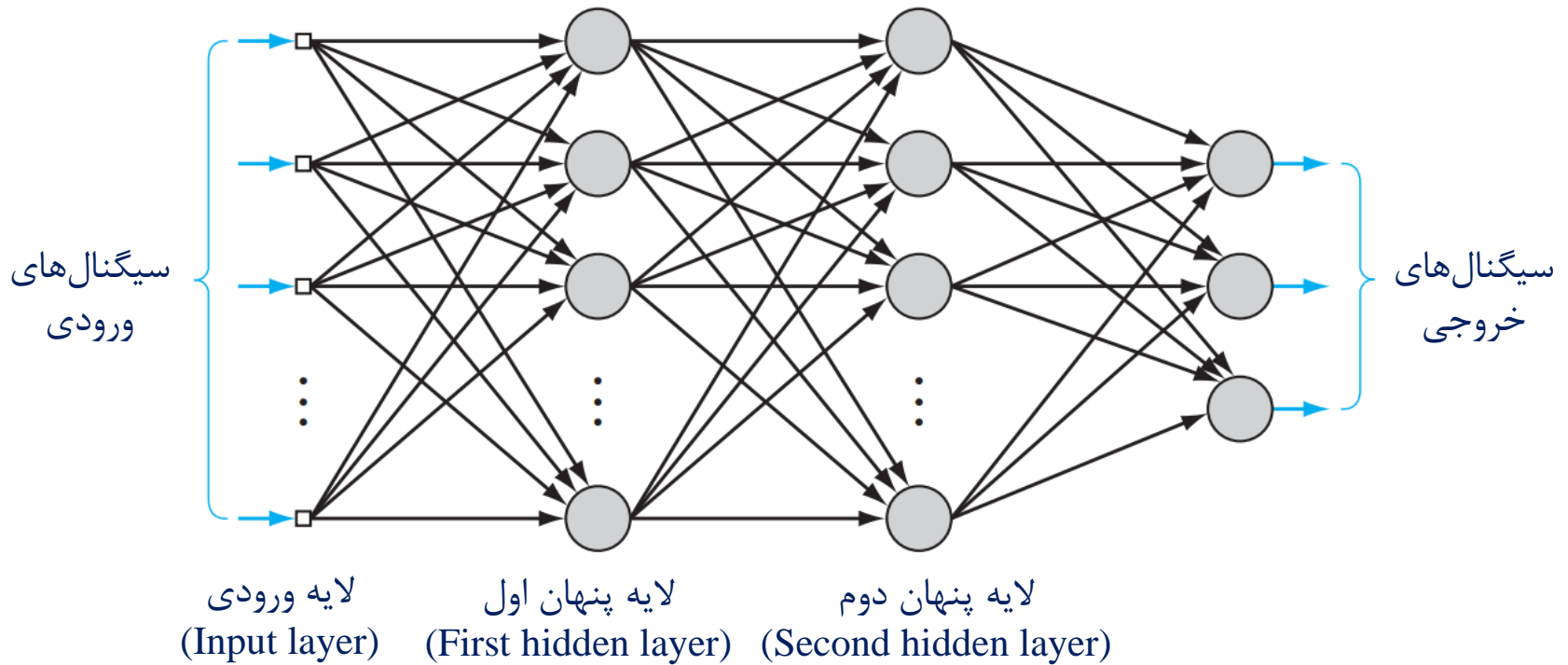
پرسپترون چندلایه (MLP)



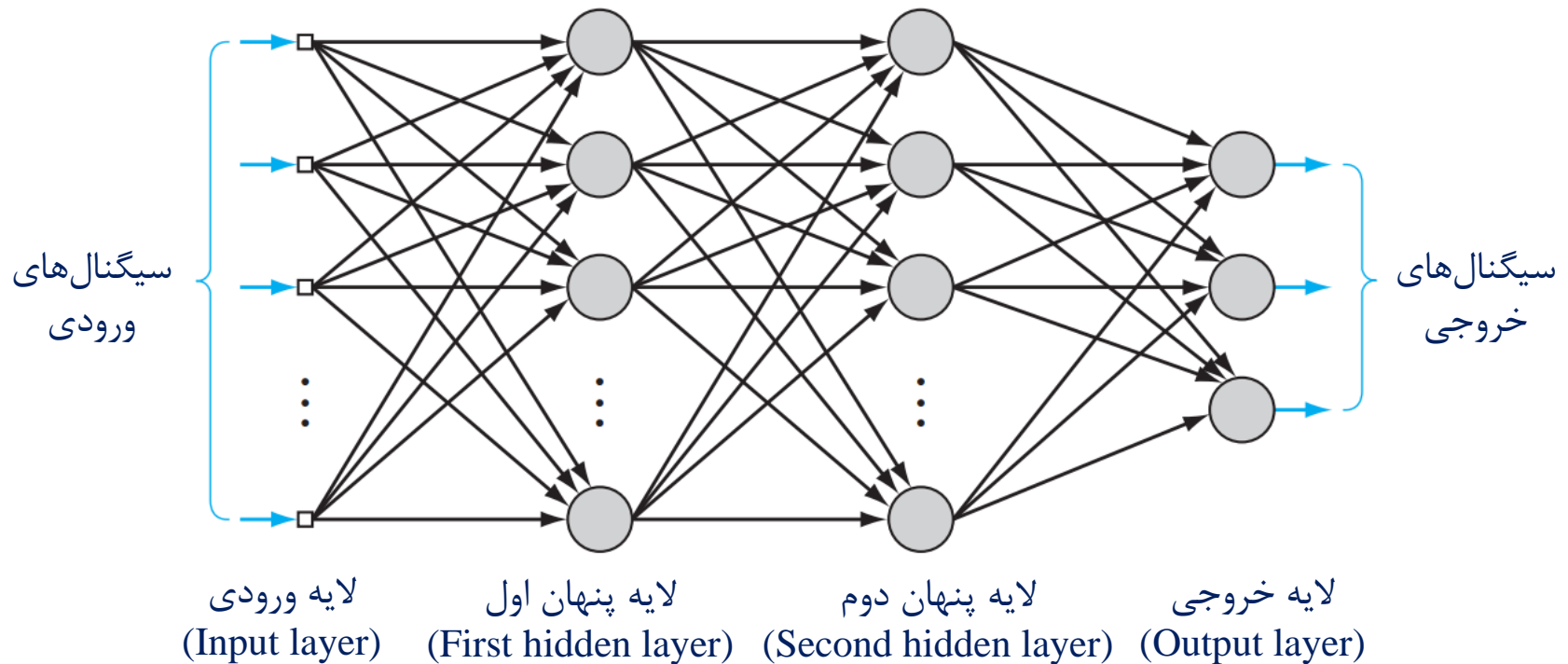
پرسپترون چندلایه (MLP)



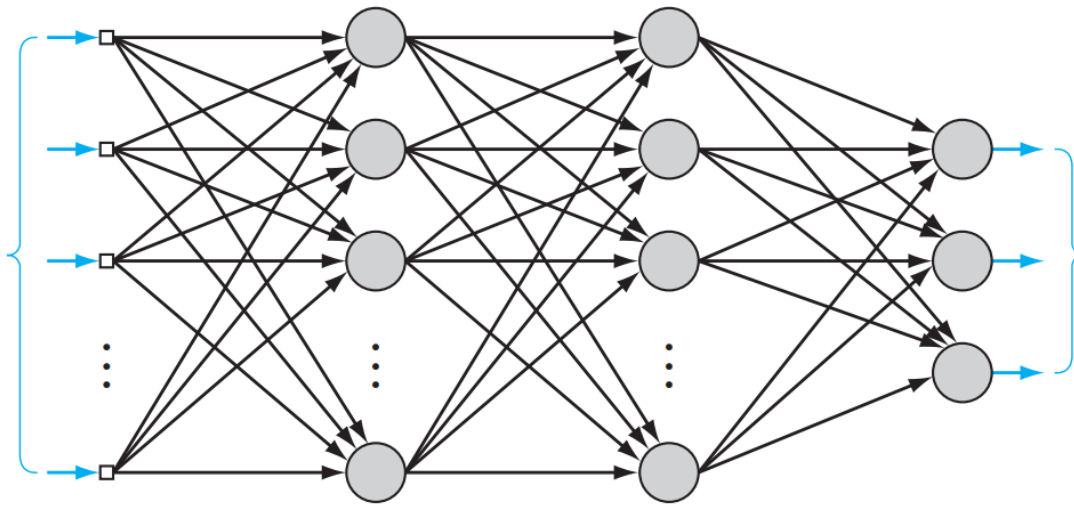
پرسپترون چندلایه (MLP)



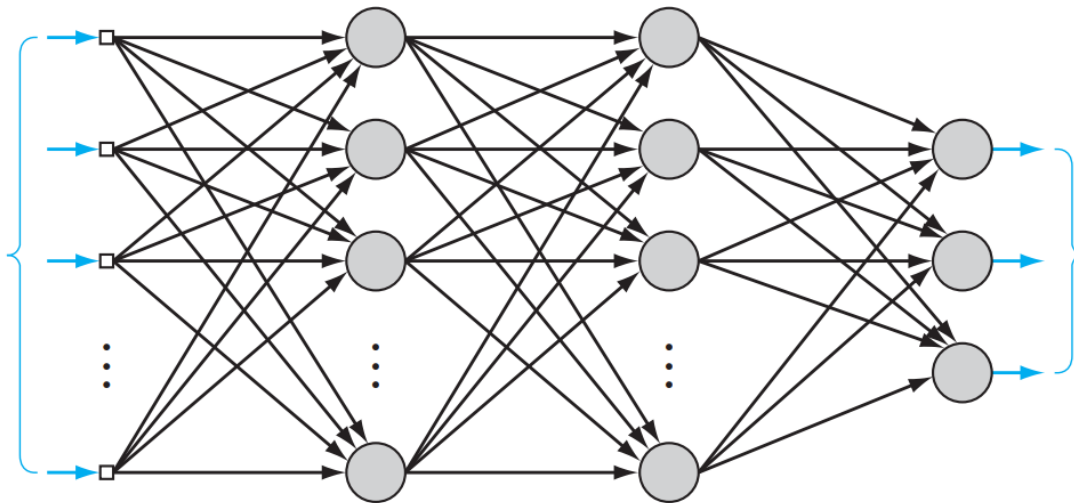
پرسپترون چندلایه (MLP)



پرسترون چندلایه (MLP)

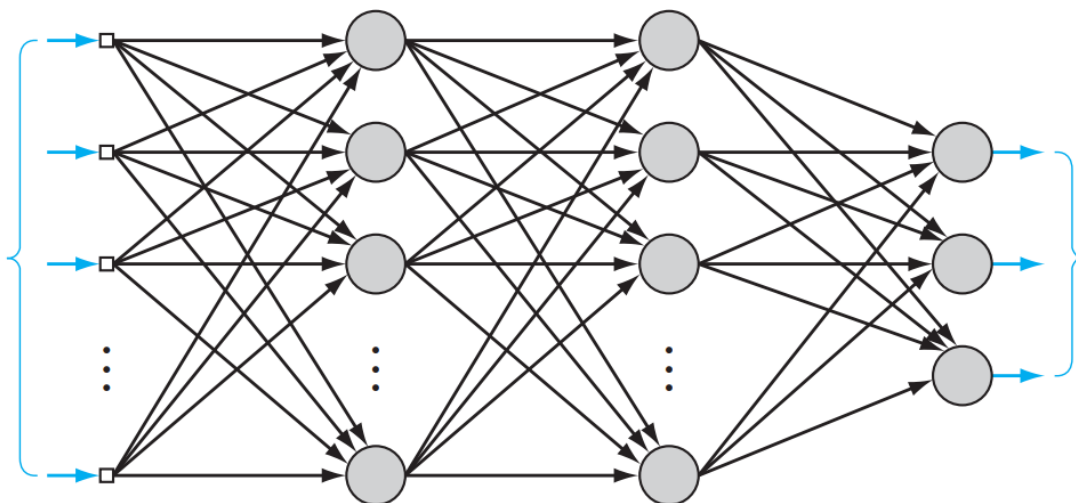


پرسپترون چندلایه (MLP)



- این الگوریتم که در واقع فرم
تعمیم یافته الگوریتم
«کمترین میانگین مربعات»
(LMS) است، از دو فاز
متمايز از هم تشکیل می شود:

پرسترون چندلایه (MLP)

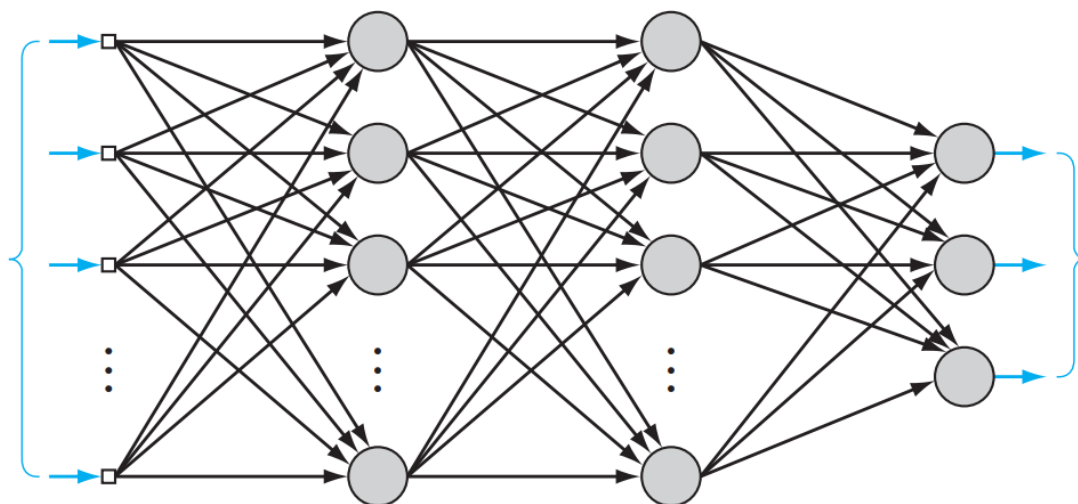


- این الگوریتم که در واقع فرم
تعمیم یافته الگوریتم
«کمترین میانگین مربعات»
(LMS) است، از دو فاز
متمايز از هم تشکیل می شود:

۱- فاز پیش گذر (Forward pass):

که در آن بردار سیگنال ها به لایه ورودی اعمال شده و اثر آن در شبکه، لایه به لایه محاسبه شده تا در نهایت پاسخ شبکه به دست آید. در این مرحله، وزن های شبکه ثابت است.

پرسترون چندلایه (MLP)



- این الگوریتم که در واقع فرم
تعمیم یافته الگوریتم
«کمترین میانگین مربعات»
(LMS) است، از دو فاز
متمايز از هم تشکیل می شود:

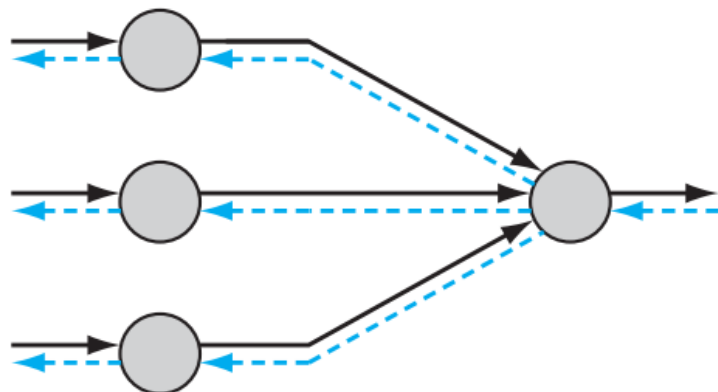
۱- فاز پیش گذر (Forward pass):

که در آن بردار سیگنال ها به لایه ورودی اعمال شده و اثر آن در شبکه، لایه به لایه محاسبه شده تا در نهایت پاسخ شبکه به دست آید. در این مرحله، وزن های شبکه ثابت است.

۲- فاز پس گذر (Backward pass):

که در آن تفاضل بین پاسخ شبکه و پاسخ دلخواه به عنوان خطای شبکه محاسبه شده و این سیگنال خطا از خروجی به ورودی، به صورت لایه به لایه انتشار می یابد. در این حالت وزن های شبکه تنظیم شده تا پاسخ شبکه به پاسخ دلخواه نزدیک شود.

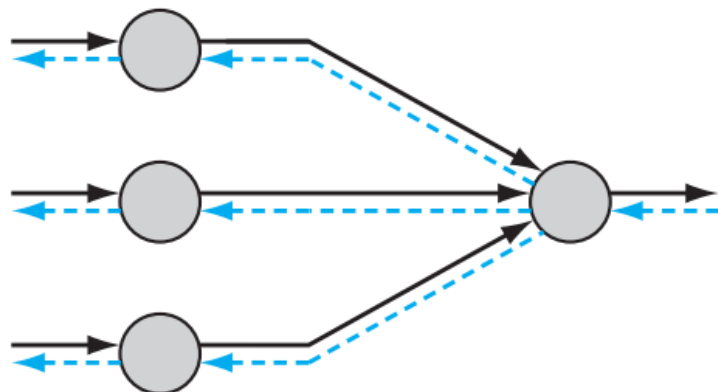
پرسپترون چندلایه (MLP)



—→ Function signals
←--- Error signals

– بنابراین، خطا از سمت خروجی به سمت ورودی در شبکه انتشار یافته و به همین دلیل نام «پس انتشار خطا» به آن نهاده شده است.

پرسپترون چندلایه (MLP)



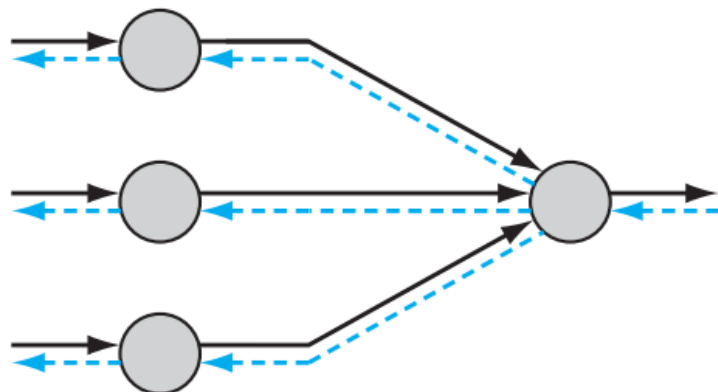
—→ Function signals

←-- Error signals

– بنابراین، خطا از سمت خروجی به سمت ورودی در شبکه انتشار یافته و به همین دلیل نام «پس انتشار خطا» به آن نهاده شده است.

– سلول‌ها در لایه پنهان (سلول‌های پنهان) به عنوان آشکارسازی ویژگی‌های عمل می‌کنند. در واقع نقش این سلول‌ها در عملکرد MLP بسیار حیاتی است.

پرسپترون چندلایه (MLP)



—→ Function signals

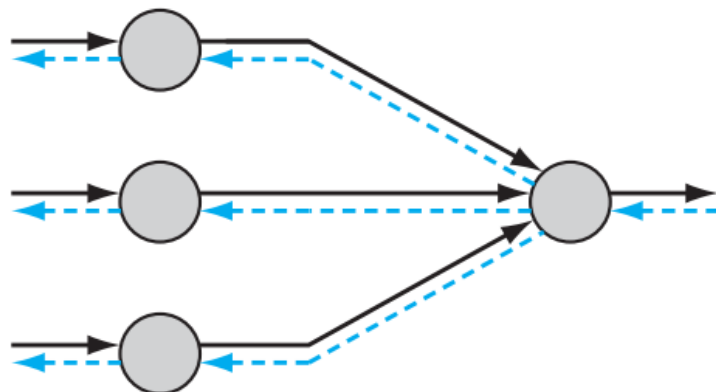
←--- Error signals

– بنابراین، خطا از سمت خروجی به سمت ورودی در شبکه انتشار یافته و به همین دلیل نام «پس انتشار خطا» به آن نهاده شده است.

– سلول‌ها در لایه پنهان (سلول‌های پنهان) به عنوان آشکارسازی ویژگی‌های عمل می‌کنند. در واقع نقش این سلول‌ها در عملکرد MLP بسیار حیاتی است.

– این کار توسط تبدیل غیرخطی داده‌های ورودی به فضایی جدیدی به نام «فضای ویژگی» صورت می‌گیرد.

پرسپترون چندلایه (MLP)



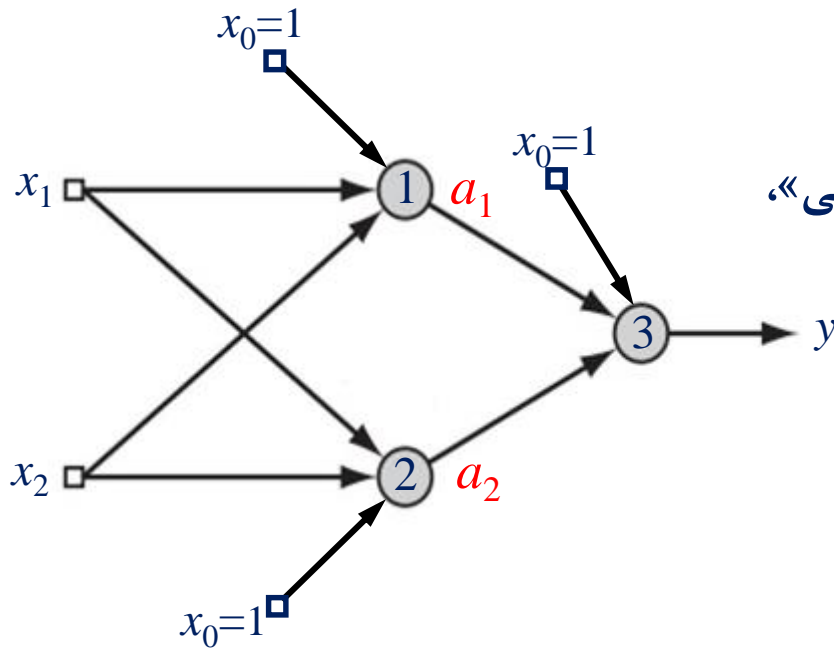
– بنابراین، خطا از سمت خروجی به سمت ورودی در شبکه انتشار یافته و به همین دلیل نام «پس انتشار خطا» به آن نهاده شده است.

– سلول‌ها در لایه پنهان (سلول‌های پنهان) به عنوان آشکارسازی ویژگی‌های عمل می‌کنند. در واقع نقش این سلول‌ها در عملکرد MLP بسیار حیاتی است.

– این کار توسط تبدیل غیرخطی داده‌های ورودی به فضایی جدیدی به نام «فضای ویژگی» صورت می‌گیرد.

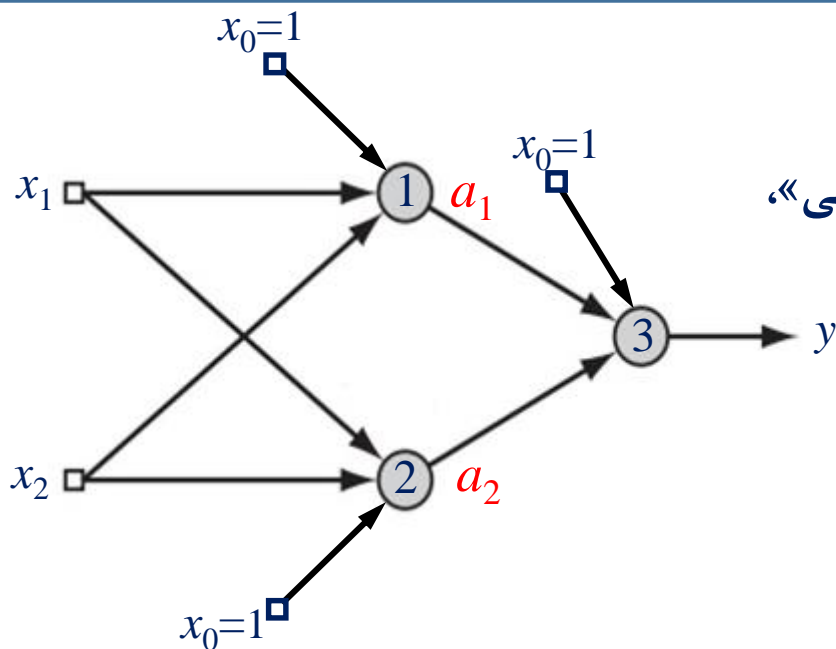
– به واقع، شکل‌گیری این فضا است که عملکرد MLP را از پرسپترون روزنبلات متمایز می‌کند.

پرسپترون چندلایه (MLP)



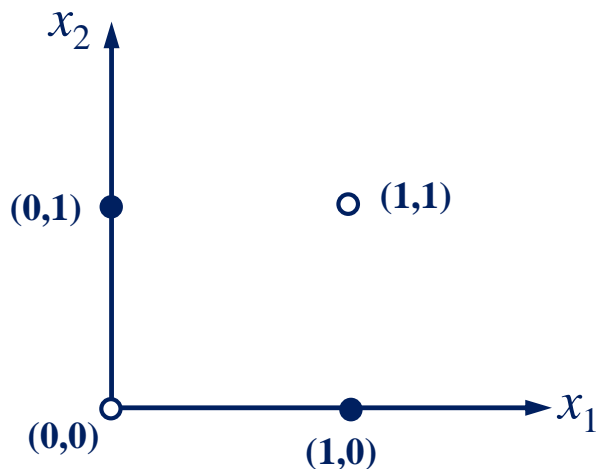
- به منظور بهتر متوجه شدن اهمیت «فضای ویژگی»،
مثال XOR را به خاطر بیاورید.

پرسپترون چندلایه (MLP)

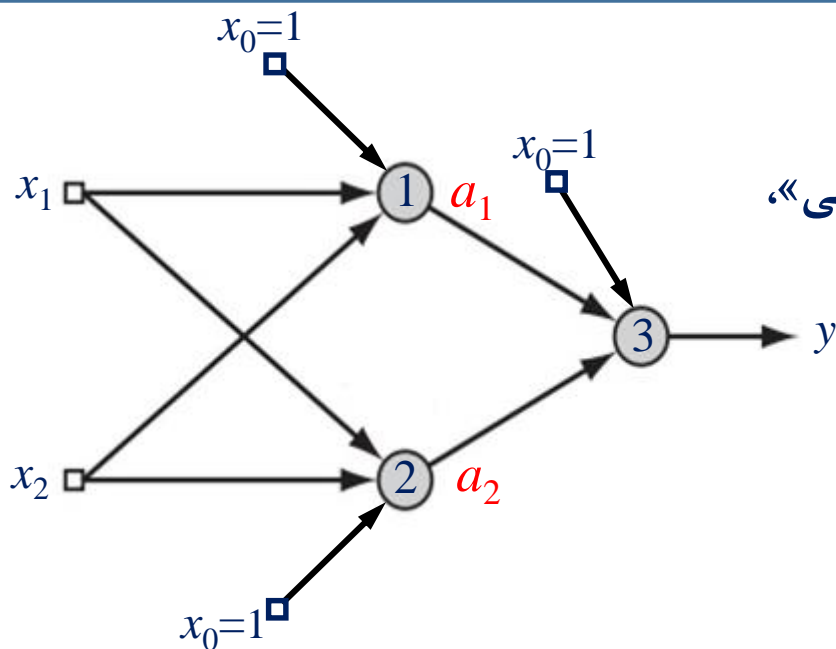


- به منظور بهتر متوجه شدن اهمیت «فضای ویژگی»، مثال XOR را به خاطر بیاورید.

- داده‌ها در فضای ورودی به طور خطی جداپذیر نبودند. ولی پس از تبدیل غیرخطی در لایه پنهان، در فضای ویژگی (فضای پنهان) به طور خطی جداپذیر شدند.

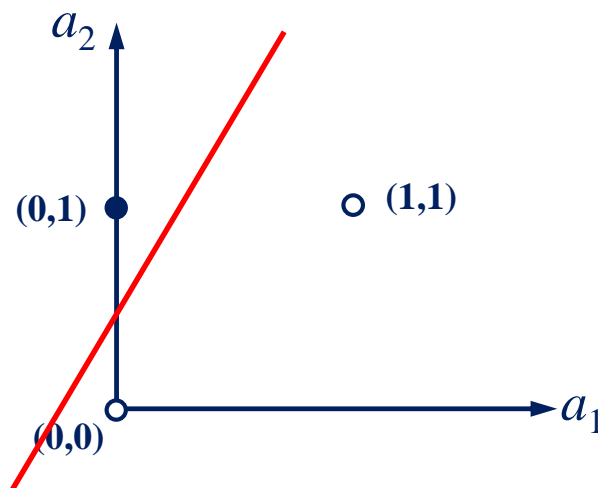
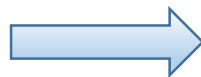
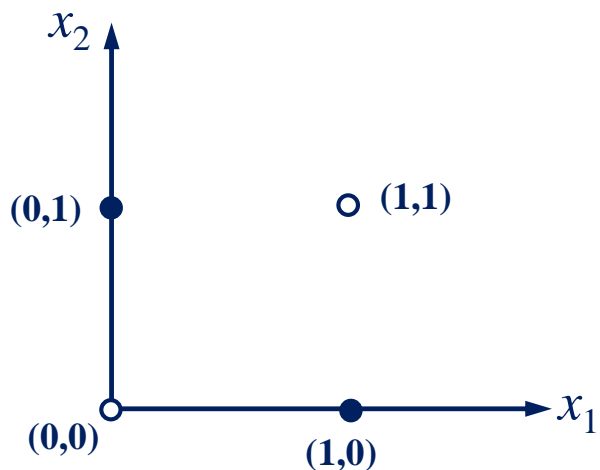


پرسپترون چندلایه (MLP)



- به منظور بهتر متوجه شدن اهمیت «فضای ویژگی»، مثال XOR را به خاطر بیاورید.

- داده‌ها در فضای ورودی به طور خطی جداپذیر نبودند. ولی پس از تبدیل غیرخطی در لایه پنهان، در فضای ویژگی (فضای پنهان) به طور خطی جداپذیر شدند.



پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.

پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

مثال‌هایی از توابع S شکل:

پرسپترون چندلایه (MLP)

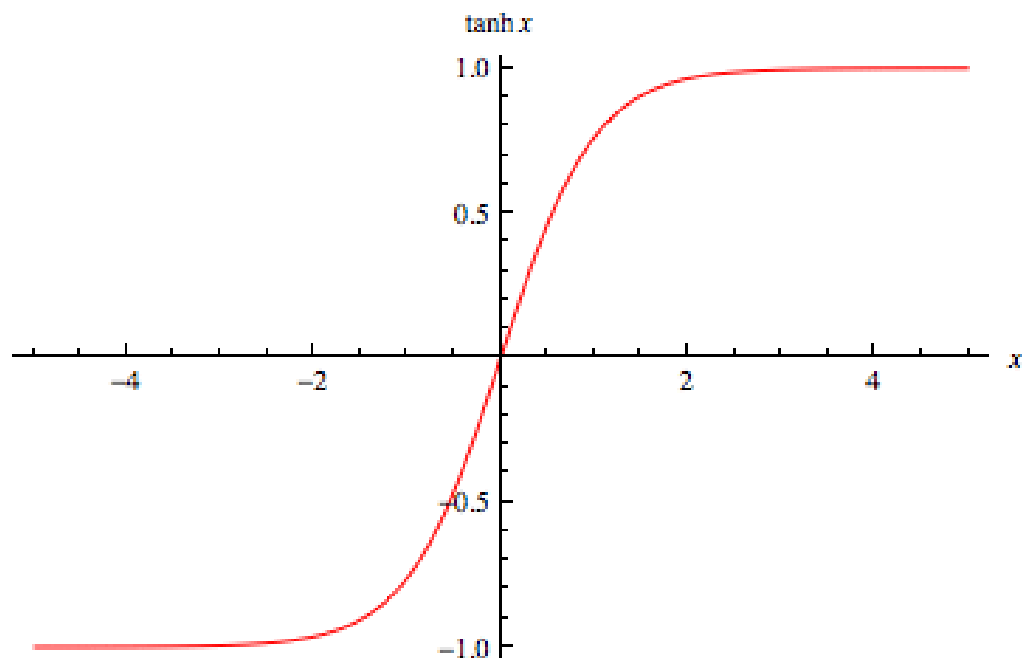
سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

مثال‌هایی از توابع S شکل:

۱- تابع تانژانت هیپربولیک

$$y_j = \frac{1 - e^{-av_j}}{1 + e^{-av_j}}$$

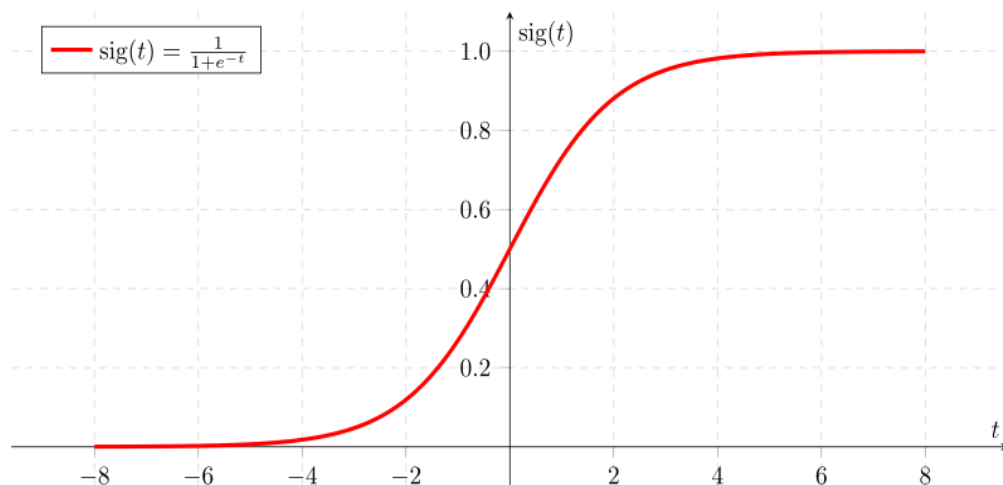


پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

مثال‌هایی از توابع S شکل:



۲- تابع لجستیکی
(Logistic function)

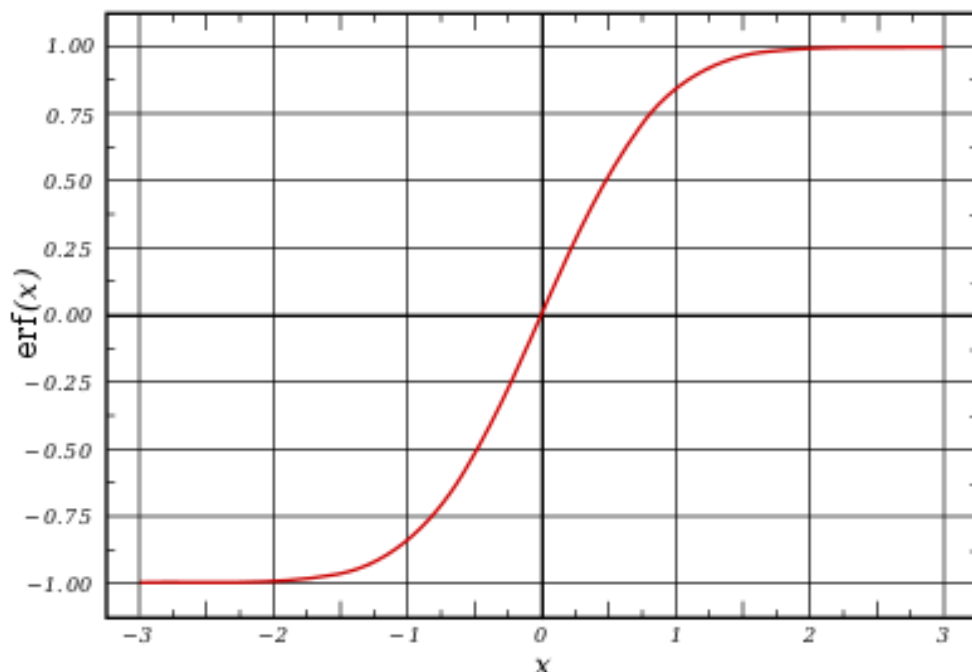
$$y_j = \frac{1}{1 + e^{-av_j}}$$

پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

مثال‌هایی از توابع S شکل:



۳- تابع خطا
(Error function)

$$y_j = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{v_j} e^{-\frac{t^2}{2}} dt$$

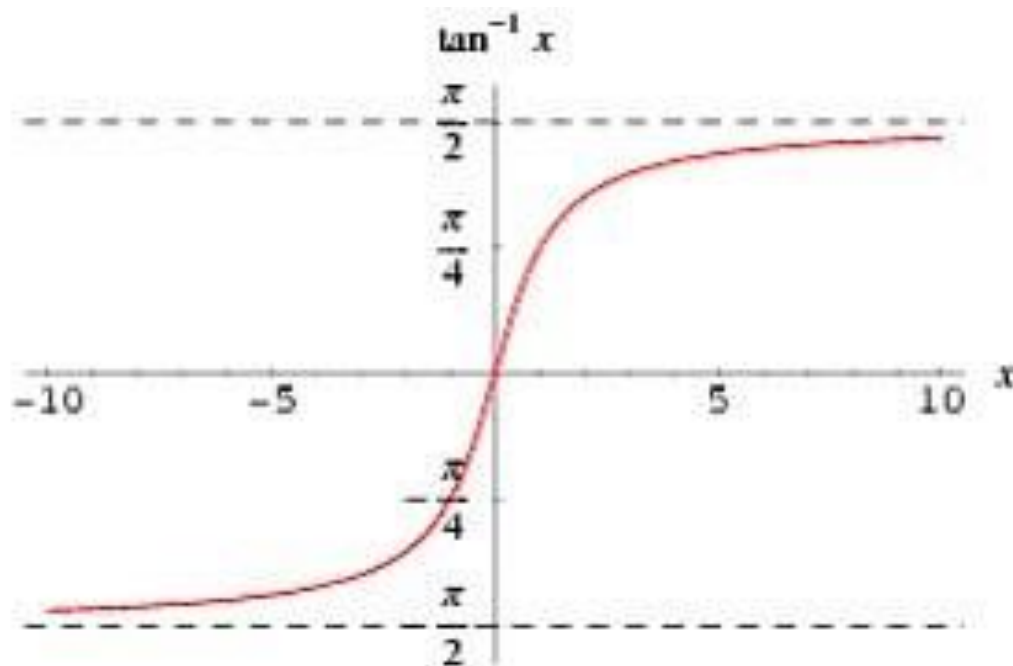
پرسپترون چندلایه (MLP)

سلول‌ها در MLP:

- هر سلول MLP دارای تابع غیرخطی در خروجی‌اش است که به آن تابع فعال‌سازی (Activation function) می‌گویند.
- این توابع از نوع S شکل (Sigmoid) است.

مثال‌هایی از توابع S شکل:

۴- تابع وارون تانژانت

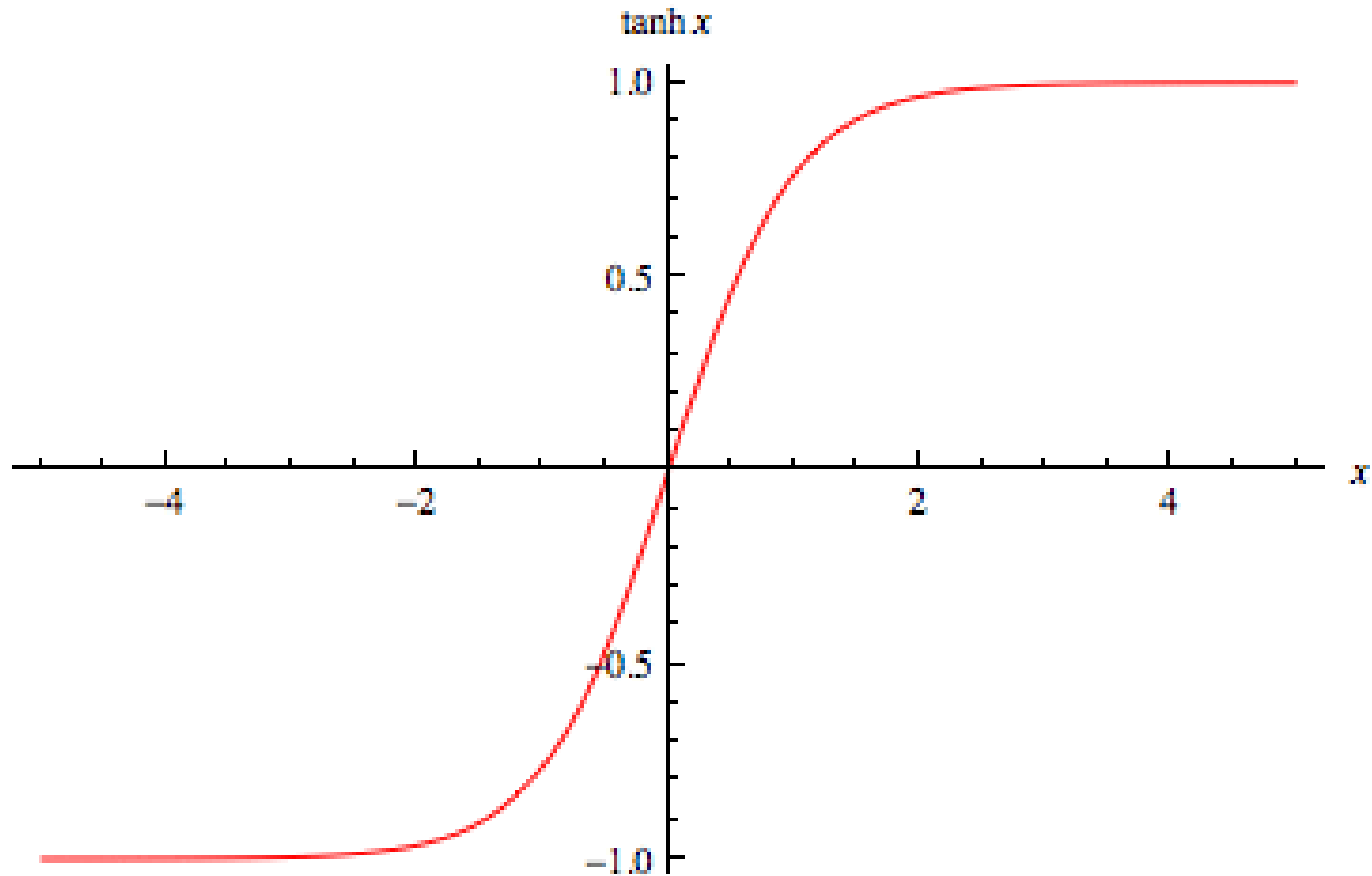


$$y_j = \frac{2}{\pi} \tan^{-1}(v_j)$$

پرسپترون چندلایه (MLP)

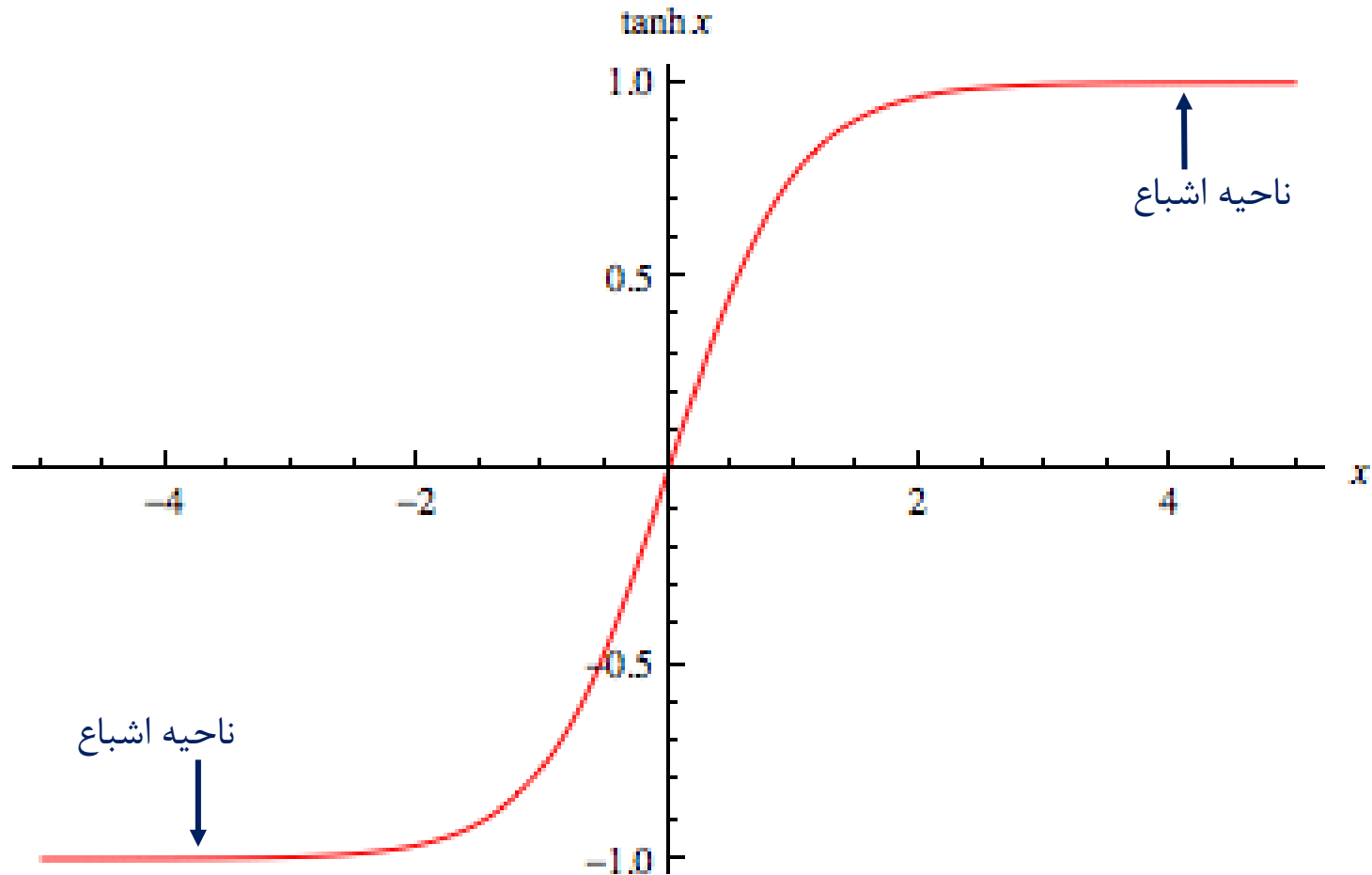
پرسپترون چندلایه (MLP)

نواحی مختلف توابع S شکل



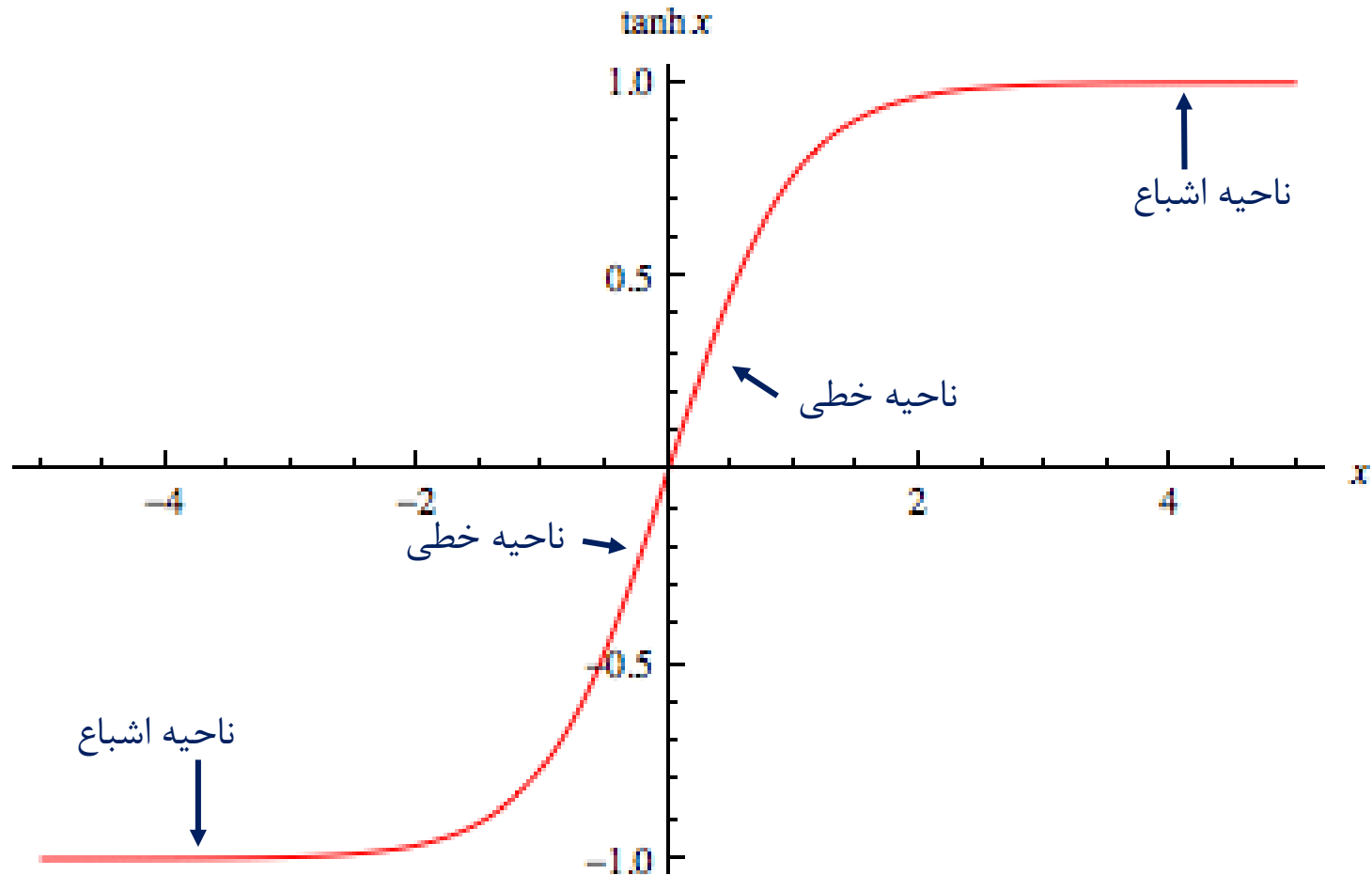
پرسپترون چندلایه (MLP)

نواحی مختلف توابع S شکل



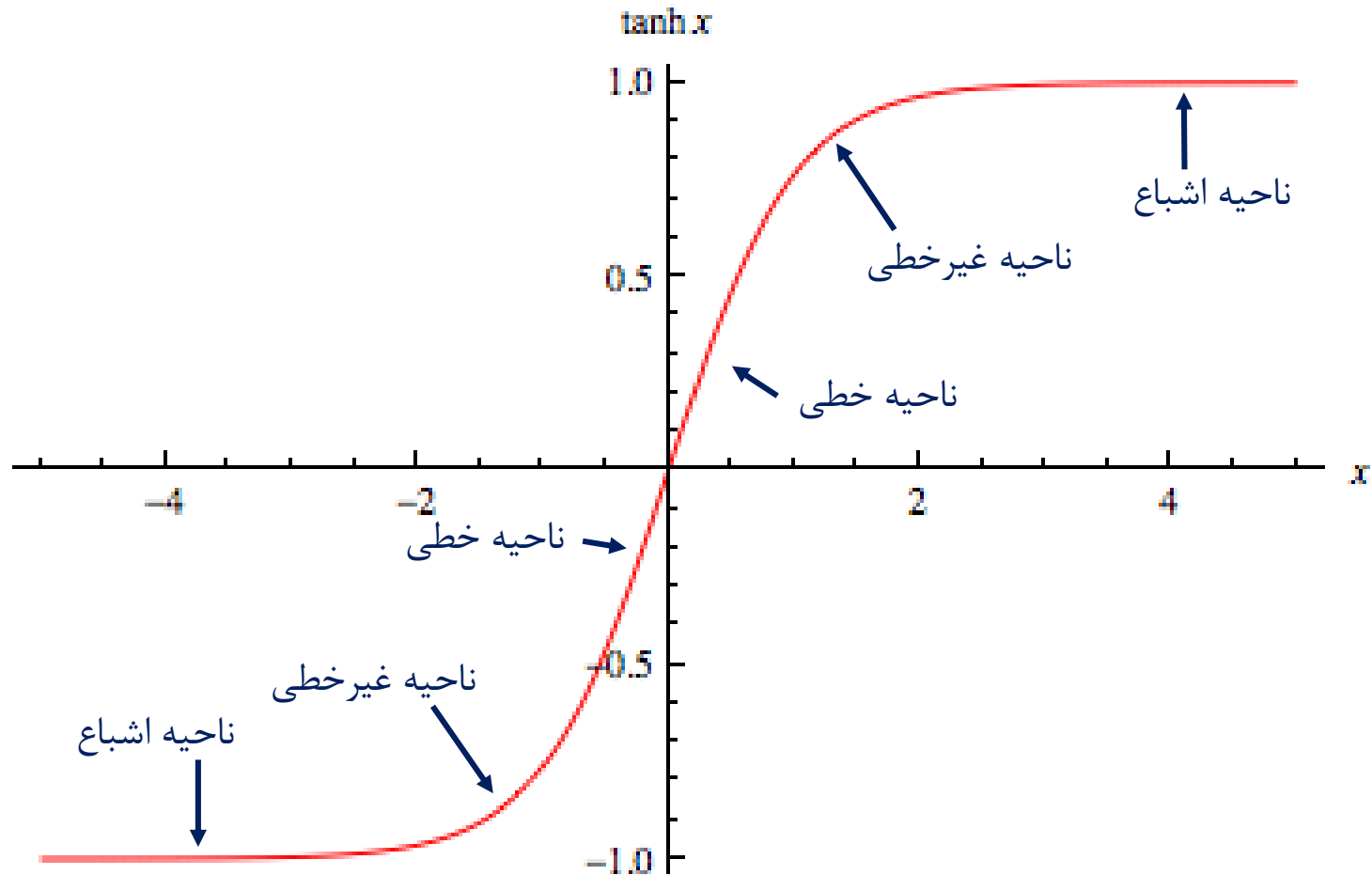
پرسپترون چندلایه (MLP)

نواحی مختلف توابع S شکل



پرسپترون چندلایه (MLP)

نواحی مختلف توابع S شکل



پرسپترون چندلایه (MLP)

پرسپترون چندلایه (MLP)

تمرین: چنانچه از توابع خطی به جای توابع غیرخطی در MLP استفاده شود، در این صورت چه نتیجه‌ای می‌توانید بگیرید؟

پرسپترون چندلایه (MLP)

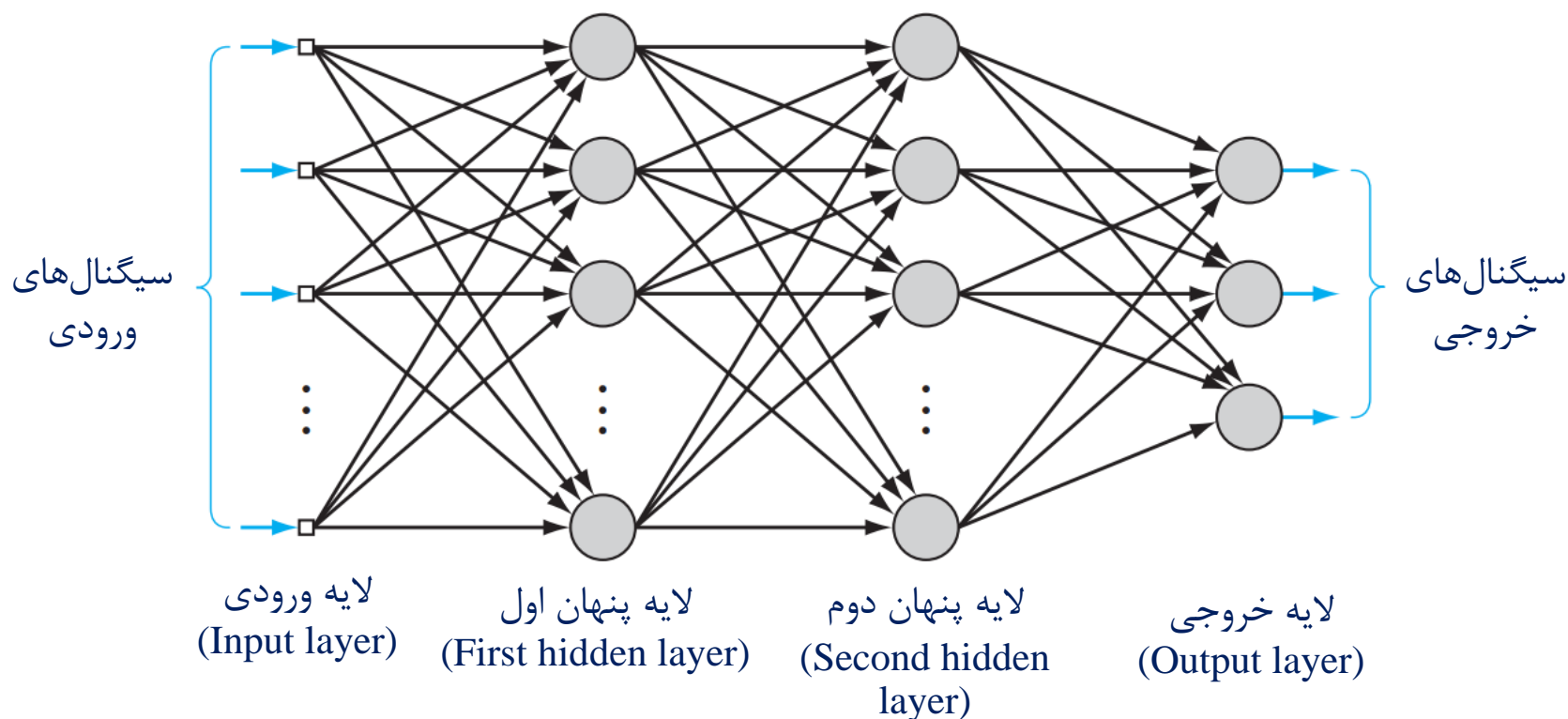
پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

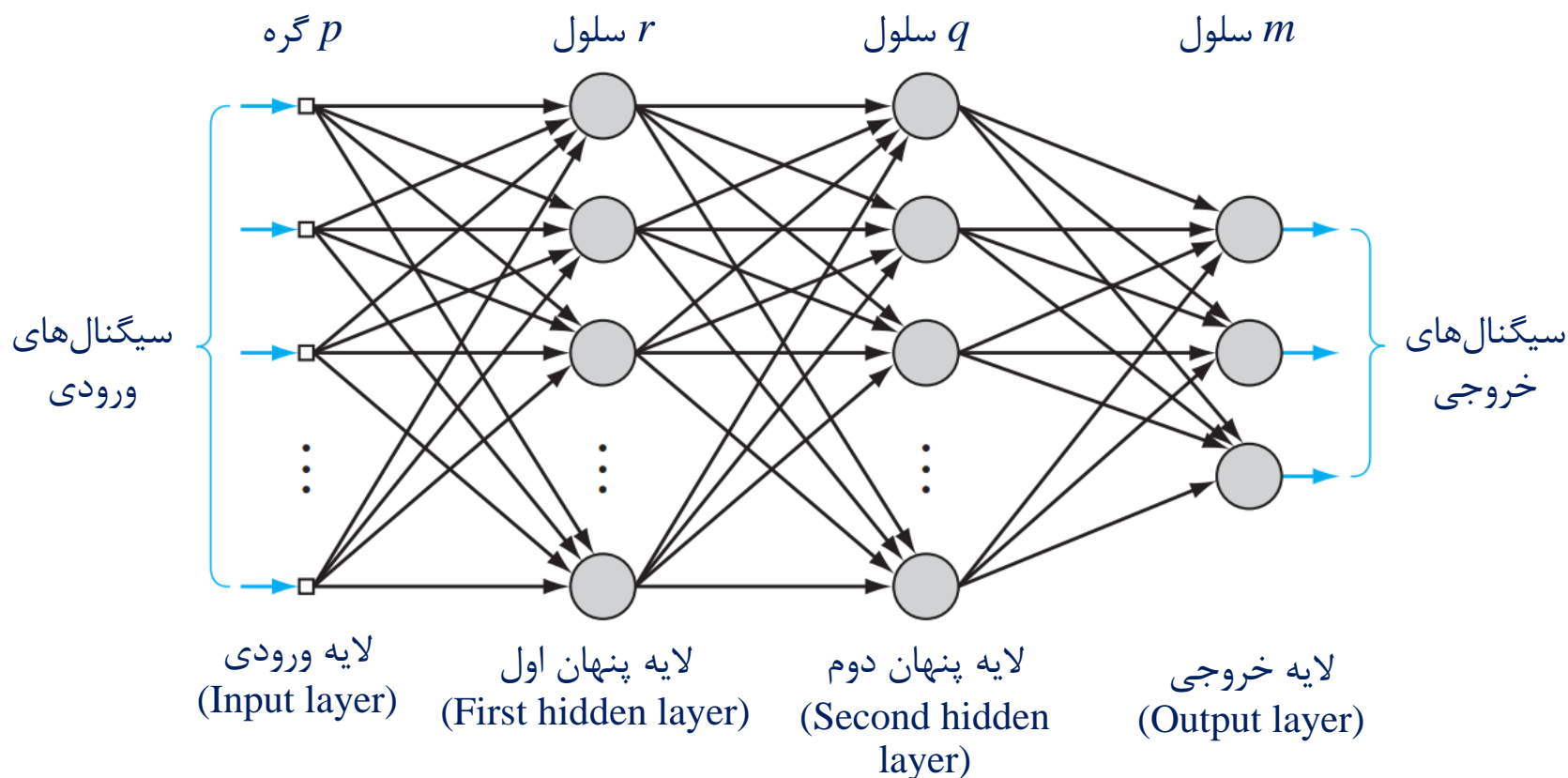
– این الگوریتم را برای شبکه زیر با دو لایه پنهان اجرا می کنیم. اگرچه برای هر تعداد لایه پنهان توسعه پذیر است.



پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

– این الگوریتم را برای شبکه زیر با دو لایه پنهان اجرا می کنیم. اگرچه برای هر تعداد لایه پنهان توسعه پذیر است.



پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس های i ، j و k برای سلول های مختلف در شبکه
سلول j در سمت راست سلول i و سلول k در سمت راست سلول j .

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس های i ، j و k برای سلول های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس های i ، j و k برای سلول های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس های i ، j و k برای سلول های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n
- $w_{j0}(n)$ مقدار آستانه سلول j در مرحله n (متصل به ورودی ثابت $y_0 = +1$)

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه‌ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس‌های i ، j و k برای سلول‌های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n
- $w_{j0}(n)$ مقدار آستانه سلول j در مرحله n (متصل به ورودی ثابت $y_0 = +1$)
- $v_j(n)$ جمع خطی سیگنال‌های ورودی به سلول j در مرحله n

پرسترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه‌ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس‌های i ، j و k برای سلول‌های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n
- $w_{j0}(n)$ مقدار آستانه سلول j در مرحله n (متصل به ورودی ثابت $y_0 = +1$)
- $v_j(n)$ جمع خطی سیگنال‌های ورودی به سلول j در مرحله n
- $\varphi_j(v_j(n))$ تابع فعال‌سازی (تابع غیرخطی) سلول j در مرحله n

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه‌ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس‌های i ، j و k برای سلول‌های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n
- $w_{j0}(n)$ مقدار آستانه سلول j در مرحله n (متصل به ورودی ثابت $y_0 = +1$)
- $v_j(n)$ جمع خطی سیگنال‌های ورودی به سلول j در مرحله n
- $\varphi_j(v_j(n))$ تابع فعال‌سازی (تابع غیرخطی) سلول j در مرحله n
- $x_i(n)$ ورودی i ام به شبکه در مرحله n

پرسپترون چندلایه (MLP)

الگوریتم پس انتشار خطا:

ابتدا، نمادهای زیر را تعریف می کنیم:

- n مرحله تکرار n ام است (اعمال الگوی آموزش n ام به شبکه)
- $\mathcal{E}(n)$ جمع لحظه‌ای مربعات خطا و \mathcal{E}_{av} میانگین مقادیر $\mathcal{E}(n)$ برای تمام مراحل n
- اندیس‌های i ، j و k برای سلول‌های مختلف در شبکه
- سلول j در سمت راست سلول i و سلول k در سمت راست سلول j
- $e_k(n)$ ، $d_k(n)$ و $y_k(n)$ به ترتیب خطای سیگنال خروجی، خروجی دلخواه و خروجی واقعی سلول k ام در لایه خروجی
- $w_{ji}(n)$ وزن اتصالی بین سلول i و سلول j در مرحله n
- $w_{j0}(n)$ مقدار آستانه سلول j در مرحله n (متصل به ورودی ثابت $y_0 = +1$)
- $v_j(n)$ جمع خطی سیگنال‌های ورودی به سلول j در مرحله n
- $\varphi_j(v_j(n))$ تابع فعال‌سازی (تابع غیرخطی) سلول j در مرحله n
- $x_i(n)$ ورودی i ام به شبکه در مرحله n
- η ضریب آموزش