



شبکه‌های عصبی مصنوعی

جلسه بیست و دوم:
شبکه کوهونن (۱)
(Kohonen Network)

سیستم‌های خودسازمانده (Self-Organizing Systems)

سیستم‌های خودسازمانده (Self-Organizing Systems)

– در جلسه قبل، نوعی از سیستم‌های خودسازمانده را بررسی کردیم که هدف آن استخراج ویژگی‌های بردار ورودی است (شبکه هب).

سیستم‌های خودسازمانده (Self-Organizing Systems)

– در جلسه قبل، نوعی از سیستم‌های خودسازمانده را بررسی کردیم که هدف آن استخراج ویژگی‌های بردار ورودی است (شبکه هب).

– در این جلسه نوع دیگری از سیستم‌های خودسازمانده را بررسی می‌کنیم که مبنای عملکرد آن براساس رقابت بین سلول‌هاست به‌طوری که در انتهای رقابت، یک سلول یا دسته‌ای از سلول‌ها برنده می‌شوند.

سیستم‌های خودسازمانده (Self-Organizing Systems)

– در جلسه قبل، نوعی از سیستم‌های خودسازمانده را بررسی کردیم که هدف آن استخراج ویژگی‌های بردار ورودی است (شبکه هب).

– در این جلسه نوع دیگری از سیستم‌های خودسازمانده را بررسی می‌کنیم که مبنای عملکرد آن براساس رقابت بین سلول‌هاست به‌طوری که در انتهای رقابت، یک سلول یا دسته‌ای از سلول‌ها برنده می‌شوند.

– این رقابت به‌صورت بدون نظارت (خودسازمانده) انجام می‌شود.

سیستم‌های خودسازمانده (Self-Organizing Systems)

– در جلسه قبل، نوعی از سیستم‌های خودسازمانده را بررسی کردیم که هدف آن استخراج ویژگی‌های بردار ورودی است (شبکه هب).

– در این جلسه نوع دیگری از سیستم‌های خودسازمانده را بررسی می‌کنیم که مبنای عملکرد آن براساس رقابت بین سلول‌هاست به‌طوری که در انتهای رقابت، یک سلول یا دسته‌ای از سلول‌ها برنده می‌شوند.

– این رقابت به‌صورت بدون نظارت (خودسازمانده) انجام می‌شود.

– هدف از این نوع رقابت بین سلول‌ها، نگاشت داده‌های بسیار زیاد و با ابعاد بزرگ از فضای ورودی به فضای خروجی با دو بعد به‌شکل مشبک شده است به‌طوری که کلاس مفهومی داده‌ها به‌فرم توپولوژیکی تعیین شود.

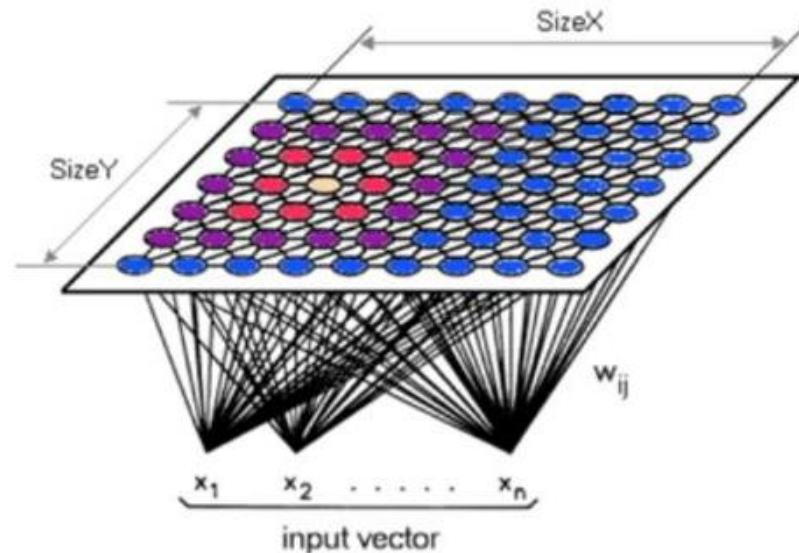
سیستم‌های خودسازمانده (Self-Organizing Systems)

سیستم‌های خودسازمانده (Self-Organizing Systems)



[Teuvo Kohonen](#)

Unsupervised Learning:
Kohonen Self-Organizing Map

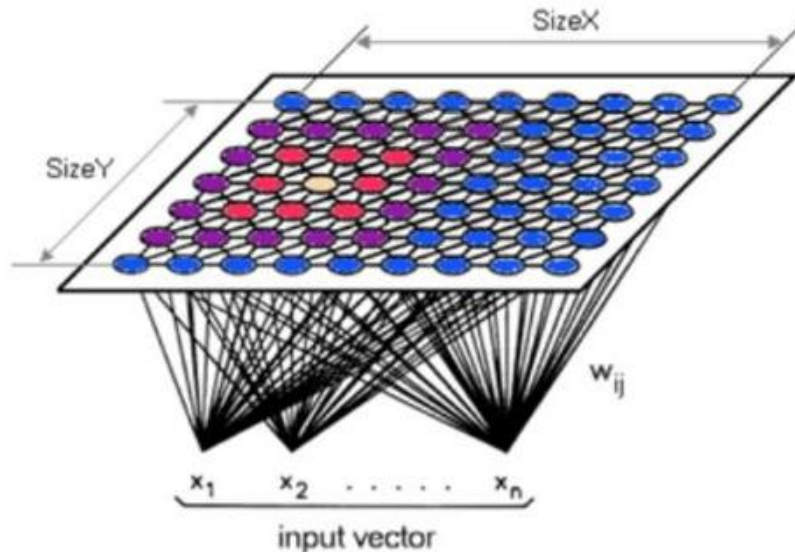


سیستم‌های خودسازمانده (Self-Organizing Systems)



[Teuvo Kohonen](#)

Unsupervised Learning:
Kohonen Self-Organizing Map

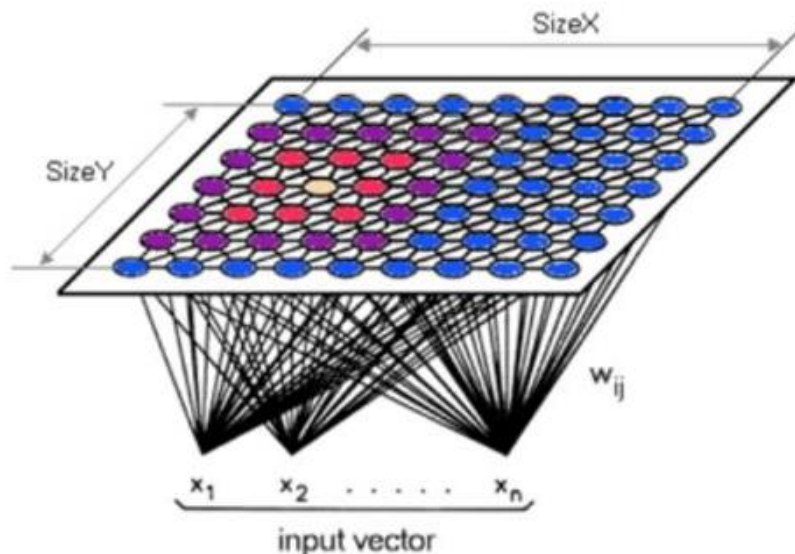


سیستم‌های خودسازمانده (Self-Organizing Systems)

Unsupervised Learning: Kohonen Self-Organizing Map



Teuvo Kohonen



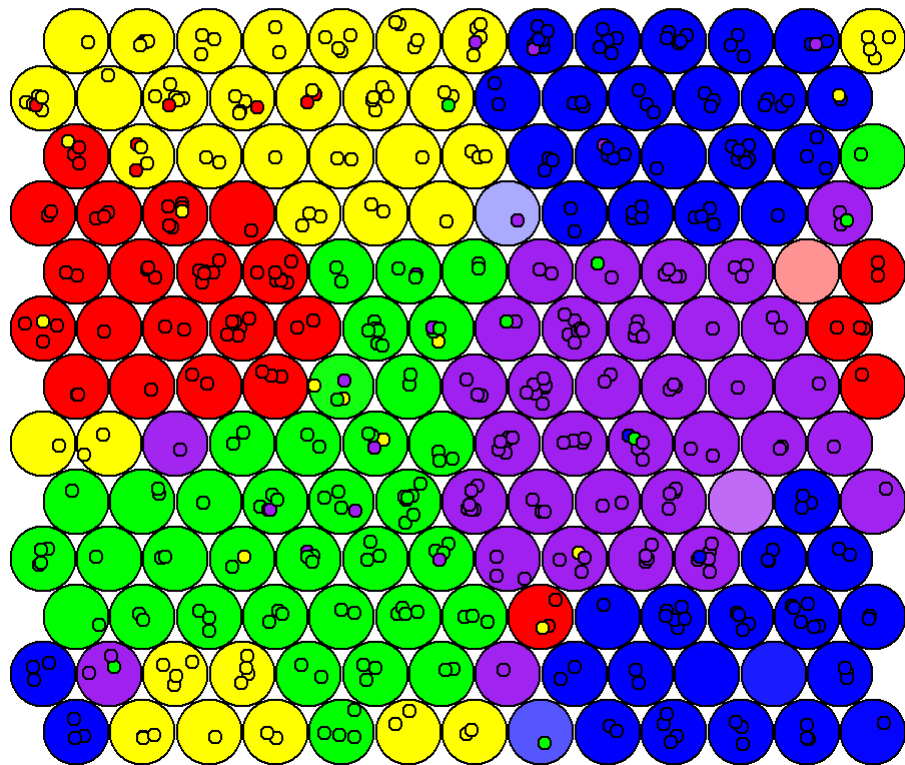
- هدف از این نوع رقابت بین سلول‌ها، نگاشت داده‌های بسیار زیاد و با ابعاد بزرگ از فضای ورودی به فضای خروجی با دو بعد به شکل مشبک شده است به طوری که کلاس مفهومی داده‌ها به فرم توپولوژیکی تعیین شود.

سیستم‌های خودسازمانده (Self-Organizing Systems)

Mapping plot



Teuvo Kohonen



– هدف از این نوع رقابت بین سلول‌ها، نگاشت داده‌های بسیار زیاد و با ابعاد بزرگ از فضای ورودی به فضای خروجی با دو بعد به شکل مشبک شده است به طوری که کلاس مفهومی داده‌ها به فرم توپولوژیکی تعیین شود.

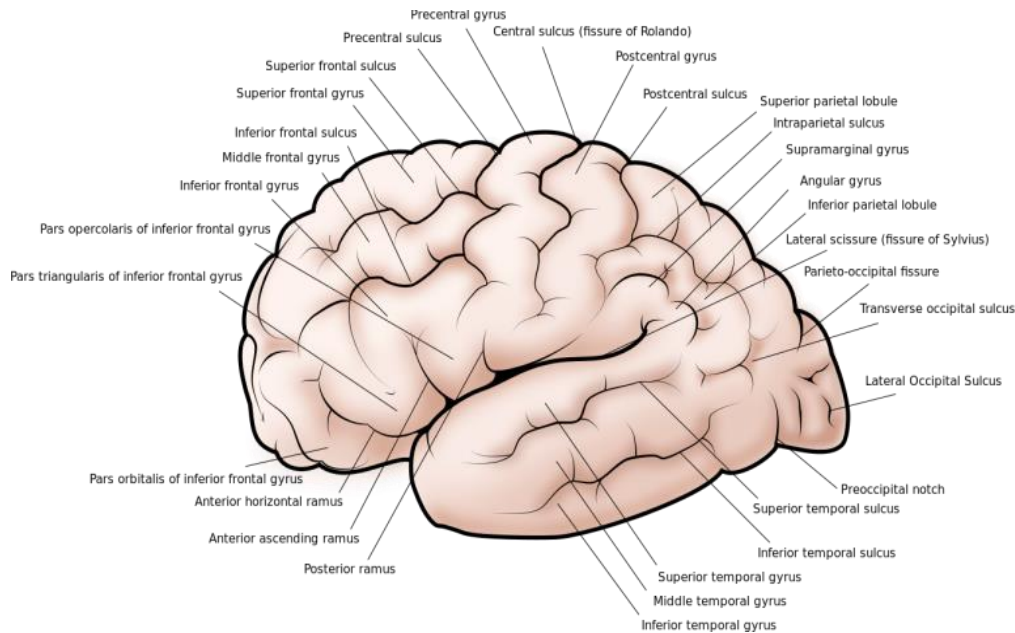
سیستم‌های خودسازمانده (Self-Organizing Systems)

سیستم‌های خودسازمانده (Self-Organizing Systems)

– به این نوع نگاشت داده‌ها، نگاشت خودسازمانده (Self-Organizing Map \equiv SOM) می‌گویند.
یا نگاشت خودسازمانده ویژگی‌ها (Self-Organizing Feature Map \equiv SOFM) می‌گویند.

سیستم‌های خودسازمانده (Self-Organizing Systems)

– به این نوع نگاشت داده‌ها، نگاشت خودسازمانده (Self-Organizing Map \equiv SOM) می‌گویند.
یا نگاشت خودسازمانده ویژگی‌ها (Self-Organizing Feature Map \equiv SOFM)



– ایده این نوع نگاشت از غشای مغزی (Cerebral Cortex) انسان گرفته شده که در آن نگاشت به فضای سه بعدی است به طوری که ورودی از سنسورهای مختلف به مکان‌های متفاوت مغز نگاشت می‌شود (منظم کردن توپولوژیکی).

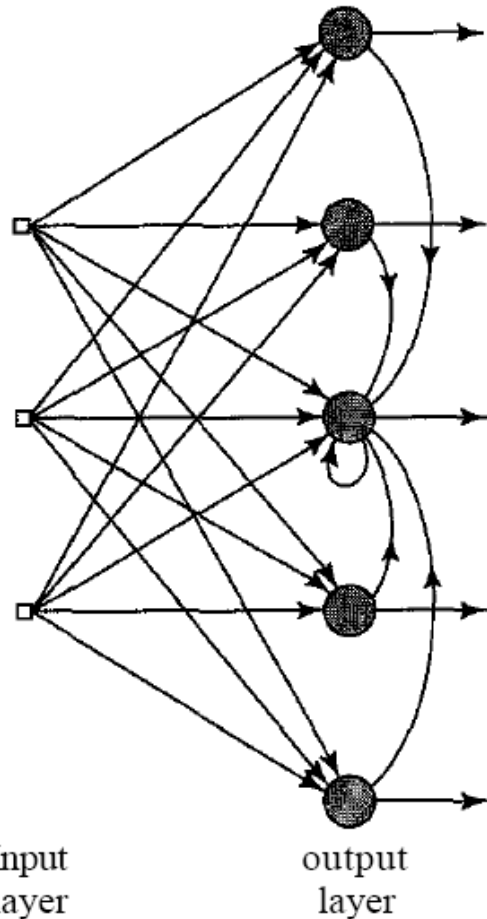
نگاشت خودسازمانده (SOM)

– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

نگاشت خودسازمانده (SOM)

– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها

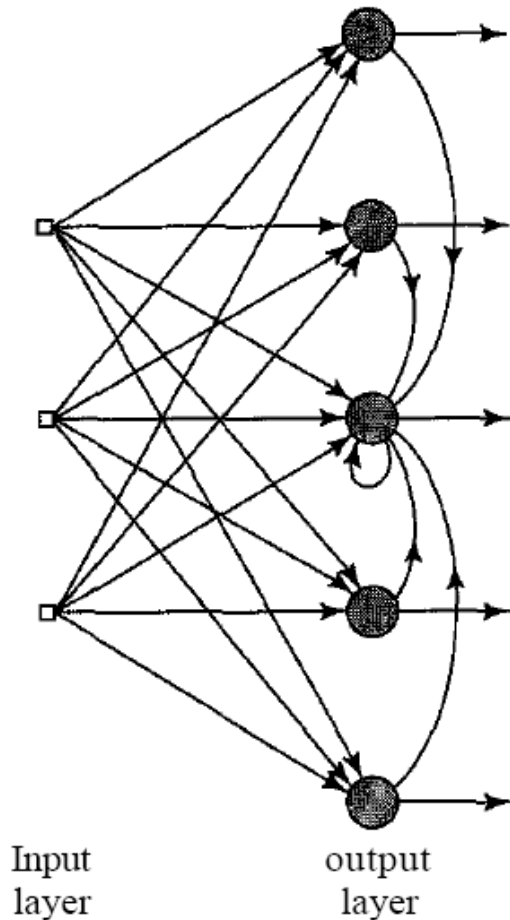


نگاشت خودسازمانده (SOM)

– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها

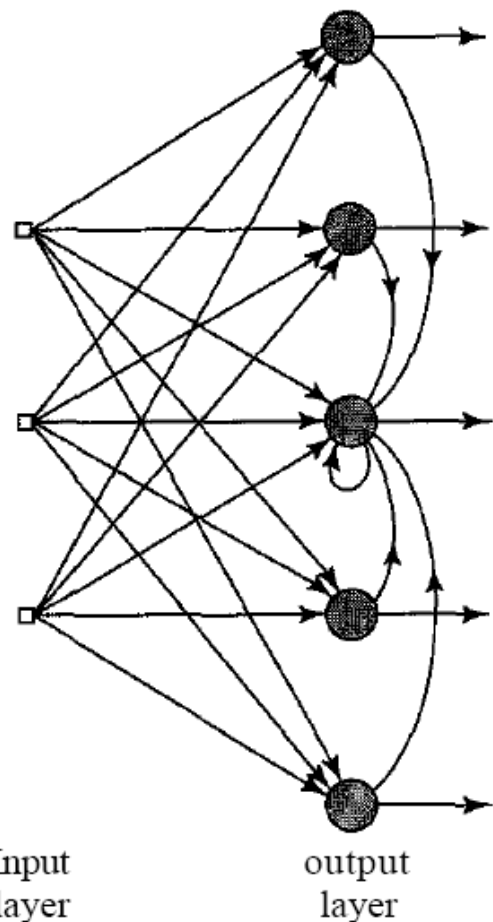
– دو دسته وزن در این جا وجود دارند:



نگاشت خودسازمانده (SOM)

– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



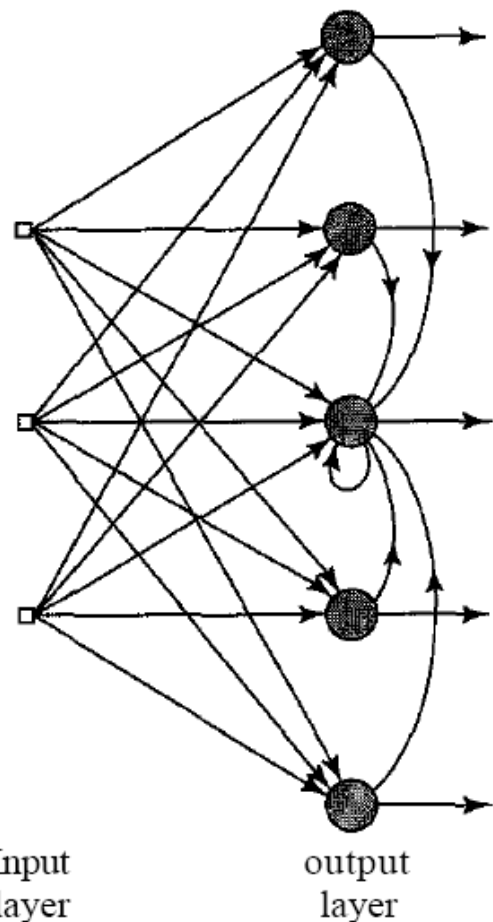
– دو دسته وزن در این جا وجود دارند:

آ- وزن‌های پیش‌خورد: که حامل اطلاعات از ورودی به خروجی هستند.

نگاشت خودسازمانده (SOM)

– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



– دو دسته وزن در این جا وجود دارند:

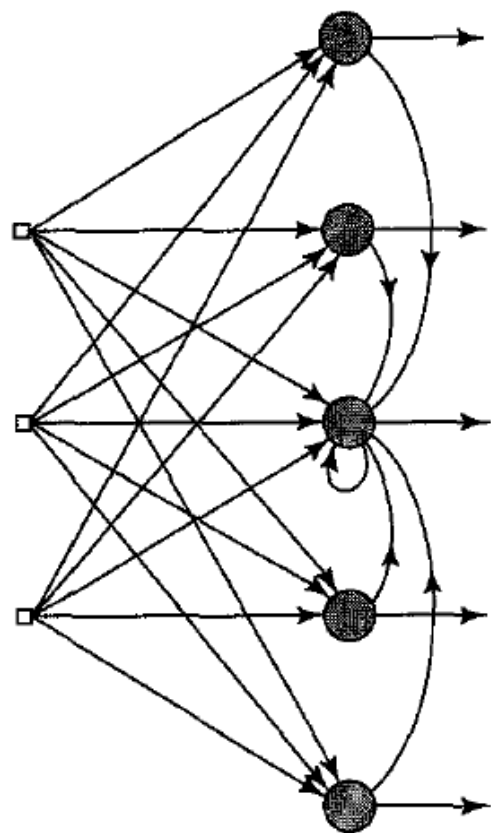
آ- وزن‌های پیش‌خورد: که حامل اطلاعات از ورودی به خروجی هستند.

ب- پس‌خوردهای جانبی و خودی: که اثر تحریک‌کننده یا بازدارنده دارند که این امر بستگی به فاصله از سلول موردنظر دارد.

نگاشت خودسازمانده (SOM)

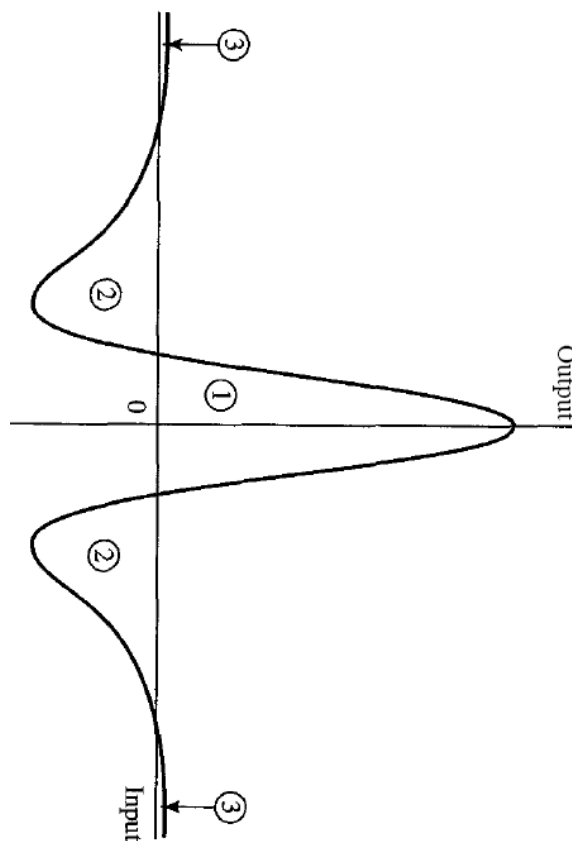
– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



Input
layer

output
layer



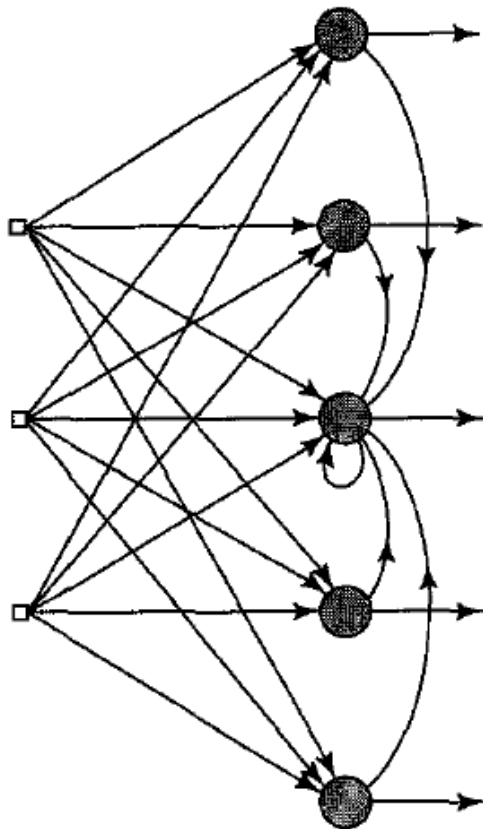
– دو دسته وزن در این جا وجود دارد:
آ- وزن‌های پیش‌خورد: که حامل اطلاعات هستند.

ب- پس‌خوردهای جانبی و خودی: که بازدارنده دارند که این امر بستگی دارد.

نگاشت خودسازمانده (SOM)

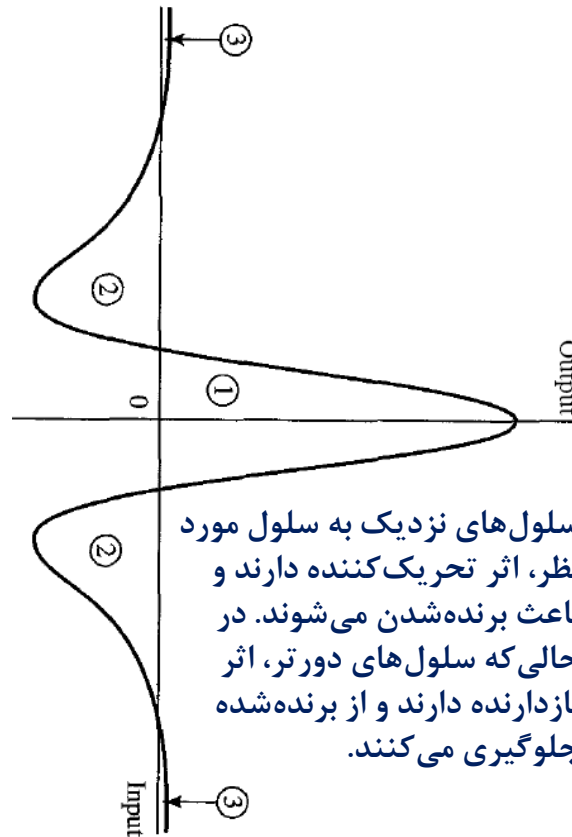
- ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



Input
layer

output
layer



سلول‌های نزدیک به سلول مورد
نظر، اثر تحریک‌کننده دارند و
باعث برنده شدن می‌شوند. در
حالی‌که سلول‌های دورتر، اثر
بازدارنده دارند و از برنده شده
جلوگیری می‌کنند.

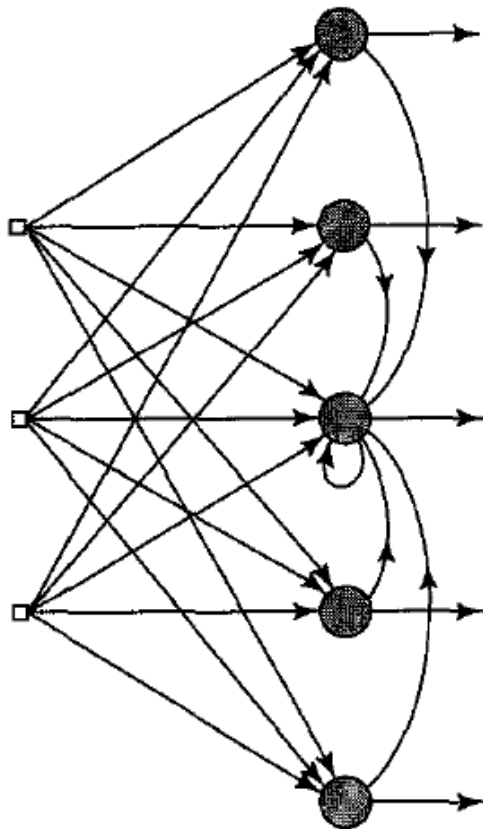
- دو دسته وزن در این جا وجود دارد:
آ- وزن‌های پیش‌خورده: که حامل اط
هستند.

ب- پس‌خورده‌های جانبی و خودی: که
بازدارنده دارند که این امر بستگی
دارد.

نگاشت خودسازمانده (SOM)

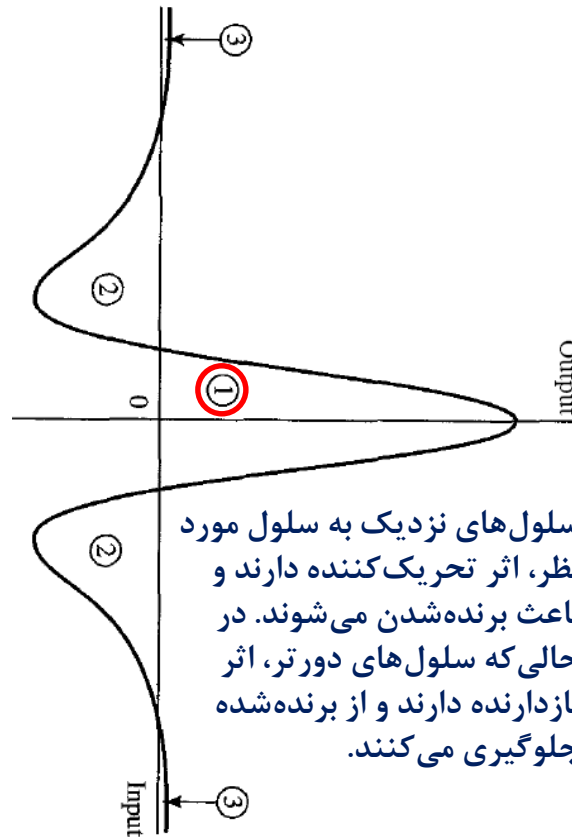
- ابتدا چگونگی به وجود آمدن رقابت بین سلول ها و برنده شدن آن ها بررسی می کنیم.

شبکه ای با یک بُعد از سلول ها



Input
layer

output
layer



- دو دسته وزن در این جا وجود دارد:
آ- وزن های پیشخورد: که حامل اطلاعات هستند.

ب- پسخوردهای جانبی و خودی: بازدارنده دارند که این امر بستگی دارد.

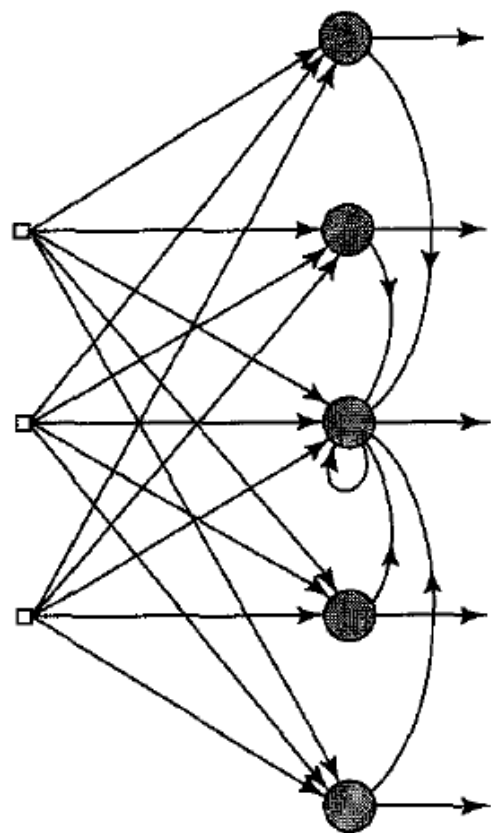
① منطقه پسخوردهای جانبی و خودی تحریک کننده

سلول های نزدیک به سلول مورد نظر، اثر تحریک کننده دارند و باعث برنده شدن می شوند. در حالی که سلول های دورتر، اثر بازدارنده دارند و از برنده شده جلوگیری می کنند.

نگاشت خودسازمانده (SOM)

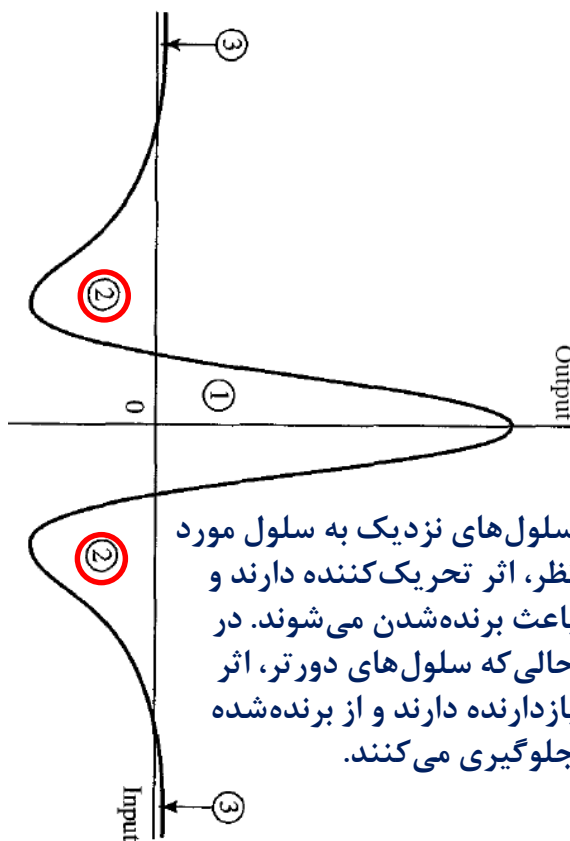
– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



Input
layer

output
layer



– دو دسته وزن در این جا وجود دارد:
آ- وزن‌های پیشخورد: که حامل اطلاعات هستند.

ب- پسخوردهای جانبی و خودی:
بازدارنده دارند که این امر بستگی دارد.

① منطقه پسخوردهای جانبی و خودی
تحریک کننده

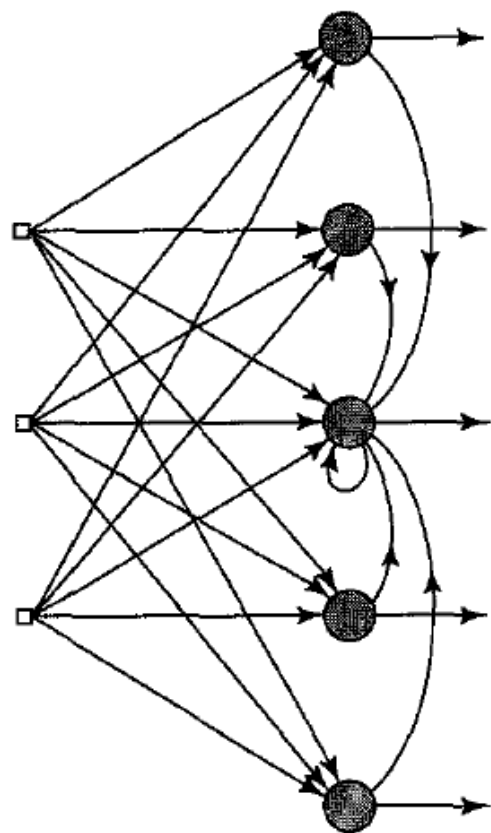
② منطقه پسخوردهای جانبی بازدارنده

سلول‌های نزدیک به سلول مورد نظر، اثر تحریک کننده دارند و باعث برنده شدن می‌شوند. در حالی که سلول‌های دورتر، اثر بازدارنده دارند و از برنده شده جلوگیری می‌کنند.

نگاشت خودسازمانده (SOM)

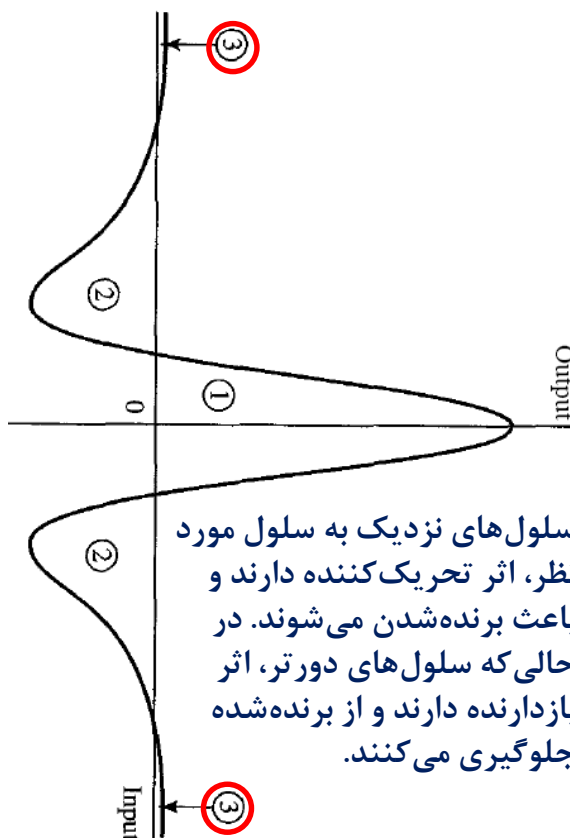
– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



Input
layer

output
layer



سلول‌های نزدیک به سلول مورد نظر، اثر تحریک‌کننده دارند و باعث برنده شدن می‌شوند. در حالی که سلول‌های دورتر، اثر بازدارنده دارند و از برنده شده جلوگیری می‌کنند.

– دو دسته وزن در این جا وجود دارد:
آ- وزن‌های پیش‌خورد: که حامل اطلاعات هستند.

ب- پس‌خوردهای جانبی و خودی: بازدارنده دارند که این امر بستگی دارد.

① منطقه پس‌خوردهای جانبی و خودی تحریک‌کننده

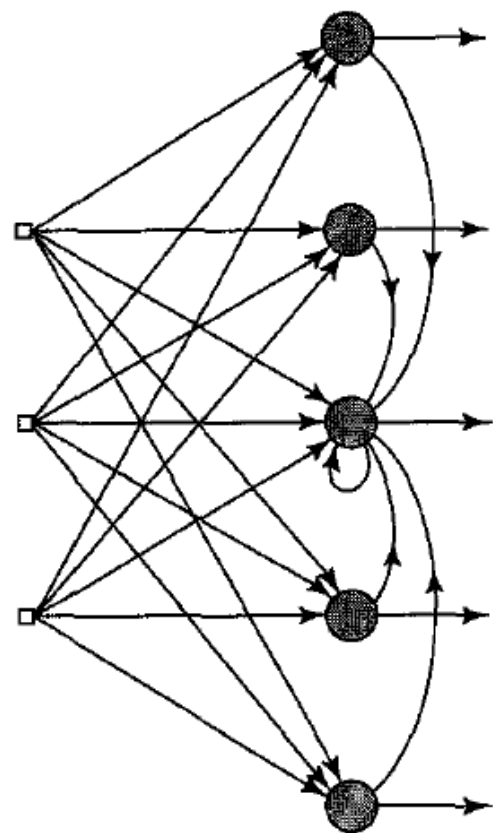
② منطقه پس‌خوردهای جانبی بازدارنده

③ منطقه پس‌خوردهای تحریک‌کننده

نگاشت خودسازمانده (SOM)

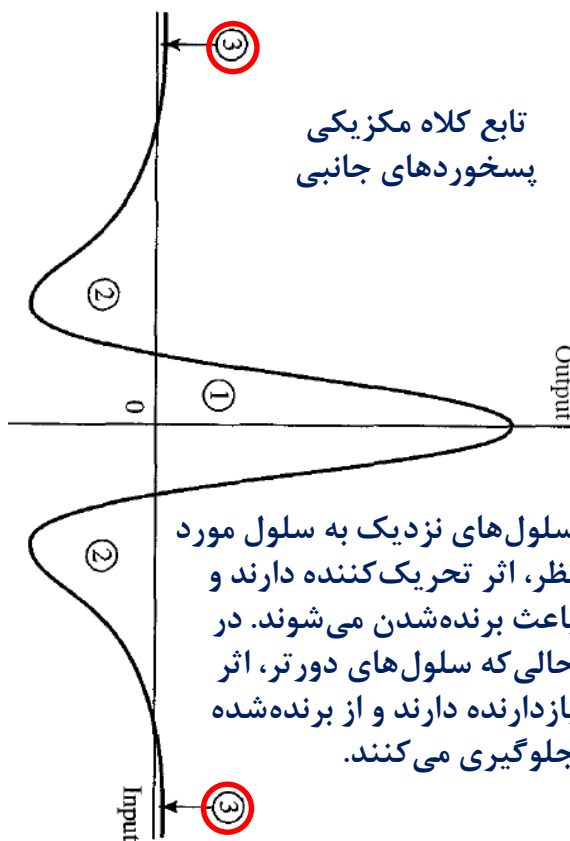
– ابتدا چگونگی به وجود آمدن رقابت بین سلول‌ها و برنده شدن آن‌ها بررسی می‌کنیم.

شبکه‌ای با یک بُعد از سلول‌ها



Input
layer

output
layer



تابع کلاه مکزیک
پسخوردهای جانبی

– دو دسته وزن در این جا وجود دارد:
آ- وزن‌های پیش‌خورد: که حامل اطلاعات هستند.

ب- پسخوردهای جانبی و خودی: بازدارنده دارند که این امر بستگی دارد.

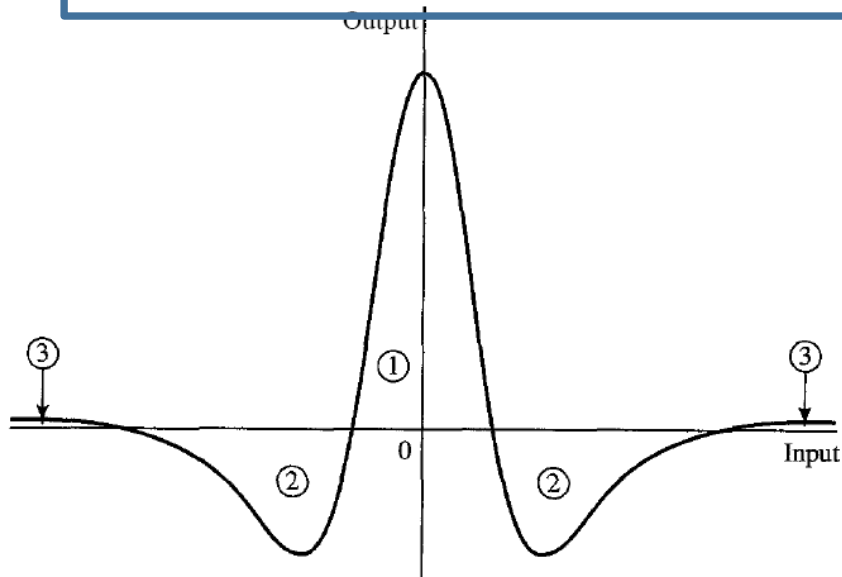
① منطقه پسخوردهای جانبی و خودی تحریک‌کننده

② منطقه پسخوردهای جانبی بازدارنده

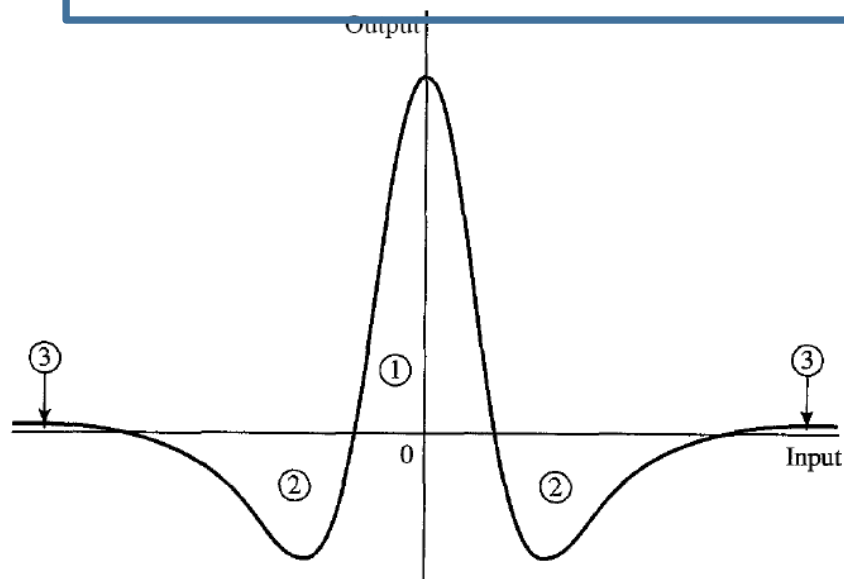
③ منطقه پسخوردهای تحریک‌کننده

نگاشت خودسازمانده (SOM)

- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:



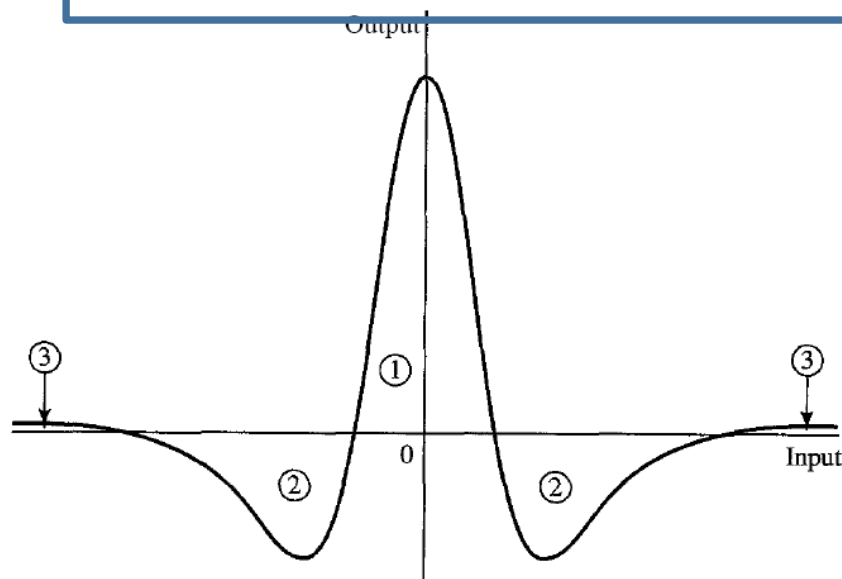
نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

نگاشت خودسازمانده (SOM)

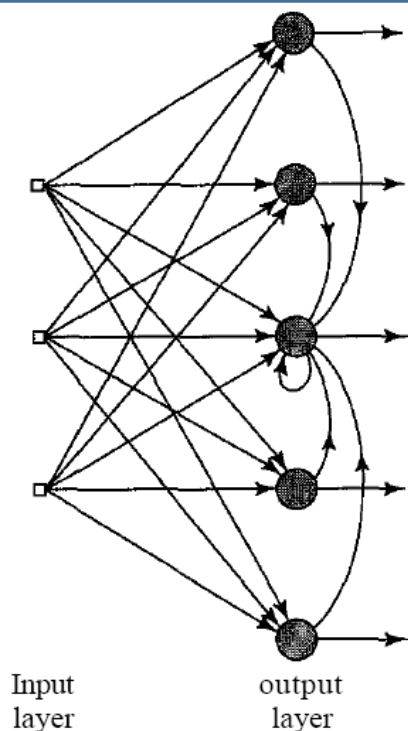


- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

نگاشت خودسازمانده (SOM)



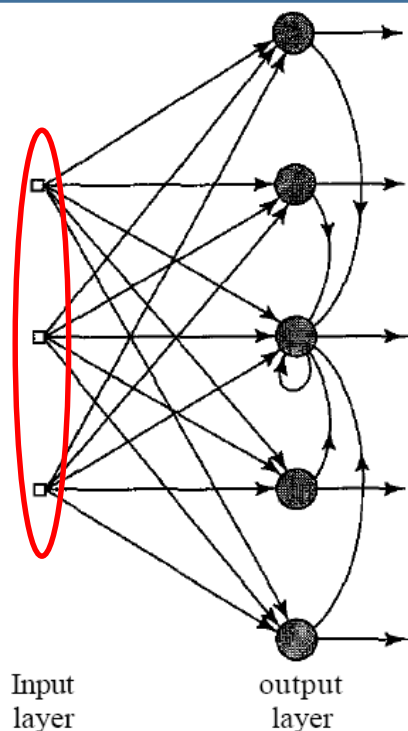
- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

- اختصارات:

نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

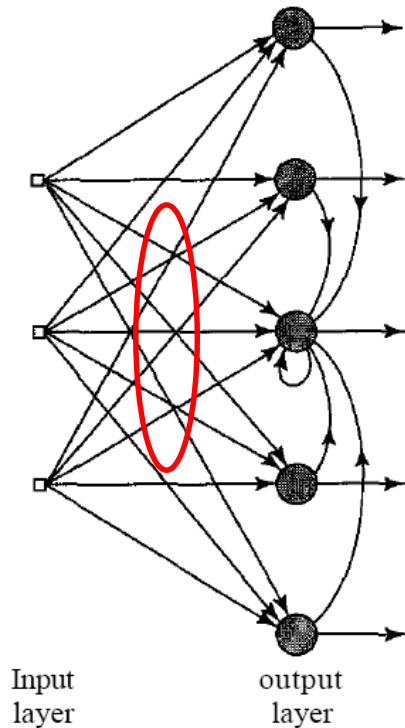
آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

- اختصارات:

سیگنال های ورودی $[x_1 \quad \dots \quad x_m]^T$

نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

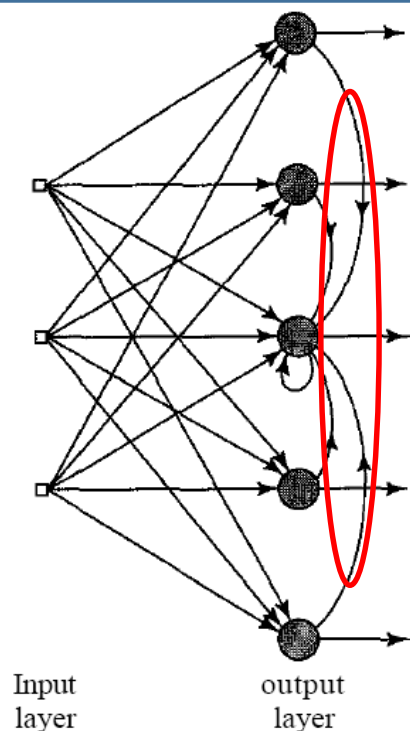
ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

- اختصارات:

سیگنال های ورودی $[x_1 \quad \dots \quad x_m]^T$

وزن های پیشخورد متصل به سلول j ام $[w_{j1} \quad \dots \quad w_{jm}]^T$

نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

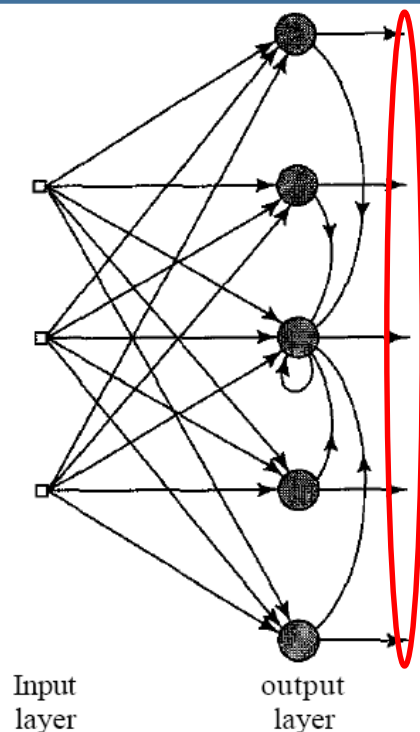
- اختصارات:

سیگنال های ورودی $[x_1 \ \cdots \ x_m]^T$

وزن های پیشخورد متصل به سلول j ام $[w_{j1} \ \cdots \ w_{jm}]^T$

وزن های پسخورد جانبی برای سلول j ام. K شعاع موثر $[c_{j,-K} \ \cdots \ c_{j,-1} \ c_{j,0} \ c_{j,1} \ \cdots \ c_{j,K}]^T$

نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

- اختصارات:

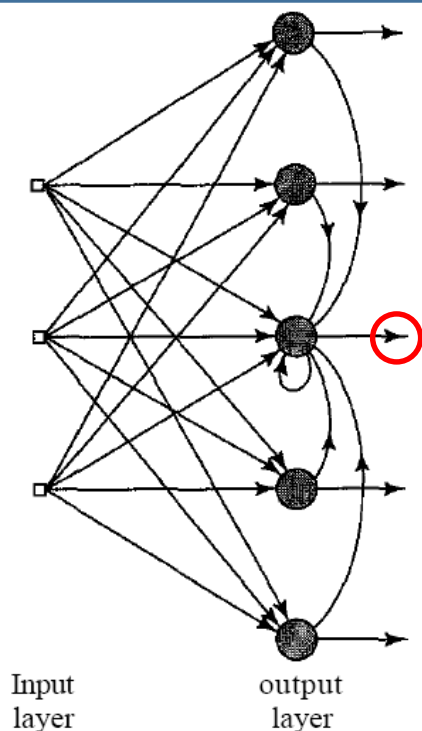
سیگنال های ورودی $[x_1 \cdots x_m]^T$

وزن های پیشخورد متصل به سلول j ام $[w_{j1} \cdots w_{jm}]^T$

وزن های پسخورد جانبی برای سلول j ام. K شعاع موثر $[c_{j,-K} \cdots c_{j,-1} \ c_{j,0} \ c_{j,1} \cdots c_{j,K}]^T$

خروجی های شبکه $[y_1 \cdots y_l]^T$

نگاشت خودسازمانده (SOM)



- در این نوع شبکه، دو مشخصه مهم را می توان تمیز داد:

آ- شبکه سعی در ایجاد دسته های محلی دارد که به آن حباب فعال (active bubble) می گویند.

ب- مکان این حباب فعال توسط سیگنال های ورودی تشکیل می شود.

- اختصارات:

سیگنال های ورودی $[x_1 \cdots x_m]^T$

وزن های پیشخورد متصل به سلول j ام $[w_{j1} \cdots w_{jm}]^T$

وزن های پسخورد جانبی برای سلول j ام. K شعاع موثر $[c_{j,-K} \cdots c_{j,-1} \ c_{j,0} \ c_{j,1} \cdots c_{j,K}]^T$

خروجی های شبکه $[y_1 \cdots y_l]^T$

- برای خروجی j ام شبکه می توان نوشت:

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

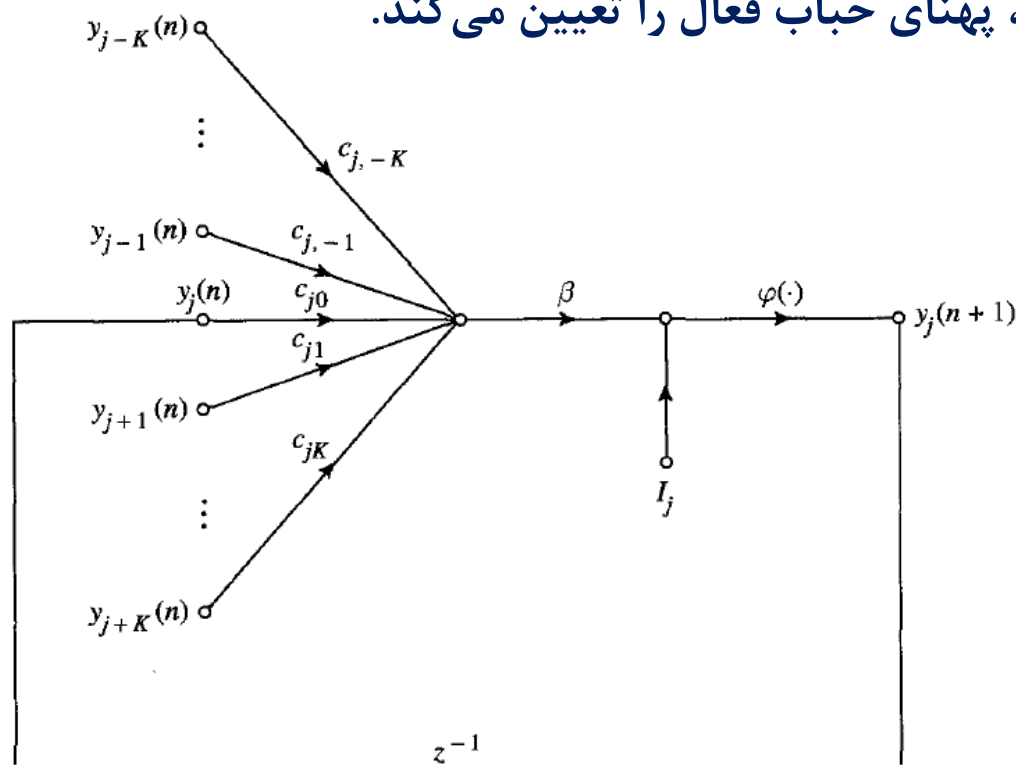
– ضریب β سرعت همگرا شدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{jk} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

- ضریب β سرعت همگراشدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.



نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

– ضریب β سرعت همگرا شدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

– ضریب β سرعت همگرا شدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.

– چنانچه پسخوردهای مثبت (پسخوردهای تحریک‌کننده) قوی‌تر شوند، در این صورت حباب‌های فعال پهن‌تر ولی کوتاه‌تر خواهند شد.

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

- ضریب β سرعت همگرا شدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.

- چنانچه پسخوردهای مثبت (پسخوردهای تحریک‌کننده) قوی‌تر شوند، در این صورت حباب‌های فعال پهن‌تر ولی کوتاه‌تر خواهند شد.

- برعکس، چنانچه پسخوردهای منفی (پسخوردهای بازدارنده) تقویت شوند، در این صورت حباب‌های فعال باریک‌تر و بلندتر خواهند شد.

نگاشت خودسازمانده (SOM)

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$

$\varphi(\cdot)$ تابع غیرخطی سلول‌ها

- ضریب β سرعت همگرا شدن شبکه را کنترل می‌کند و بنابراین، نقش ضریب پسخورد را دارد. این ضریب، پهنای حباب فعال را تعیین می‌کند.

- چنانچه پسخوردهای مثبت (پسخوردهای تحریک‌کننده) قوی‌تر شوند، در این صورت حباب‌های فعال پهن‌تر ولی کوتاه‌تر خواهند شد.

- برعکس، چنانچه پسخوردهای منفی (پسخوردهای بازدارنده) تقویت شوند، در این صورت حباب‌های فعال باریک‌تر و بلندتر خواهند شد.

- واضح است که پسخوردهای منفی خیلی بزرگ شوند، در این صورت هیچگونه حبابی تشکیل نخواهد شد.

نگاشت خودسازمانده (SOM)

مثال:

نگاشت خودسازمانده (SOM)

مثال:

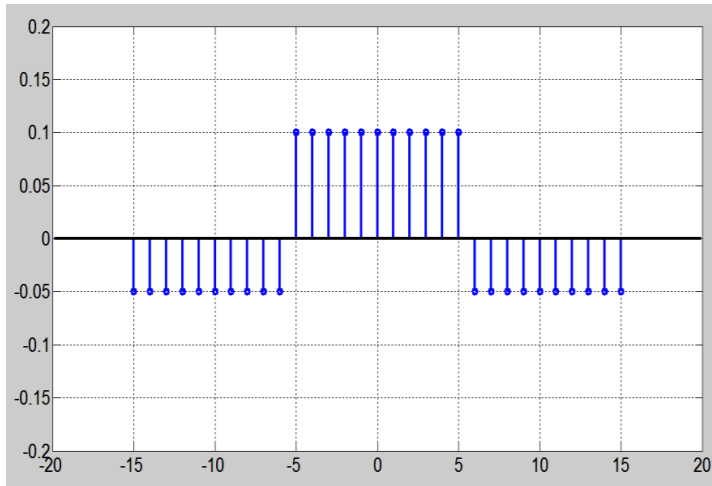
– در این مثال، برای سادگی کار

نگاشت خودسازمانده (SOM)

مثال:

– در این مثال، برای سادگی کار

آ – تابع کلاه مکزیکی به صورت مقابل در نظر گرفته می شود:



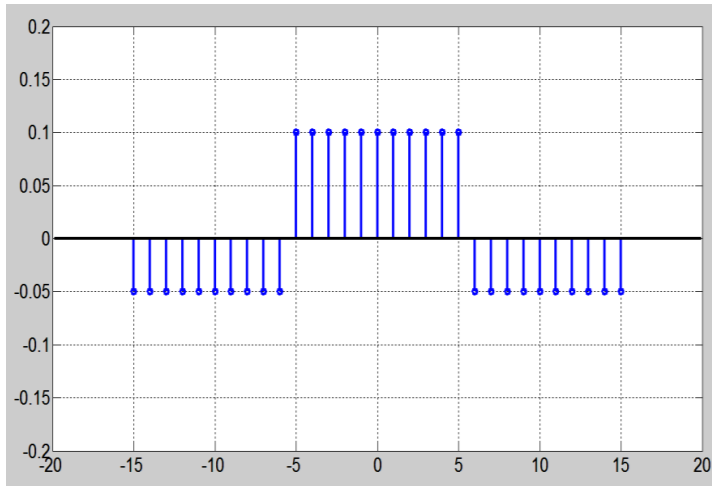
نگاشت خودسازمانده (SOM)

مثال:

- در این مثال، برای سادگی کار

آ- تابع کلاه مکزیکی به صورت مقابل در نظر گرفته می شود:

a- منطقه ③ حذف می شود.



نگاشت خودسازمانده (SOM)

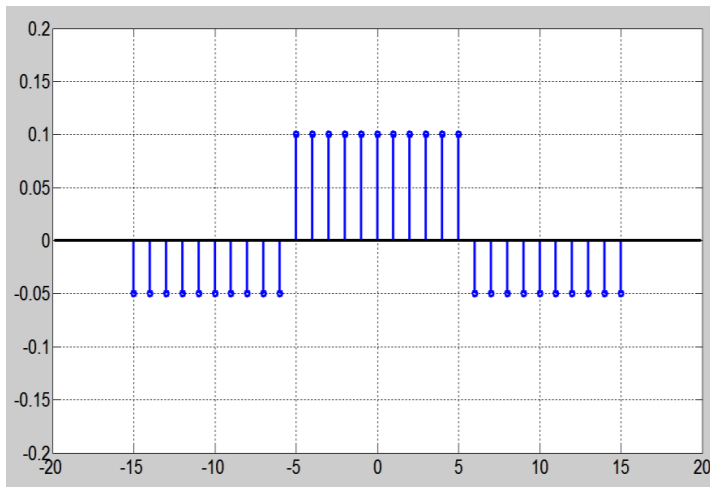
مثال:

- در این مثال، برای سادگی کار

آ- تابع کلاه مکزیکی به صورت مقابل در نظر گرفته می شود:

a- منطقه ③ حذف می شود.

b- مقدار وزن های جانبی تحریک کننده و بازدارنده ثابت می باشد



نگاشت خودسازمانده (SOM)

مثال:

- در این مثال، برای سادگی کار

آ- تابع کلاه مکزیکی به صورت مقابل در نظر گرفته می شود:

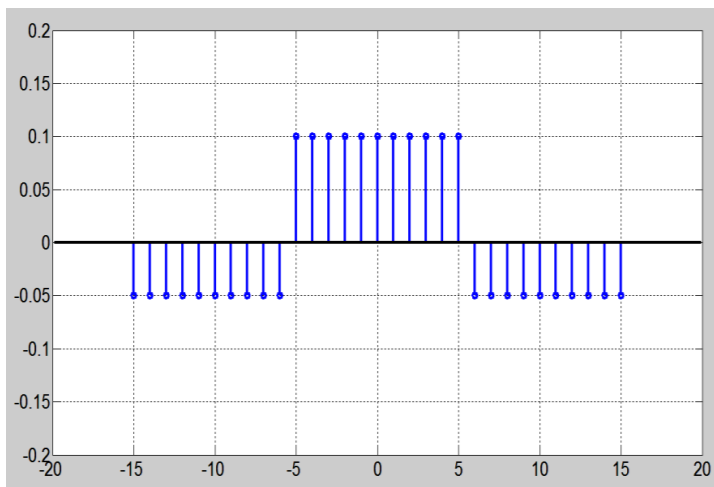
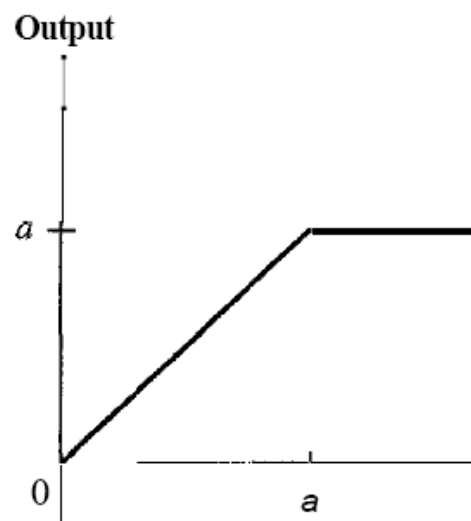
a- منطقه ③ حذف می شود.

b- مقدار وزن های جانبی تحریک کننده و بازدارنده

ثابت می باشد

ب- تابع غیر خطی $\varphi(\cdot)$ به صورت زیر در نظر گرفته می شود:

$$\varphi(x) = \begin{cases} a & x \geq a \\ x & 0 \leq x < a \\ 0 & x < 0 \end{cases}$$



نگاشت خودسازمانده (SOM)

مثال:

– تعداد سلول‌های شبکه برابر $N = 51$ می‌باشد و مقدار ورودی به سلول j ام در اثر سیگنال‌های ورودی برابر است با

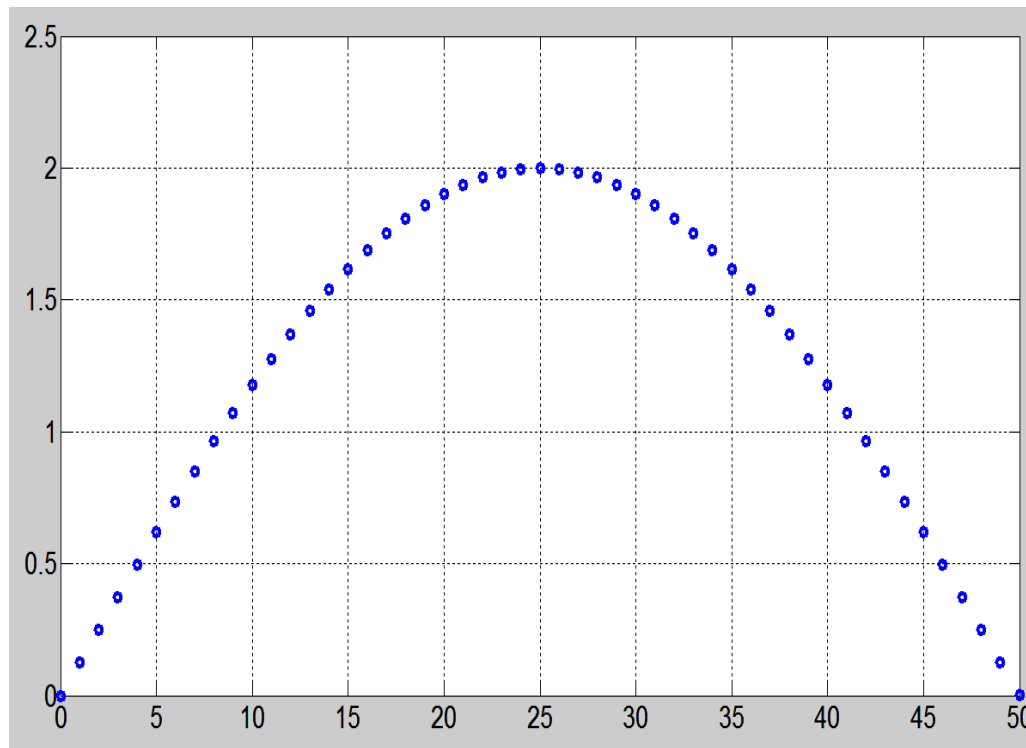
$$I_j = \sum_{i=1}^l w_{ji} x_i = 2 \sin\left(\frac{\pi j}{50}\right), \quad 0 \leq j \leq 50$$

نگاشت خودسازمانده (SOM)

مثال:

– تعداد سلول‌های شبکه برابر $N = 51$ می‌باشد و مقدار ورودی به سلول j ام در اثر سیگنال‌های ورودی برابر است با

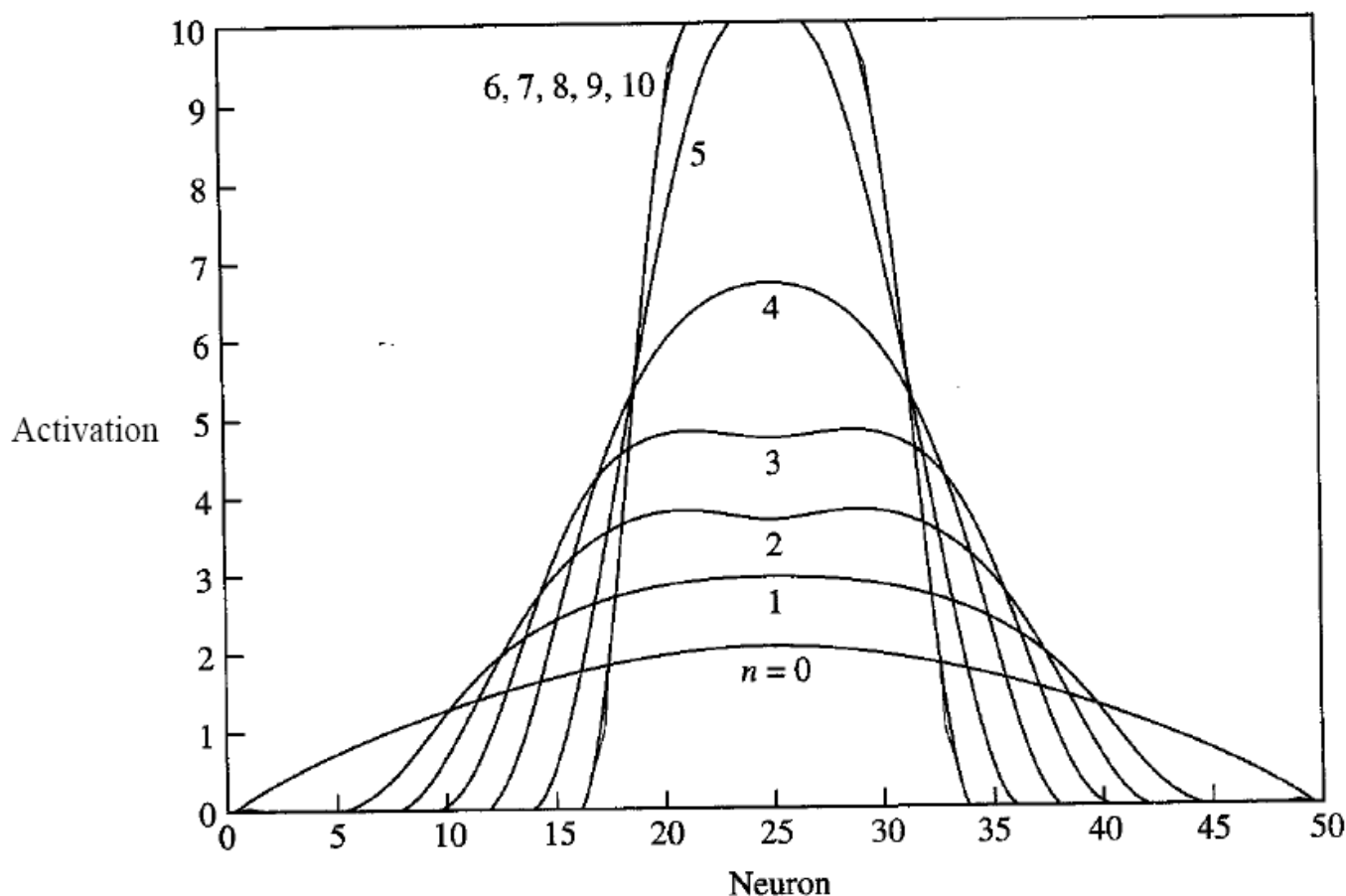
$$I_j = \sum_{i=1}^l w_{ji} x_i = 2 \sin\left(\frac{\pi j}{50}\right), \quad 0 \leq j \leq 50$$



نگاشت خودسازمانده (SOM)

- نتیجه شبیه سازی برای $\beta = 2$

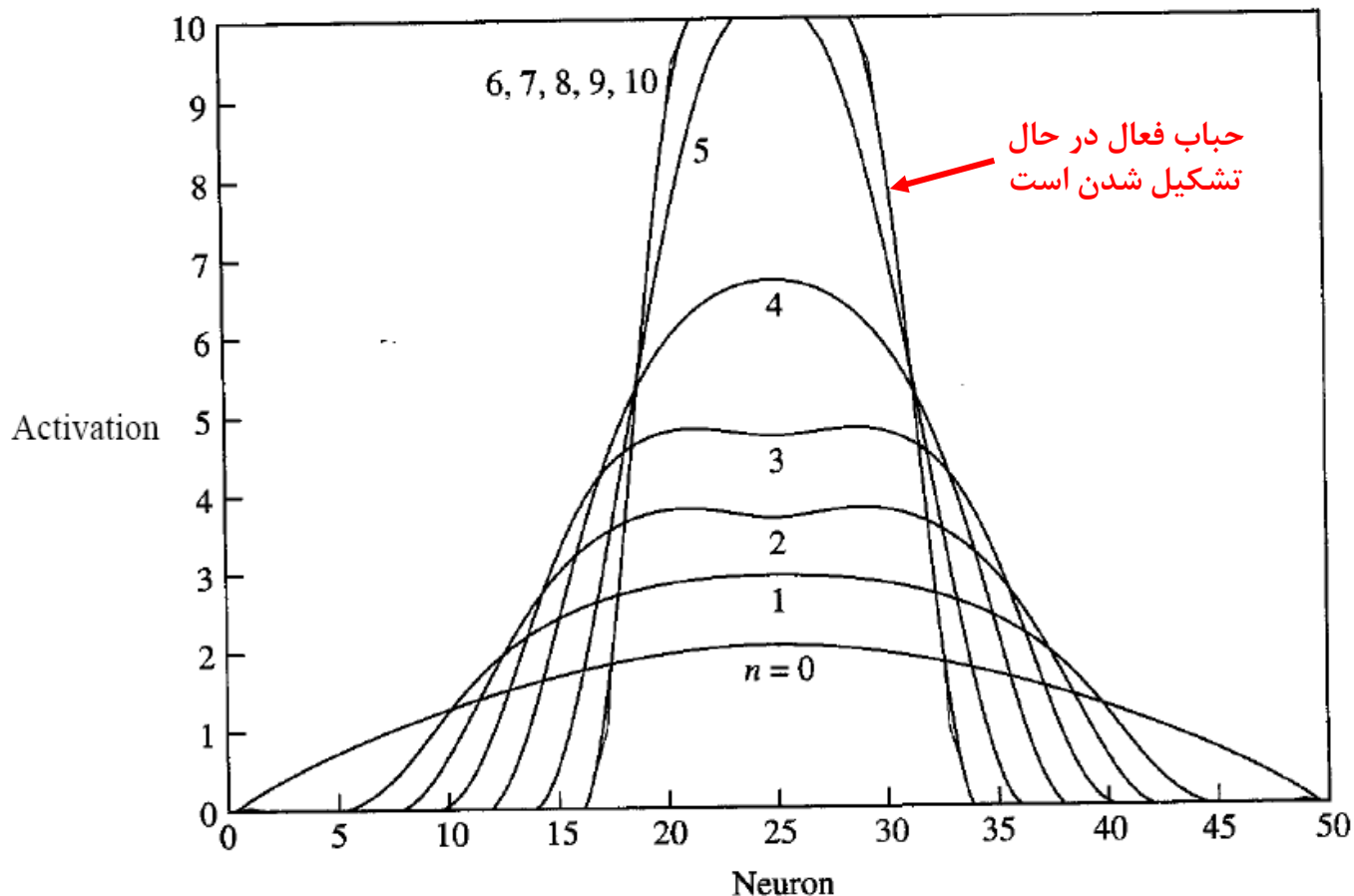
$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$



نگاشت خودسازمانده (SOM)

- نتیجه شبیه سازی برای $\beta = 2$

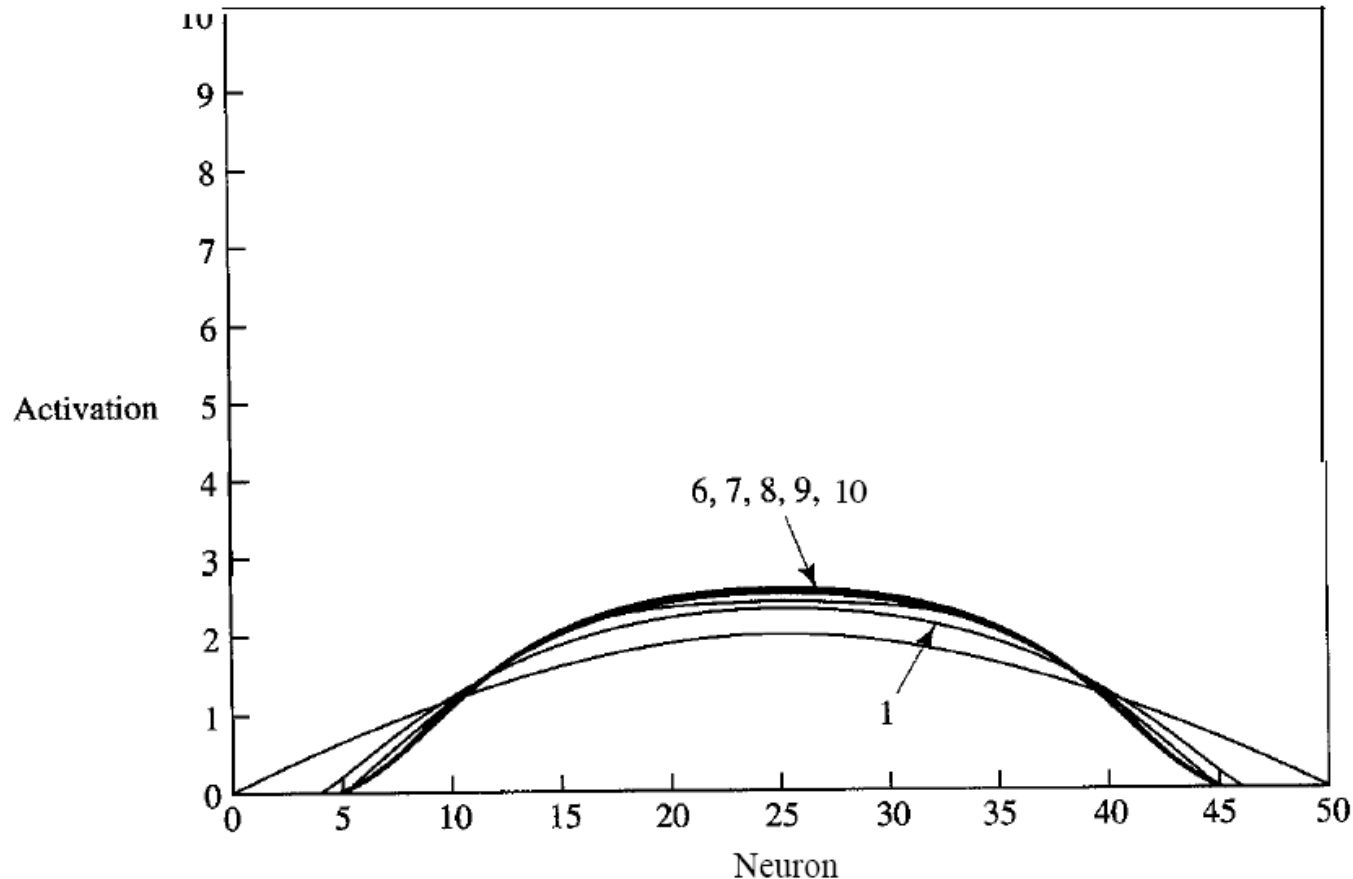
$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$



نگاشت خودسازمانده (SOM)

- نتیجه شبیه سازی برای $\beta = 0.75$

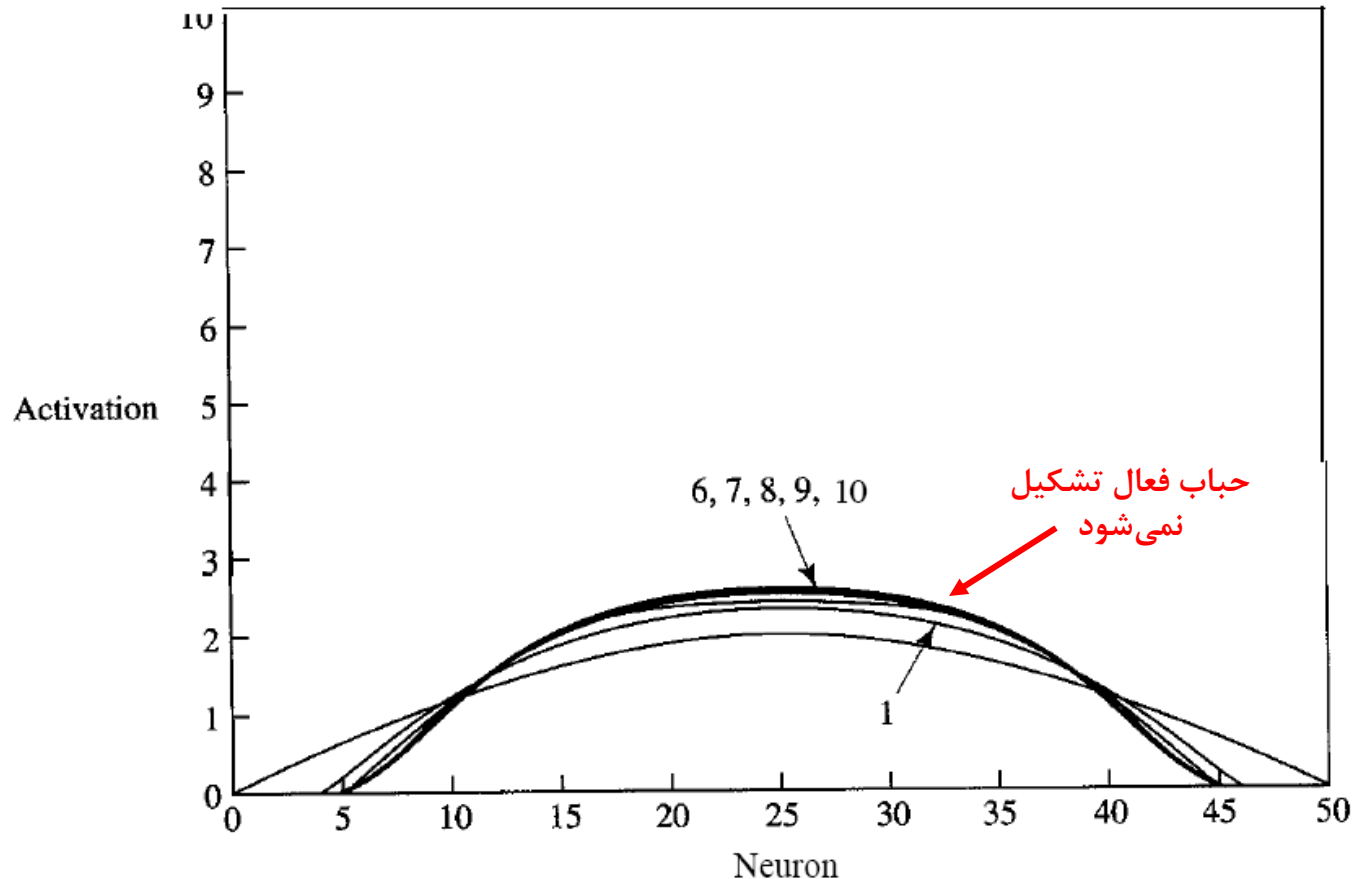
$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$



نگاشت خودسازمانده (SOM)

- نتیجه شبیه سازی برای $\beta = 0.75$

$$y_j(n+1) = \varphi \left[\sum_{i=1}^l w_{ji} x_i + \beta \sum_{k=-K}^K c_{ji} y_{j+k}(n) \right], \quad j = 1, \dots, N$$



نگاشت خودسازمانده (SOM)

– از این مثال می توان نتایج زیر را گرفت:

نگاشت خودسازمانده (SOM)

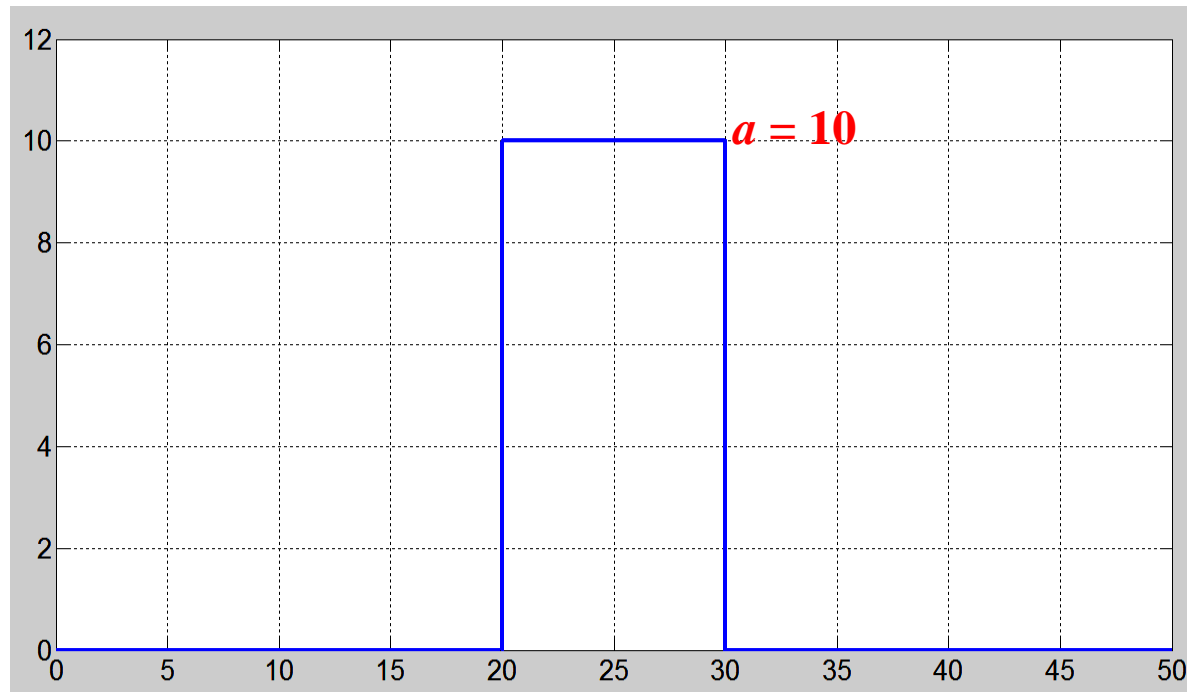
– از این مثال می توان نتایج زیر را گرفت:

۱– واضح است که $\beta = 2$ تشکیل حباب فعال را می دهد به طوری که در حالت ایده آل باید چنین شکلی به دست آید:

نگاشت خودسازمانده (SOM)

– از این مثال می توان نتایج زیر را گرفت:

۱– واضح است که $\beta = 2$ تشکیل حباب فعال را می دهد به طوری که در حالت ایده آل باید چنین شکلی به دست آید:

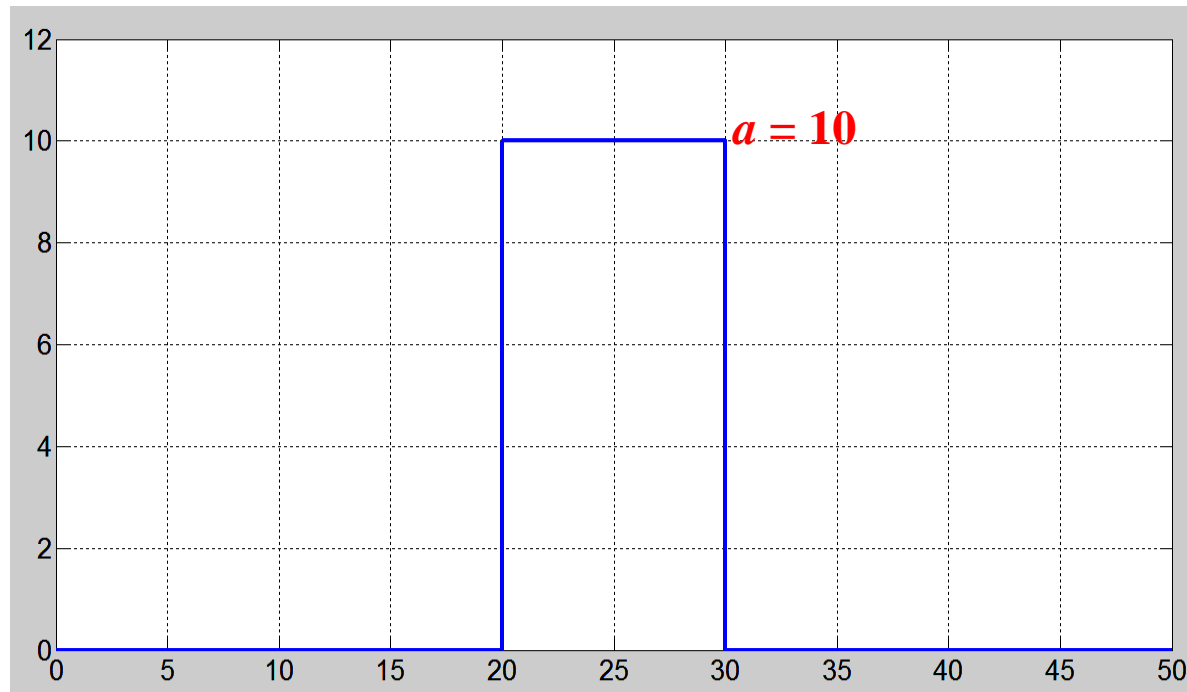


نگاشت خودسازمانده (SOM)

– از این مثال می توان نتایج زیر را گرفت:

۱– واضح است که $\beta = 2$ تشکیل حباب فعال را می دهد به طوری که در حالت ایده آل باید چنین شکلی به دست آید:

$$y_j = \begin{cases} a, & \text{neuron } j \text{ is inside the bubble} \\ 0, & \text{neuron } j \text{ is outside the bubble} \end{cases}$$



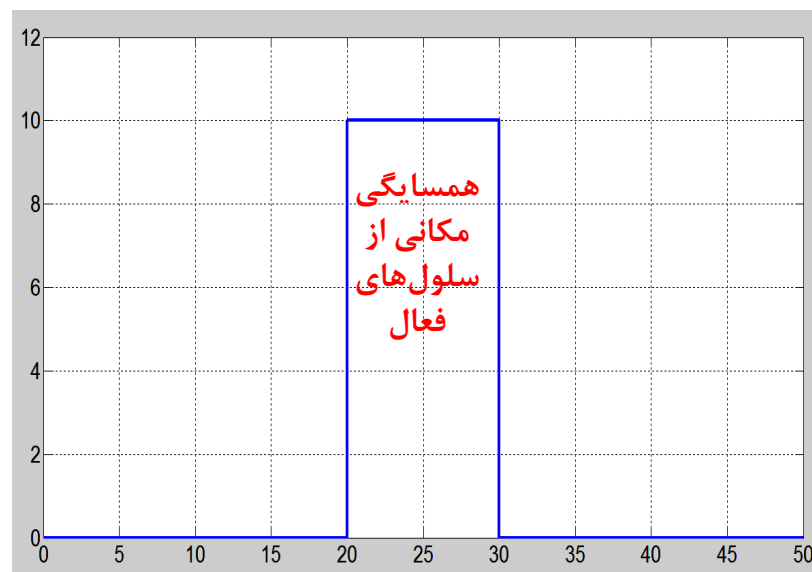
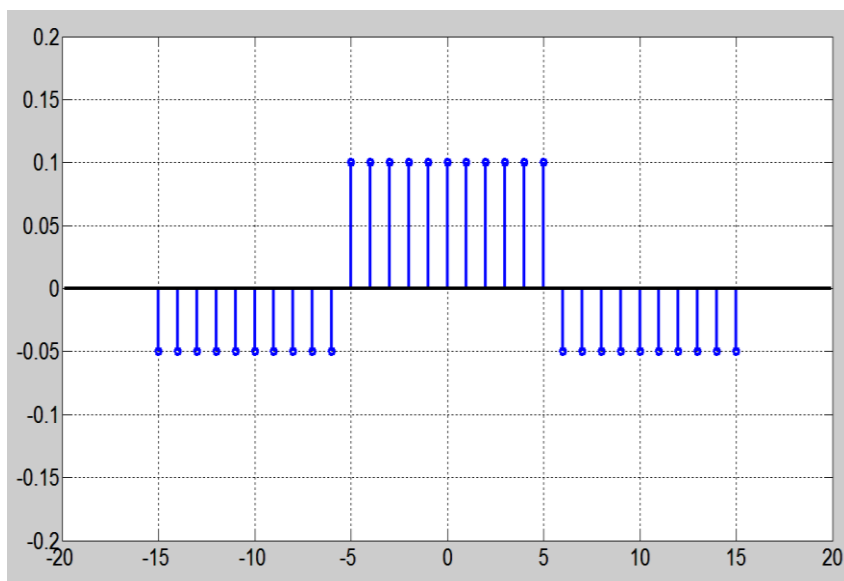
نگاشت خودسازمانده (SOM)

– از این مثال می‌توان نتایج زیر را گرفت:

نگاشت خودسازمانده (SOM)

– از این مثال می توان نتایج زیر را گرفت:

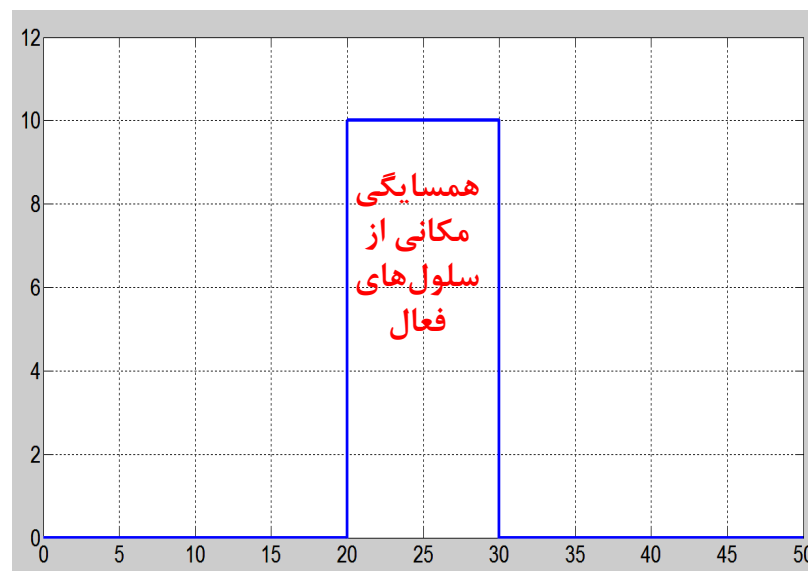
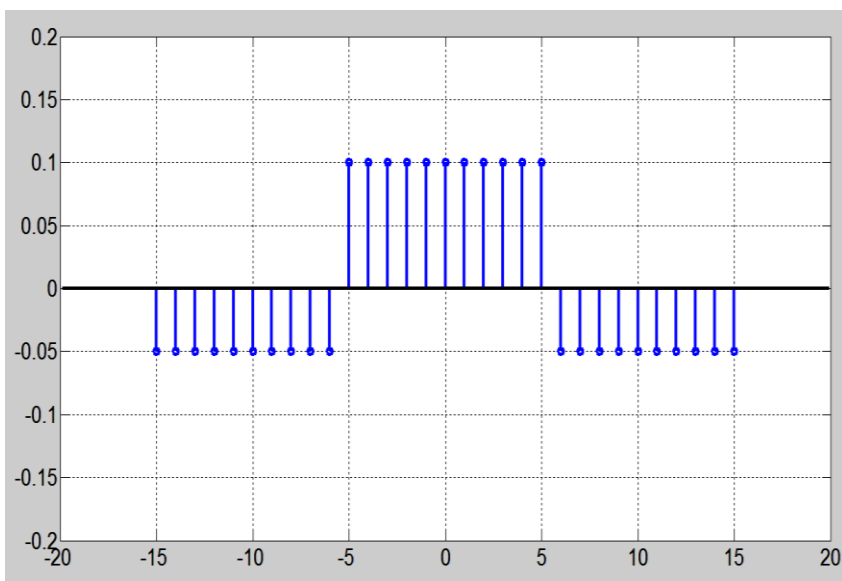
۲– به جای پسخوردهای جانبی و تابع کلاه مکزیکی و تنظیم مناسب ضریب β ، می توان یک «همسایگی مکانی (توپولوژیکی) از سلول های فعال» تعریف کرد به طوری که سلول های داخل این محدوده، برنده و سلول های خارج از این محدوده، بازنده خواهند بود.



نگاشت خودسازمانده (SOM)

– از این مثال می توان نتایج زیر را گرفت:

۲– به جای پسخوردهای جانبی و تابع کلاه مکزیکی و تنظیم مناسب ضریب β ، می توان یک «همسایگی مکانی (توپولوژیکی) از سلول های فعال» تعریف کرد به طوری که سلول های داخل این محدوده، برنده و سلول های خارج از این محدوده، بازنده خواهند بود. علاوه بر آن، به جای تنظیم وزن های جانبی می توان محدوده همسایگی را پهن تر یا باریک تر کرد به طوری که عریض تر شدن محدوده همسایگی معادل با قوی تر ساختن پسخوردهای مثبت (تحریک کننده) می باشد و باریک تر شدن این محدوده معادل با قوی تر ساختن پسخوردهای منفی است.

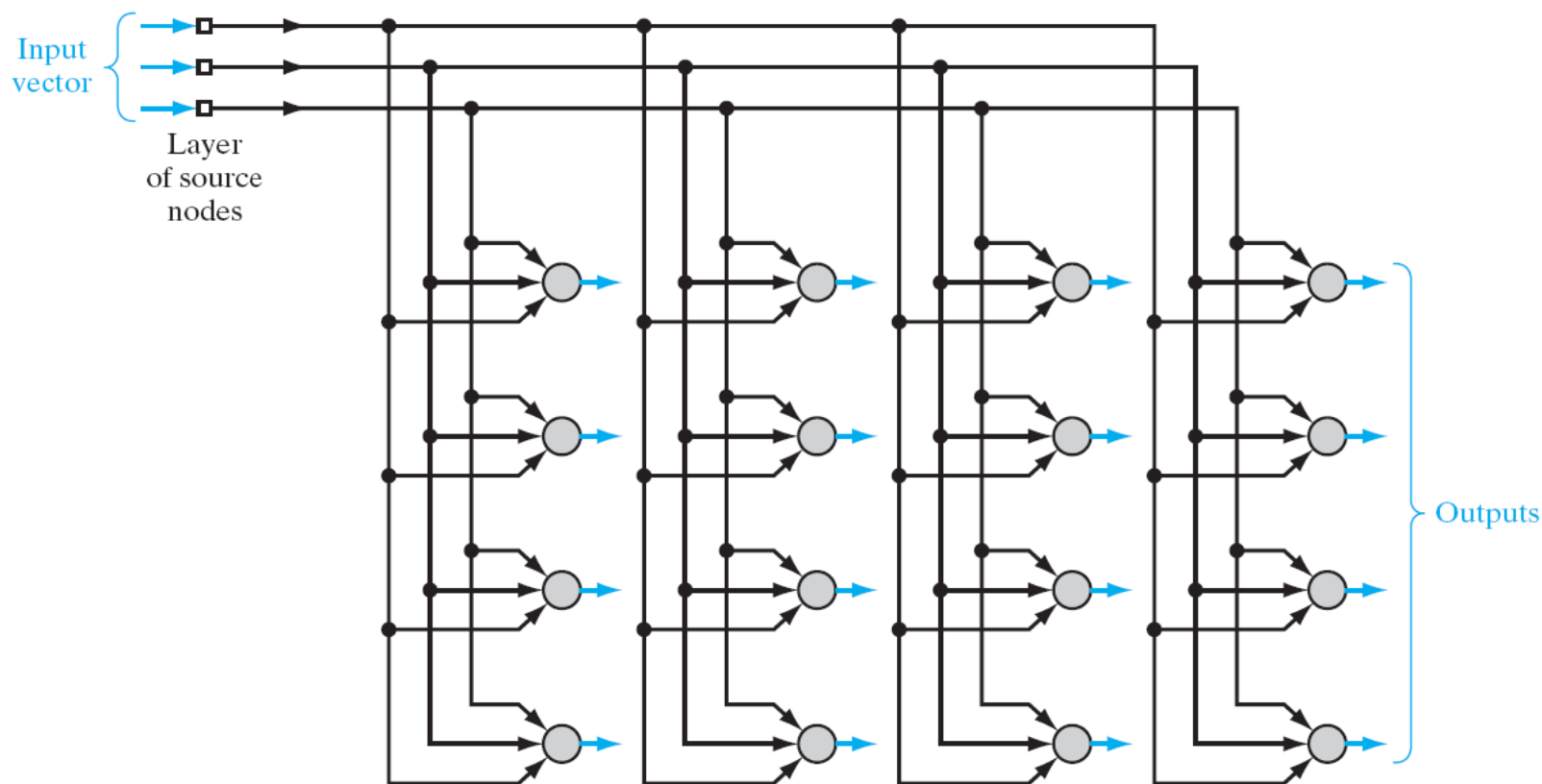


نگاشت خودسازمانده (SOM)

– چنانچه این دو نکته را برای شبکه ای با یک یا دو بُعد از سلول ها اجرا کنیم، شبکه ای به نام نگاشت خودسازمانده (Self-Organizing Map \equiv SOM) یا نگاشت خودسازمانده ویژگی ها (Self-Organizing Feature Map \equiv SOFM) به دست خواهد آمد.

نگاشت خودسازمانده (SOM)

– چنانچه این دو نکته را برای شبکه ای با یک یا دو بُعد از سلول ها اجرا کنیم، شبکه ای به نام نگاشت خودسازمانده (Self-Organizing Map \equiv SOM) یا نگاشت خودسازمانده ویژگی ها (Self-Organizing Feature Map \equiv SOFM) به دست خواهد آمد.



نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– اصول این الگوریتم را می توان به صورت زیر خلاصه کرد:

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– اصول این الگوریتم را می توان به صورت زیر خلاصه کرد:

- انتخاب شبکه ای با یک یا دو بُعد از سلول ها که توابع انتخاب کننده ای را بر روی ورودی ها با ابعاد دلخواه اعمال کند.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– اصول این الگوریتم را می توان به صورت زیر خلاصه کرد:

- انتخاب شبکه ای با یک یا دو بُعد از سلول ها که توابع انتخاب کننده ای را بر روی ورودی ها با ابعاد دلخواه اعمال کند.
- استفاده از مکانیزمی که این توابع انتخاب کننده را با هم مقایسه کرده و سلولی که دارای بزرگترین انتخاب باشد را استخراج کند.

نگاشت خودسازمانده (SOM)

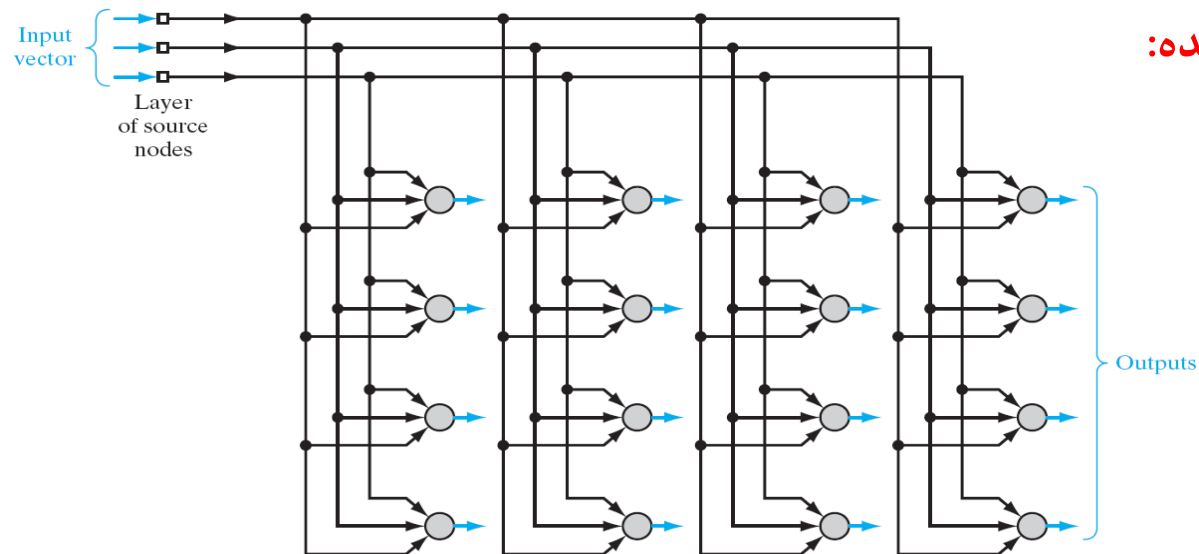
الگوریتم نگاشت خودسازمانده:

– اصول این الگوریتم را می توان به صورت زیر خلاصه کرد:

- انتخاب شبکه ای با یک یا دو بُعد از سلول ها که توابع انتخاب کننده ای را بر روی ورودی ها با ابعاد دلخواه اعمال کند.
- استفاده از مکانیزمی که این توابع انتخاب کننده را با هم مقایسه کرده و سلولی که دارای بزرگترین انتخاب باشد را استخراج کند.
- به کار بردن پردازشی تطبیقی که سلول های فعال را قادر سازد تا مقدار تابع انتخاب کننده شان را افزایش دهد.

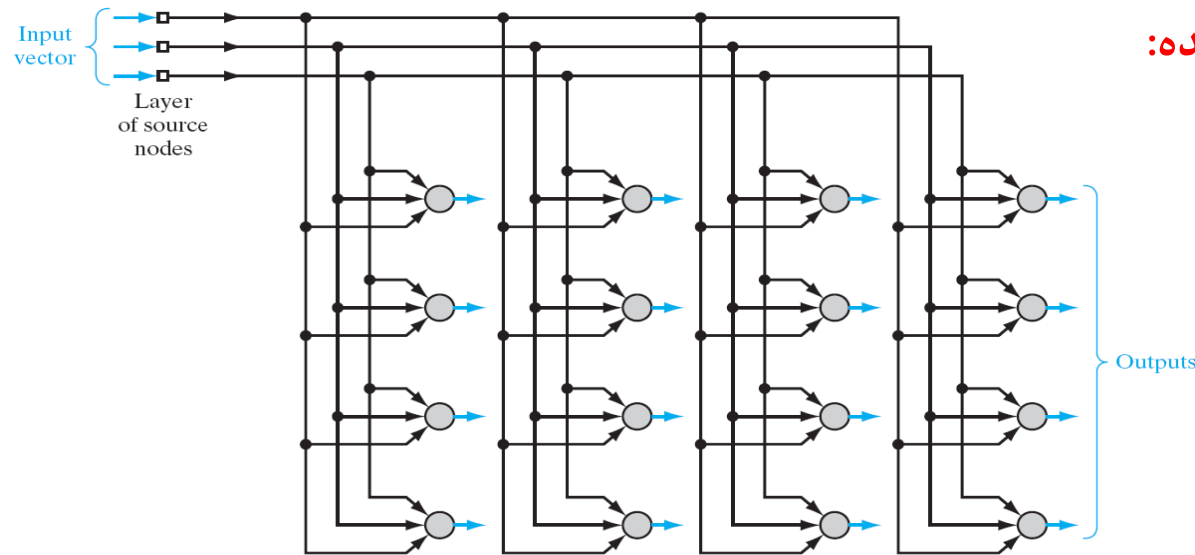
نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



نگاشت خودسازمانده (SOM)

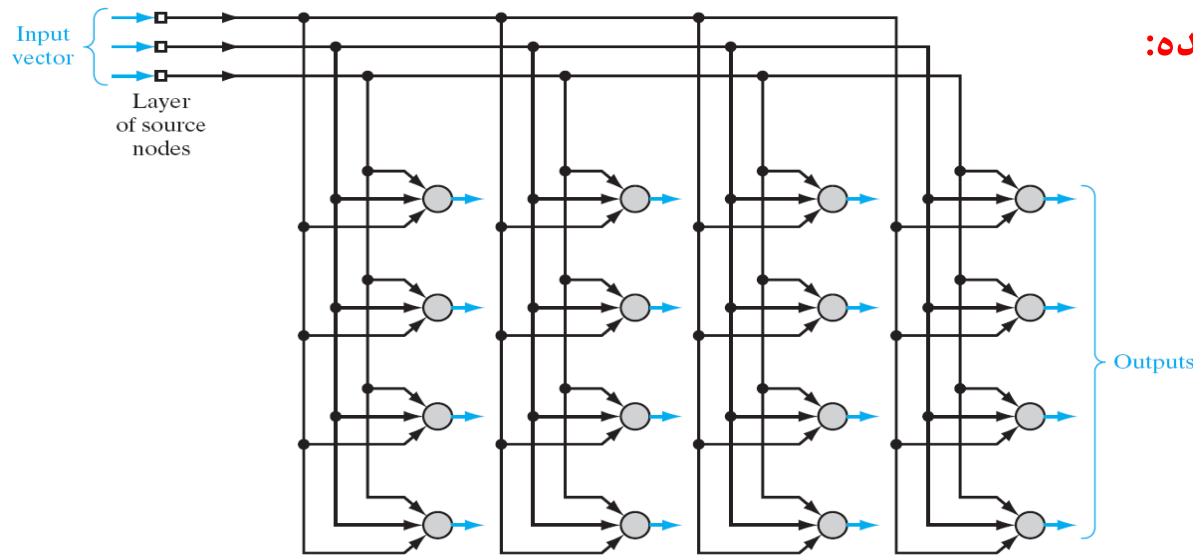
الگوریتم نگاشت خودسازمانده:



– ابتدا یکی از بردارهای ورودی را به شبکه اعمال کرده و سلولی که مناسب‌ترین پاسخ را تولید کند، پیدا می‌کنیم.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



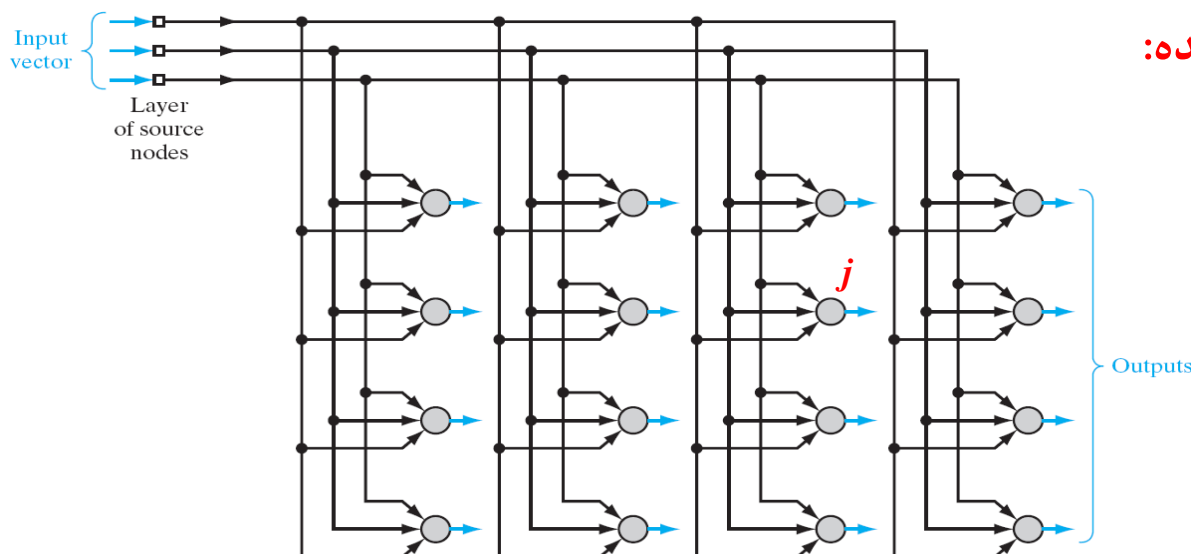
- ابتدا یکی از بردارهای ورودی را به شبکه اعمال کرده و سلولی که مناسب ترین پاسخ را تولید کند، پیدا می کنیم.

- برای این کار، مقدار ورودی خطی به سلول ها را محاسبه کرده

$$v_j = \mathbf{w}_j^T \mathbf{x}, \quad j = 1, \dots, N$$

و بزرگترین را انتخاب می کنیم.

نگاشت خودسازمانده (SOM)



الگوریتم نگاشت خودسازمانده:

- ابتدا یکی از بردارهای ورودی را به شبکه اعمال کرده و سلولی که مناسب ترین پاسخ را تولید کند، پیدا می کنیم.

- برای این کار، مقدار ورودی خطی به سلول ها را محاسبه کرده

$$v_j = \mathbf{w}_j^T \mathbf{x}, \quad j = 1, \dots, N$$

و بزرگترین را انتخاب می کنیم.

- به این ترتیب، محل تشکیل حباب فعال را یافته ایم.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (w_j) و بردار ورودی (x) می باشد

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (\mathbf{w}_j) و بردار ورودی (\mathbf{x}) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(\mathbf{x})$ نشان دهیم، در این صورت

$$i(\mathbf{x}) = \min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j = 1, \dots, N$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (\mathbf{w}_j) و بردار ورودی (\mathbf{x}) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(\mathbf{x})$ نشان دهیم، در این صورت

$$i(\mathbf{x}) = \min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j = 1, \dots, N$$

- سپس باید محدوده همسایگی سلول برنده را انتخاب کنیم که می تواند تابعی از زمان گسسته n باشد و آن را با $\Lambda_{i(\mathbf{x})}(n)$ نشان می دهیم (تابع همسایگی).

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (w_j) و بردار ورودی (x) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(x)$ نشان دهیم، در این صورت

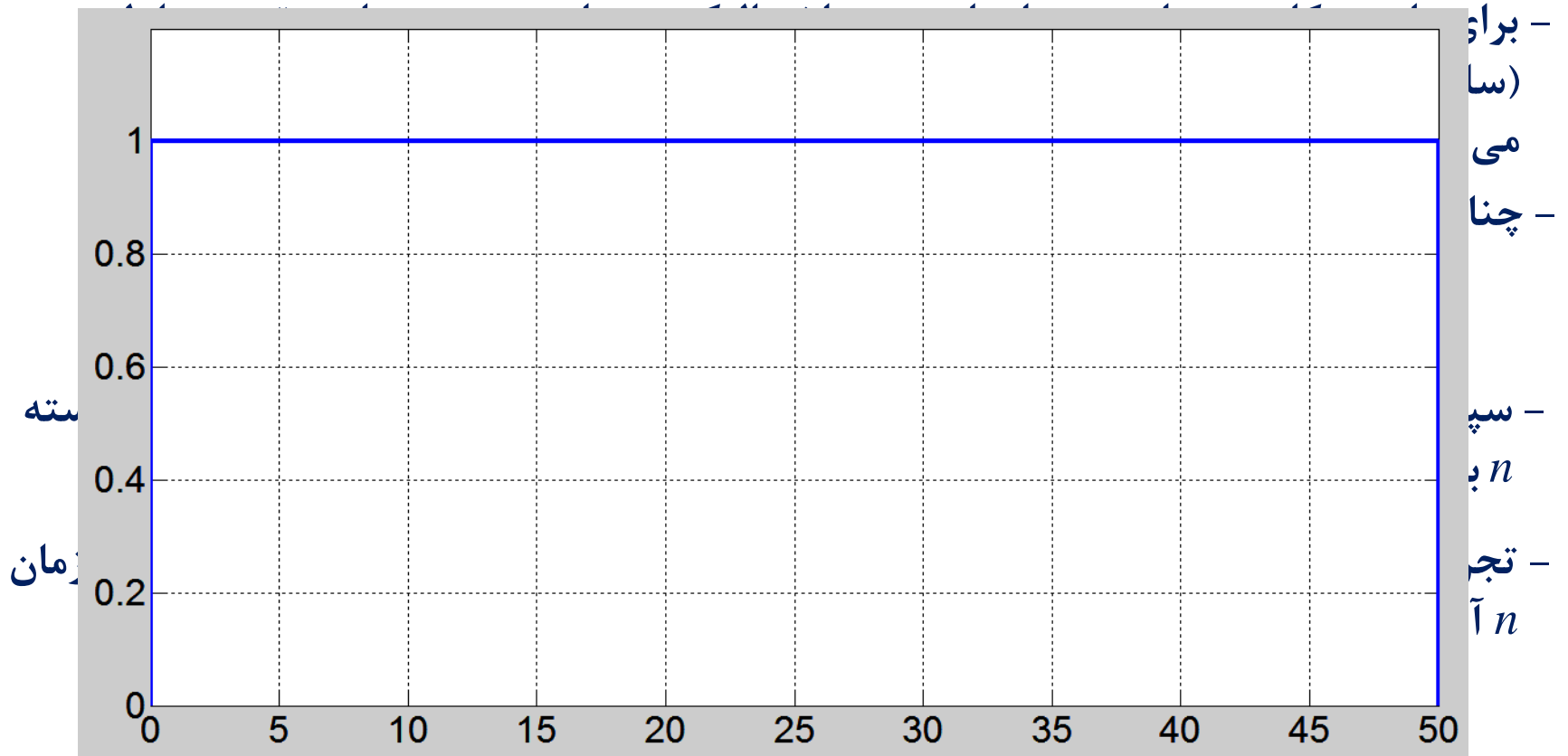
$$i(x) = \min_j \|x - w_j\|, \quad j = 1, \dots, N$$

- سپس باید محدوده همسایگی سلول برنده را انتخاب کنیم که می تواند تابعی از زمان گسسته n باشد و آن را با $\Lambda_{i(x)}(n)$ نشان می دهیم (تابع همسایگی).

- تجربه نشان داده است که بهتر است تابع همسایگی را ابتدا بزرگ انتخاب کرده و در طول زمان n آن را کوچک کنیم.

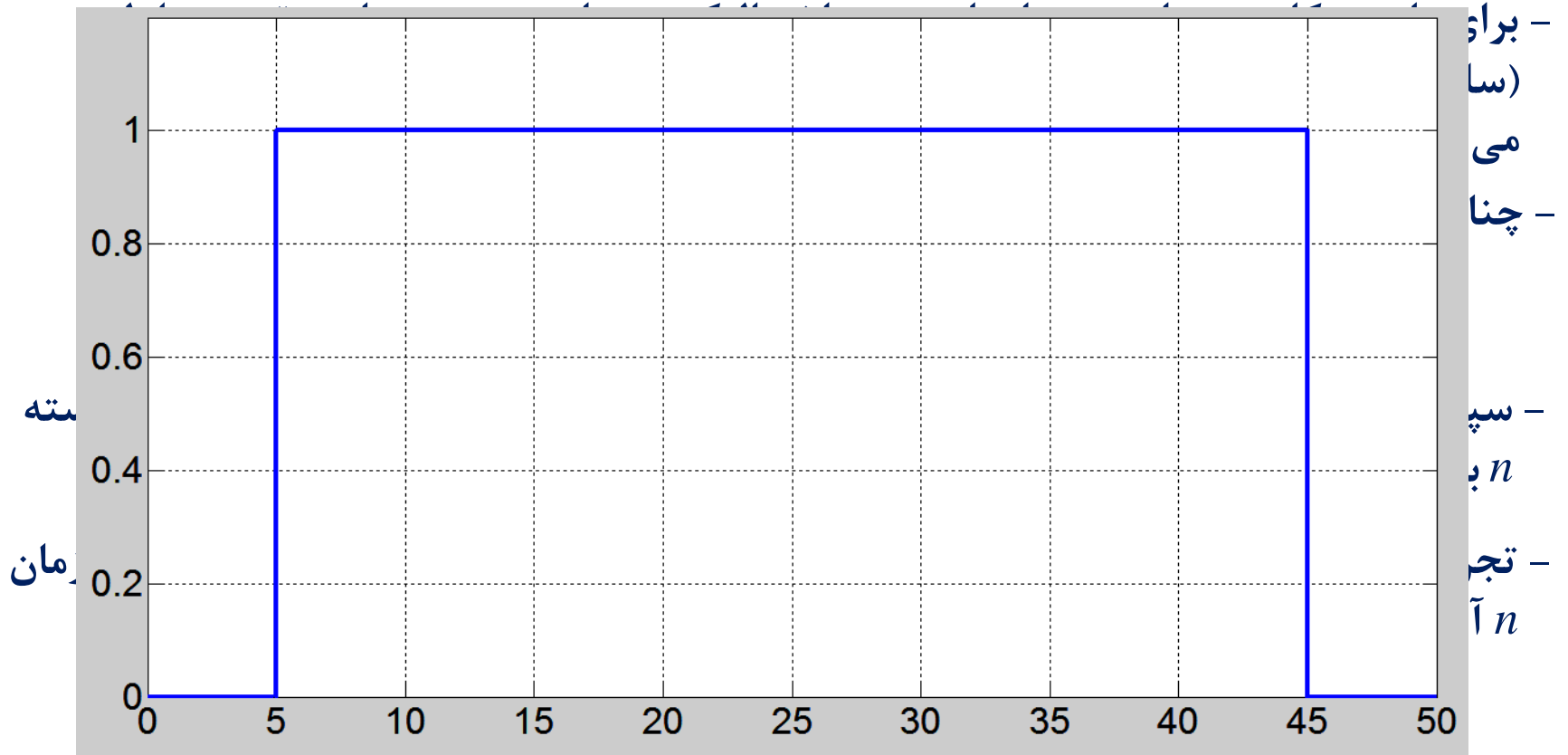
نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



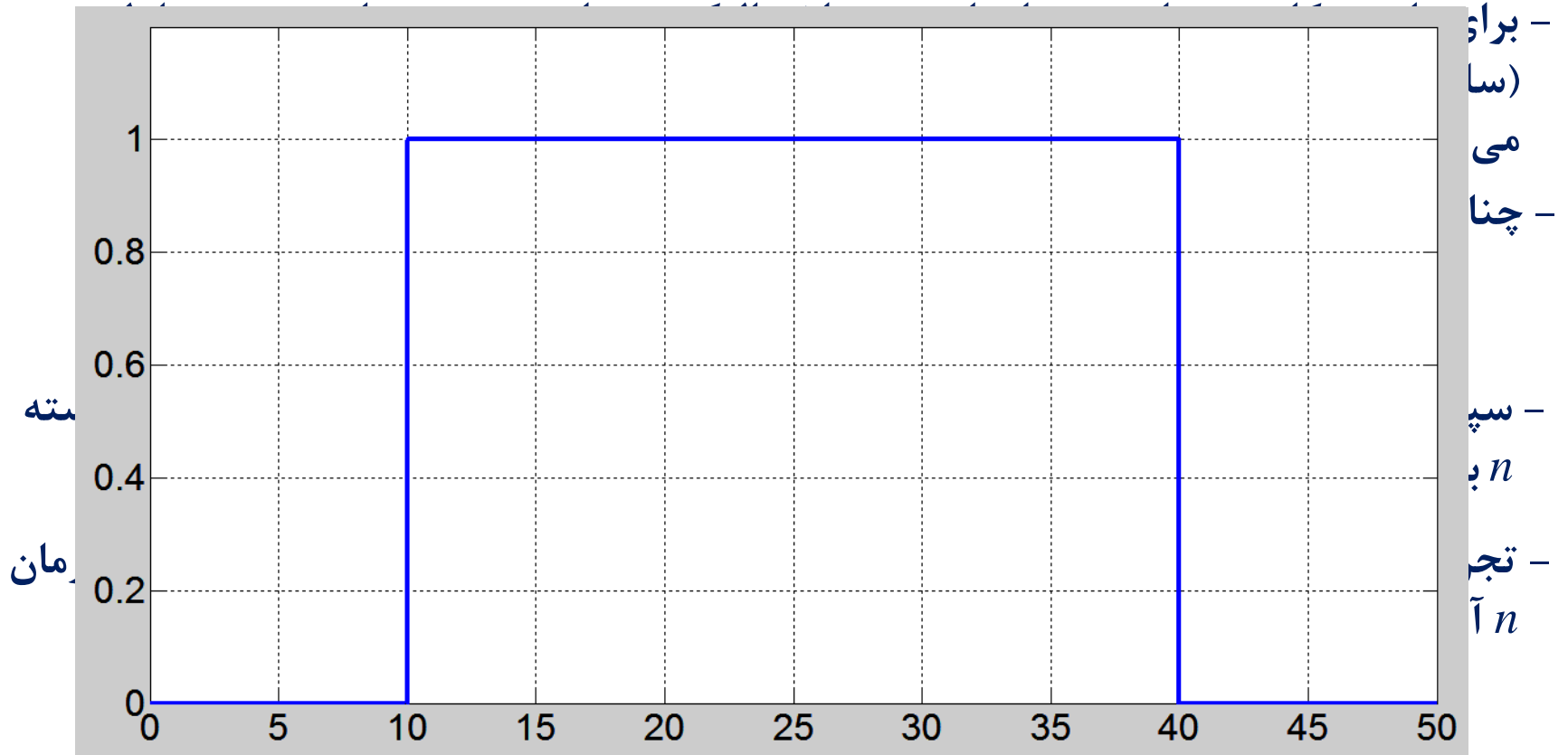
نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



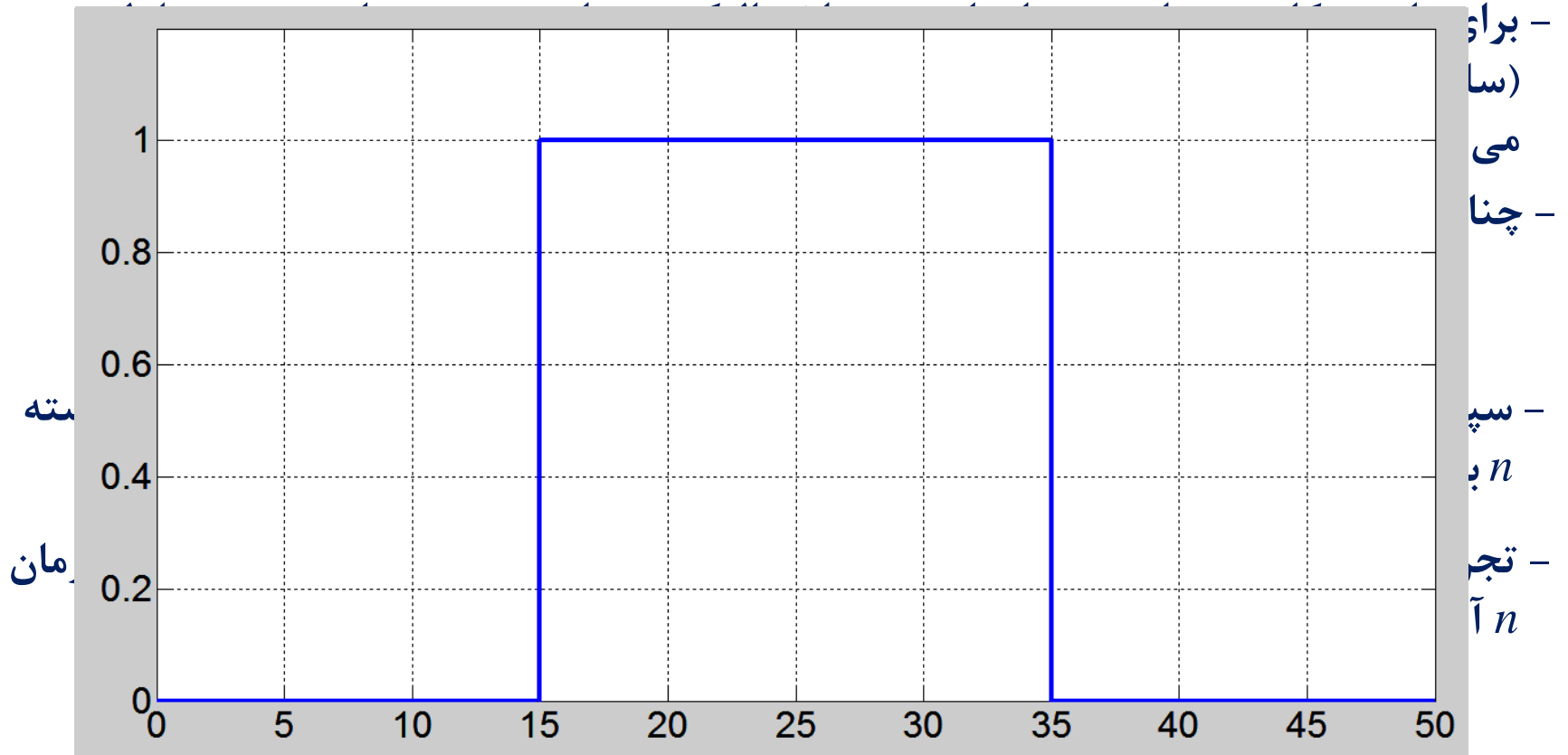
نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



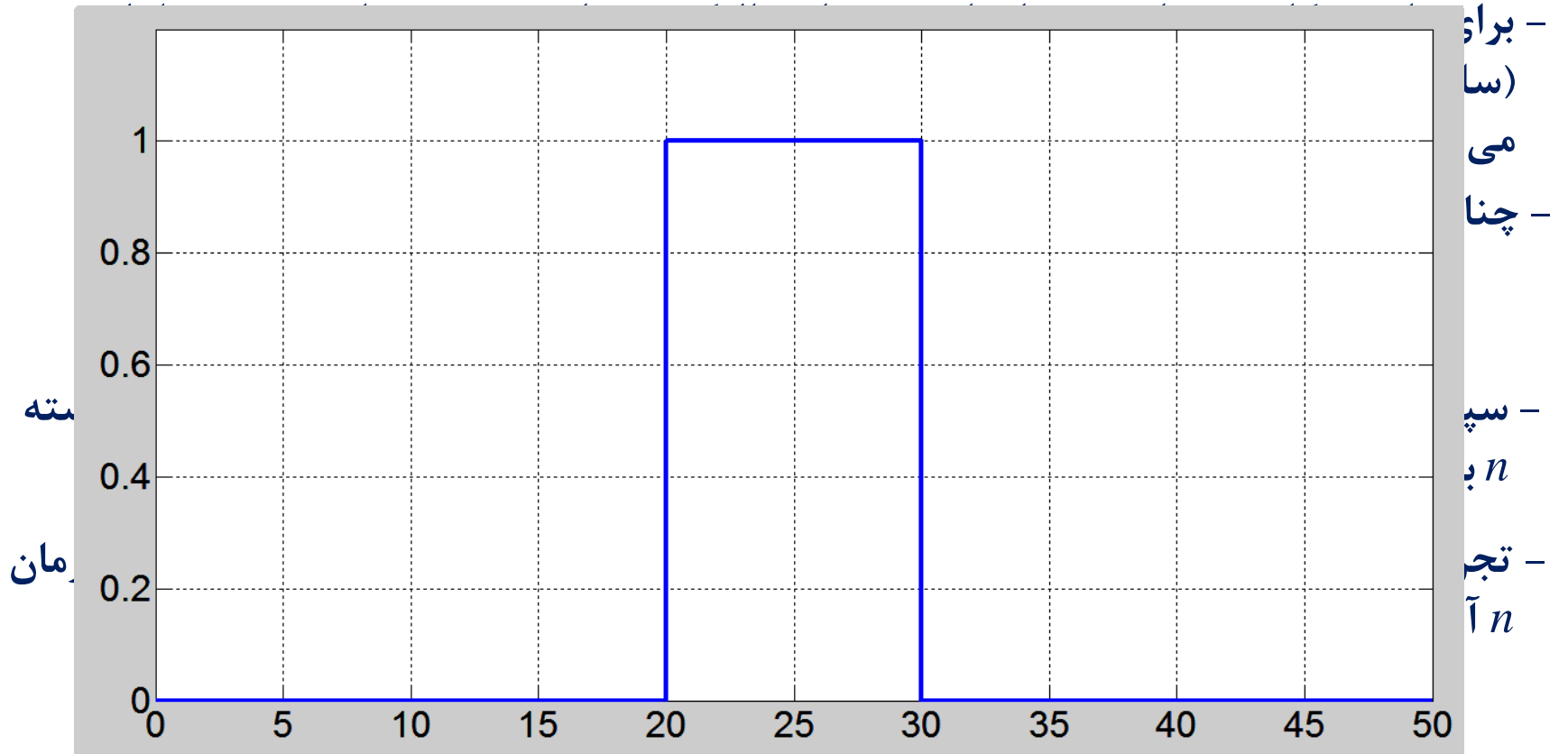
نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:



نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (w_j) و بردار ورودی (x) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(x)$ نشان دهیم، در این صورت

$$i(x) = \min_j \|x - w_j\|, \quad j = 1, \dots, N$$

- سپس باید محدوده همسایگی سلول برنده را انتخاب کنیم که می تواند تابعی از زمان گسسته n باشد و آن را با $\Lambda_{i(x)}(n)$ نشان می دهیم (تابع همسایگی).

- تجربه نشان داده است که بهتر است تابع همسایگی را ابتدا بزرگ انتخاب کرده و در طول زمان n آن را کوچک کنیم.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (w_j) و بردار ورودی (x) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(x)$ نشان دهیم، در این صورت

$$i(x) = \min_j \|x - w_j\|, \quad j = 1, \dots, N$$

- سپس باید محدوده همسایگی سلول برنده را انتخاب کنیم که می تواند تابعی از زمان گسسته n باشد و آن را با $\Lambda_{i(x)}(n)$ نشان می دهیم (تابع همسایگی).

- تجربه نشان داده است که بهتر است تابع همسایگی را ابتدا بزرگ انتخاب کرده و در طول زمان n آن را کوچک کنیم.

- به عبارت دیگر، در ابتدا پسخوردهای جانبی مثبت (تحریک کننده) را قوی انتخاب کنیم و سپس پسخوردهای جانبی منفی (بازدارنده) را تقویت کنیم.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

- برای راحتی کار، بهتر است بردارهای وزن را نرمال کرد. در این صورت، مناسب ترین سلول (سلول برنده) دارای کمترین فاصله (اقلیدسی) بین وزن آن سلول (w_j) و بردار ورودی (x) می باشد

- چنانچه اندیس مناسب ترین سلول را با $i(x)$ نشان دهیم، در این صورت

$$i(x) = \min_j \|x - w_j\|, \quad j = 1, \dots, N$$

- سپس باید محدوده همسایگی سلول برنده را انتخاب کنیم که می تواند تابعی از زمان گسسته n باشد و آن را با $\Lambda_{i(x)}(n)$ نشان می دهیم (تابع همسایگی).

- تجربه نشان داده است که بهتر است تابع همسایگی را ابتدا بزرگ انتخاب کرده و در طول زمان n آن را کوچک کنیم.

- به عبارت دیگر، در ابتدا پسخوردهای جانبی مثبت (تحریک کننده) را قوی انتخاب کنیم و سپس پسخوردهای جانبی منفی (بازدارنده) را تقویت کنیم.

- به این ترتیب، با استفاده از تابع همسایگی، همان عمل پسخوردهای جانبی را انجام داده ایم ولی با محاسبات ساده تر.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بعد از این مراحل اولیه الگوریتم، باید رابطه‌ای برای تنظیم وزن‌های شبکه پیدا کرد.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بعد از این مراحل اولیه الگوریتم، باید رابطه‌ای برای تنظیم وزن‌های شبکه پیدا کرد.

– برای این کار از فرم تغییر یافته روش آموزش هب استفاده می‌کنیم:

$$\frac{d\mathbf{w}_j}{dt} = \eta [y_j \mathbf{x} - g(y_j) \mathbf{w}_j], \quad j = 1, \dots, N$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بعد از این مراحل اولیه الگوریتم، باید رابطه‌ای برای تنظیم وزن‌های شبکه پیدا کرد.

– برای این کار از فرم تغییر یافته روش آموزش هب استفاده می‌کنیم:

$$\frac{d\mathbf{w}_j}{dt} = \eta [y_j \mathbf{x} - g(y_j) \mathbf{w}_j], \quad j = 1, \dots, N$$

که در آن

$$y_j = \begin{cases} 1, & \text{neuron } j \text{ is active } (j \in \Lambda_{i(\mathbf{x})}) \\ 0, & \text{neuron } j \text{ is not active } (j \notin \Lambda_{i(\mathbf{x})}) \end{cases}$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بعد از این مراحل اولیه الگوریتم، باید رابطه‌ای برای تنظیم وزن‌های شبکه پیدا کرد.

– برای این کار از فرم تغییر یافته روش آموزش هب استفاده می‌کنیم:

$$\frac{d\mathbf{w}_j}{dt} = \eta [y_j \mathbf{x} - g(y_j) \mathbf{w}_j], \quad j = 1, \dots, N$$

که در آن

$$y_j = \begin{cases} 1, & \text{neuron } j \text{ is active } (j \in \Lambda_{i(\mathbf{x})}) \\ 0, & \text{neuron } j \text{ is not active } (j \notin \Lambda_{i(\mathbf{x})}) \end{cases}$$

– برای سادگی کار می‌توان تابع غیرخطی $g(y_j)$ را برابر با y_j فرض کرد:

$$g(y_j) = y_j$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بنابراین، تغییرات در وزن‌ها

$$\frac{d\mathbf{w}_j}{dt} = \begin{cases} \eta (\mathbf{x} - \mathbf{w}_j), & j \in \Lambda_{i(\mathbf{x})} \\ 0, & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بنابراین، تغییرات در وزن‌ها

$$\frac{d\mathbf{w}_j}{dt} = \begin{cases} \eta(\mathbf{x} - \mathbf{w}_j), & j \in \Lambda_{i(\mathbf{x})} \\ 0, & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

– در این صورت، معادله تنظیم وزن‌ها برابر است با

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بنابراین، تغییرات در وزن‌ها

$$\frac{d\mathbf{w}_j}{dt} = \begin{cases} \eta(\mathbf{x} - \mathbf{w}_j), & j \in \Lambda_{i(\mathbf{x})} \\ 0, & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

– در این صورت، معادله تنظیم وزن‌ها برابر است با

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

– به این ترتیب، تمام وزن‌های \mathbf{w}_j در محدوده همسایگی $\Lambda_{i(\mathbf{x})}(n)$ سعی در نزدیک شدن به بردار ورودی \mathbf{x} را دارند.

نگاشت خودسازمانده (SOM)

الگوریتم نگاشت خودسازمانده:

– بنابراین، تغییرات در وزن ها

$$\frac{d\mathbf{w}_j}{dt} = \begin{cases} \eta(\mathbf{x} - \mathbf{w}_j), & j \in \Lambda_{i(\mathbf{x})} \\ 0, & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

– در این صورت، معادله تنظیم وزن ها برابر است با

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

– به این ترتیب، تمام وزن های \mathbf{w}_j در محدوده همسایگی $\Lambda_{i(\mathbf{x})}(n)$ سعی در نزدیک شدن به بردار ورودی \mathbf{x} را دارند.

– به عبارت دیگر، سلول های مجاور سعی در داشتن بردار وزن های مشابه را دارند.

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j = 1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

- ۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j = 1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.
- ۲- انتخاب یک نمونه x از بردارهای ورودی به طور اتفاقی.

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j=1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

۲- انتخاب یک نمونه \mathbf{x} از بردارهای ورودی به طور اتفاقی.

۳- یافتن مناسب ترین سلول $i(\mathbf{x})$ در زمان n با استفاده از

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x}(n) - \mathbf{w}_j(n)\|, \quad j = 1, \dots, N$$

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j=1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

۲- انتخاب یک نمونه \mathbf{x} از بردارهای ورودی به طور اتفاقی.

۳- یافتن مناسب ترین سلول $i(\mathbf{x})$ در زمان n با استفاده از

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x}(n) - \mathbf{w}_j(n)\|, \quad j = 1, \dots, N$$

و تعیین تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j=1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

۲- انتخاب یک نمونه \mathbf{x} از بردارهای ورودی به طور اتفاقی.

۳- یافتن مناسب ترین سلول $i(\mathbf{x})$ در زمان n با استفاده از

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x}(n) - \mathbf{w}_j(n)\|, \quad j = 1, \dots, N$$

و تعیین تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

۴- تنظیم وزن تمام سلول ها بر طبق

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j=1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

۲- انتخاب یک نمونه \mathbf{x} از بردارهای ورودی به طور اتفاقی.

۳- یافتن مناسب ترین سلول $i(\mathbf{x})$ در زمان n با استفاده از

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x}(n) - \mathbf{w}_j(n)\|, \quad j = 1, \dots, N$$

و تعیین تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

۴- تنظیم وزن تمام سلول ها بر طبق

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

برای گرفتن نتایج مناسب، باید ضریب آموزش $\eta(n)$ و تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$ را در طول آموزش شبکه تغییر داد.

نگاشت خودسازمانده (SOM)

جمع بندی الگوریتم SOM:

۱- انتخاب اعداد اتفاقی و کوچک برای مقادیر اولیه وزن ها $w_j(0)$. تنها محدودیت در این جا این است که مقادیر $w_j(0)$ برای $j=1, \dots, N$ (تعداد سلول های شبکه) متفاوت باشد.

۲- انتخاب یک نمونه \mathbf{x} از بردارهای ورودی به طور اتفاقی.

۳- یافتن مناسب ترین سلول $i(\mathbf{x})$ در زمان n با استفاده از

$$i(\mathbf{x}) = \arg \min_j \|\mathbf{x}(n) - \mathbf{w}_j(n)\|, \quad j = 1, \dots, N$$

و تعیین تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

۴- تنظیم وزن تمام سلول ها بر طبق

$$\mathbf{w}_j(n+1) = \begin{cases} \mathbf{w}_j(n) + \eta(n)(\mathbf{x}(n) - \mathbf{w}_j(n)), & j \in \Lambda_{i(\mathbf{x})} \\ \mathbf{w}_j(n), & j \notin \Lambda_{i(\mathbf{x})} \end{cases}$$

برای گرفتن نتایج مناسب، باید ضریب آموزش $\eta(n)$ و تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$ را در طول آموزش شبکه تغییر داد.

۵- گام های ۲ تا ۴ را تکرار کرده تا تغییر محسوسی در نگاشت ویژگی ها حاصل نشود.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱– فاز مرتب شدن (Ordering Phase):

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱– فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱– فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

۲– فاز همگرا شدن (Convergence Phase):

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱– فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

۲– فاز همگرا شدن (Convergence Phase):

که در آن، ضمن همگرا شدن وزن ها، تنظیم دقیق محدوده هر حباب فعال تعیین می شود.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

– این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱– فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

۲– فاز همگرا شدن (Convergence Phase):

که در آن، ضمن همگرا شدن وزن ها، تنظیم دقیق محدوده هر حباب فعال تعیین می شود.

a– ضریب آموزش $\eta(n)$

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

- این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱- فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

۲- فاز همگرا شدن (Convergence Phase):

که در آن، ضمن همگرا شدن وزن ها، تنظیم دقیق محدوده هر حباب فعال تعیین می شود.

a- ضریب آموزش $\eta(n)$

۱- فاز مرتب شدن: که در آن، برای مدتی (مثلا ۱۰۰۰ بار) $\eta(n) = 1$ و سپس، به طور آهسته کاهش یافته (به صورت خطی، نمایی یا به نسبت وارون n) و در نهایت ثابت نگه داشته می شود (مثلا حدود ۰٫۱ یا بیشتر).

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

- این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱- فاز مرتب شدن (Ordering Phase):

که در آن، مراکز و حدود تقریبی حباب های فعال تعیین می شود.

۲- فاز همگرا شدن (Convergence Phase):

که در آن، ضمن همگرا شدن وزن ها، تنظیم دقیق محدوده هر حباب فعال تعیین می شود.

a- ضریب آموزش $\eta(n)$

۱- فاز مرتب شدن: که در آن، برای مدتی (مثلا ۱۰۰۰ بار) $\eta(n) = 1$ و سپس، به طور آهسته کاهش یافته (به صورت خطی، نمایی یا به نسبت وارون n) و در نهایت ثابت نگه داشته می شود (مثلا حدود ۰٫۱ یا بیشتر).

۲- فاز همگرا شدن:

که در آن، برای مدتی نسبتا طولانی (چندین هزار بار) مقدار $\eta(n)$ به مقادیر نسبتا کوچکتر (مثلا ۰٫۱ یا کمتر) کاهش یافته تا همگرایی در شبکه حاصل شود.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

- این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱- فاز مرتبه
که در آن

۲- فاز همگرا
که در آن

a- ضریب آموزش

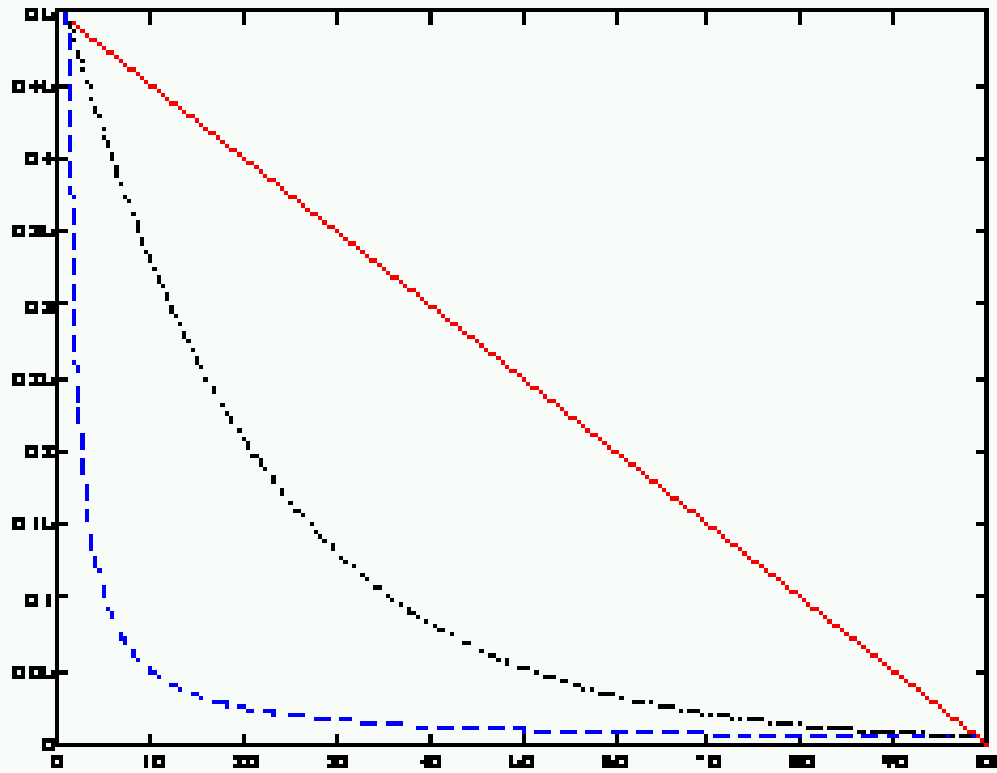
۱- فاز مرتبه
کاهش یافت
(مثلا حدود

۲- فاز همگرا
که در آن،
کوچکتر)

بال تعیین می شود.

س، به طور آهسته
نگه داشته می شود

فادیر نسبتا
د.



نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

- این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱- فاز مرتبش

که در آن

۲- فاز همگراش

که در آن

a- ضریب آموزش

۱- فاز مرتبش

کاهش یافته

(مثلا حدود ۱

۲- فاز همگراش

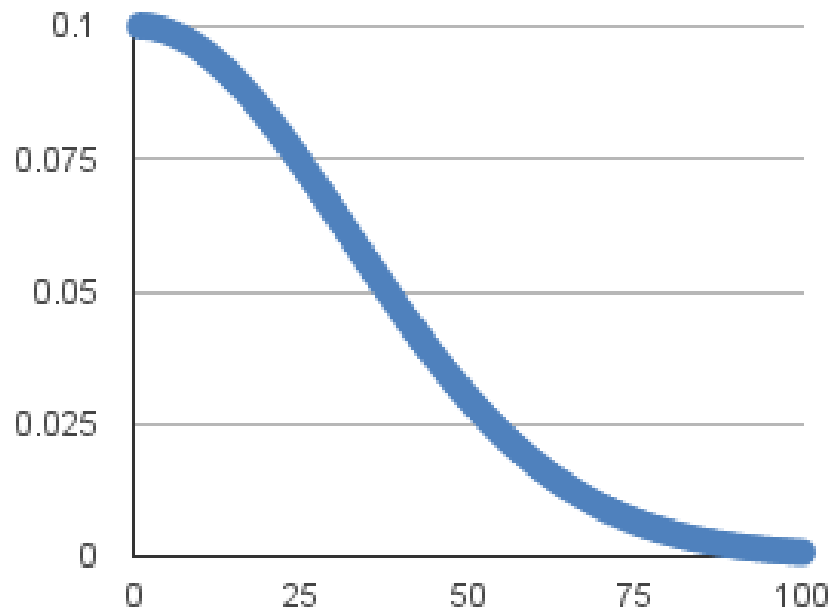
که در آن،

کوچکتر (م

ال تعیین می شود.

ن، به طور آهسته
نگه داشته می شود

ادیر نسبتا



نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

- این پارامترها را می توان در دو فاز زیر تنظیم کرد:

۱- فاز مرتب‌ش

که در آن

۲- فاز همگراش

که در آن

a- ضریب آموزش

۱- فاز مرتب‌ش

کاهش یافته

(مثلا حدود ۱

۲- فاز همگراش

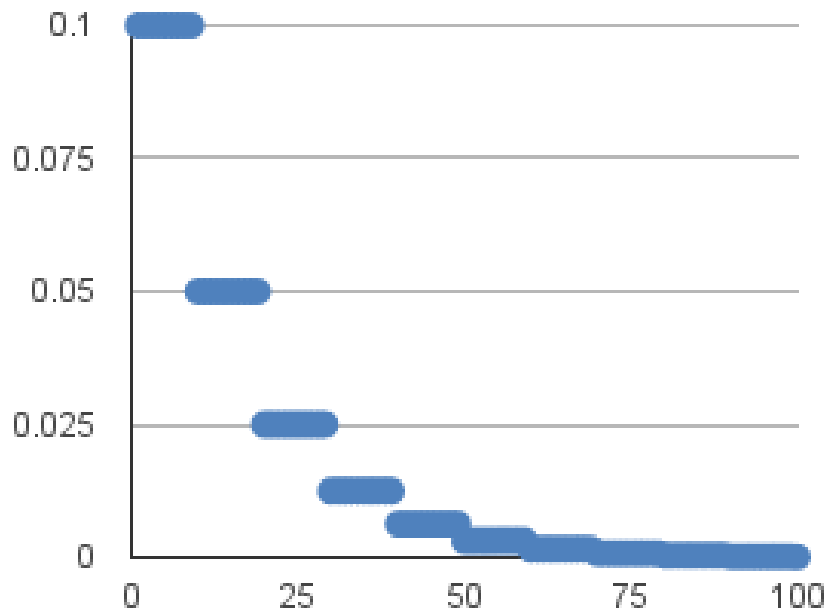
که در آن،

کوچکتر (م

ال تعیین می شود.

ن، به طور آهسته
نگه داشته می شود

ادیر نسبتا



نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

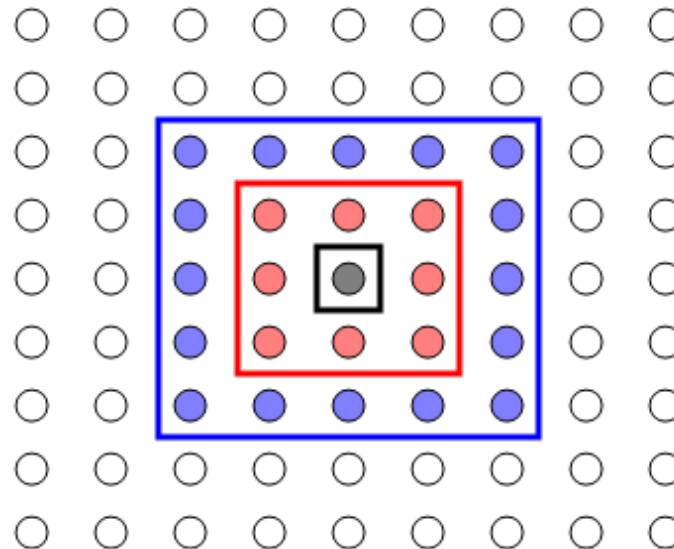
b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- معمولا این محدوده به صورت مربعی در نظر گرفته می شود.



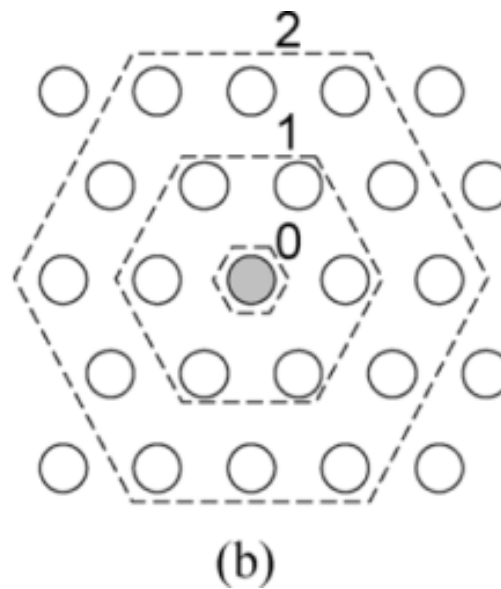
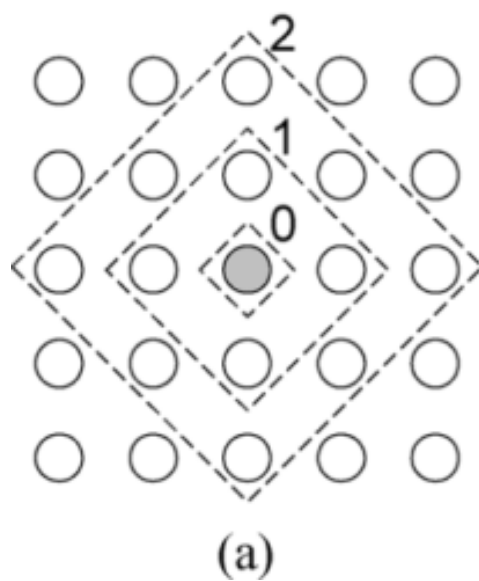
نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- معمولا این محدوده به صورت مربعی در نظر گرفته می شود.

- ولی می توان سایر فرم ها نیز مانند لوزی و شش ضلعی

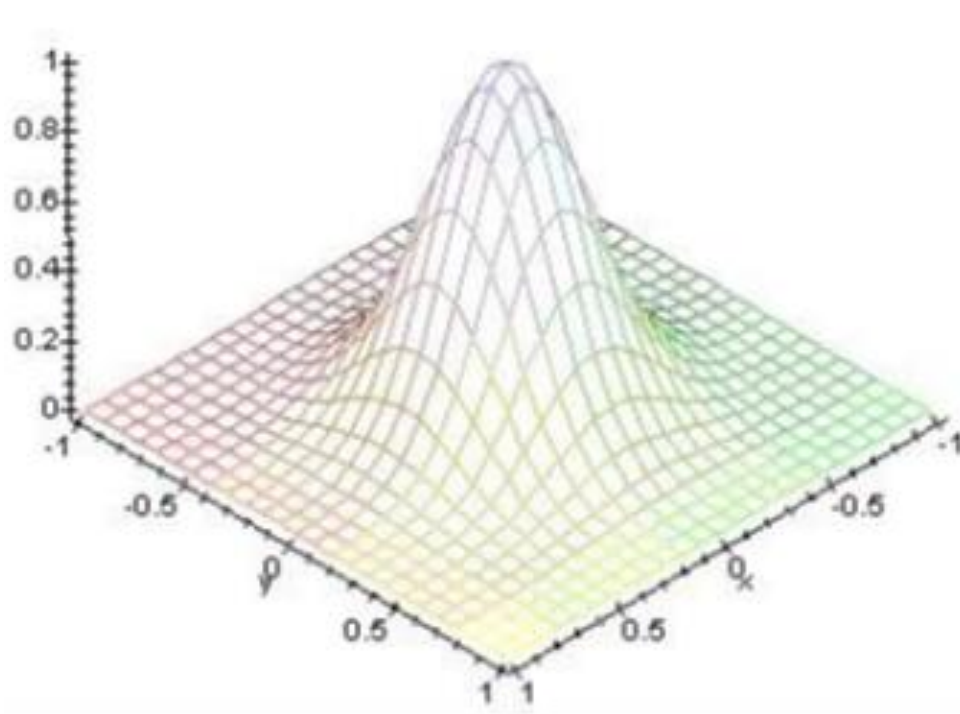


نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- معمولا این محدوده به صورت مربعی در نظر گرفته می شود.
- ولی می توان سایر فرم ها نیز مانند لوزی و شش ضلعی

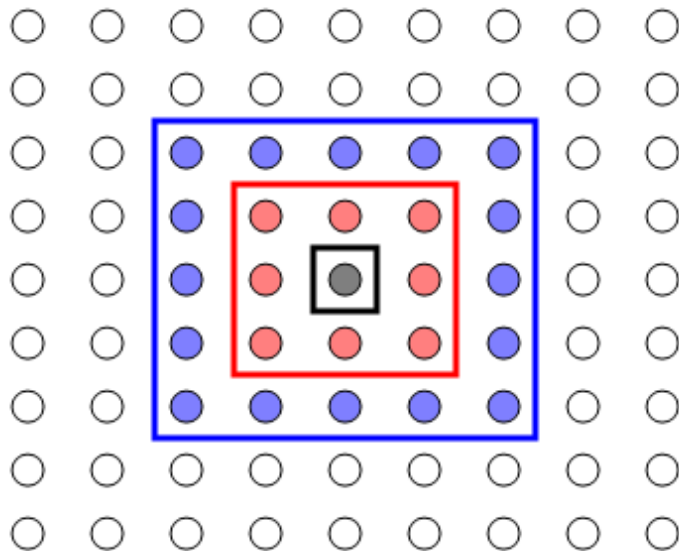


نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت مربعی،



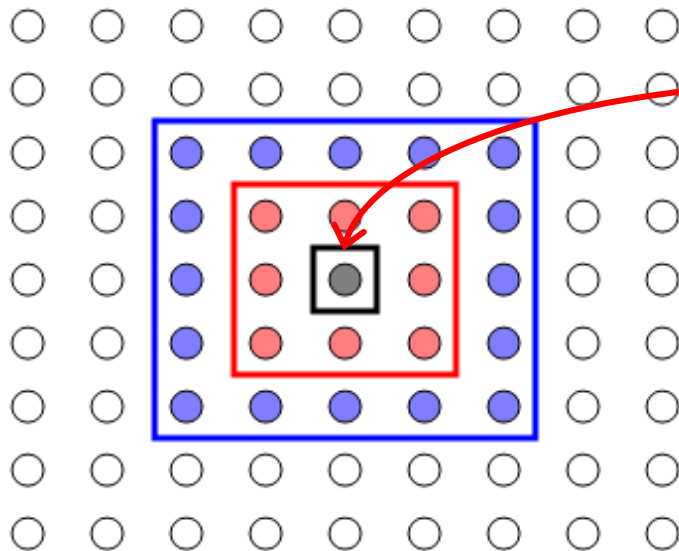
نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت مربعی،

$\Lambda_i(n) = 0$ یعنی فقط سلول برنده

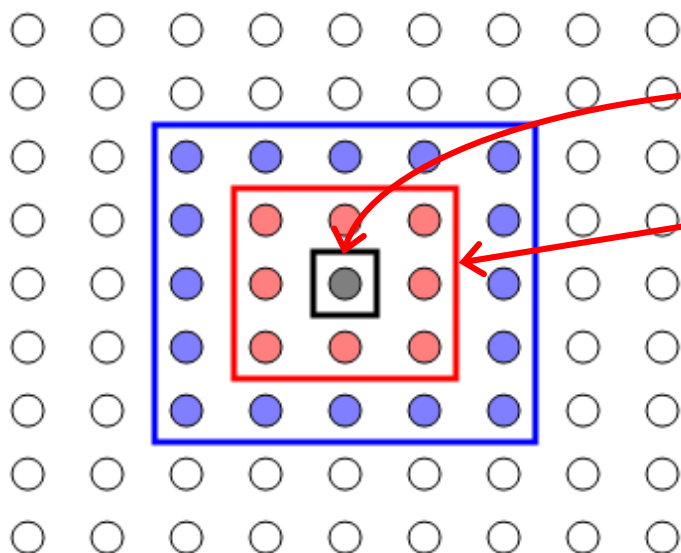


نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت مربعی،



$\Lambda_i(n) = 0$ یعنی فقط سلول برنده

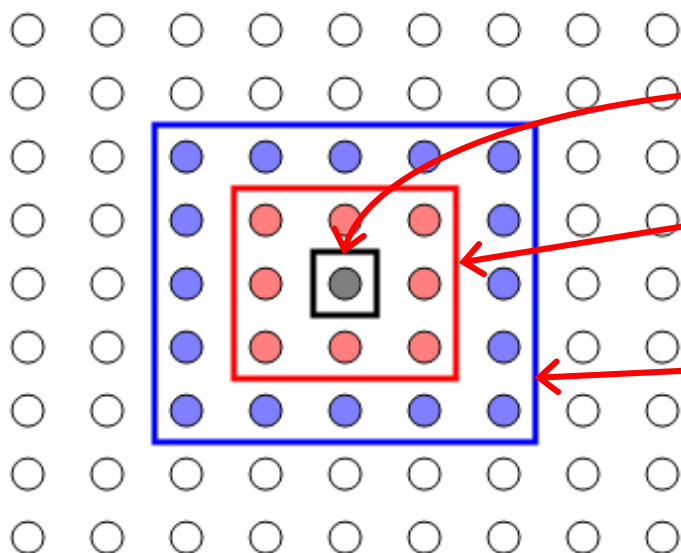
$\Lambda_i(n) = 1$ یعنی سلول برنده و ۸ سلول اطراف آن

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت مربعی،



$\Lambda_i(n) = 0$ یعنی فقط سلول برنده

$\Lambda_i(n) = 1$ یعنی سلول برنده و ۸ سلول اطراف آن

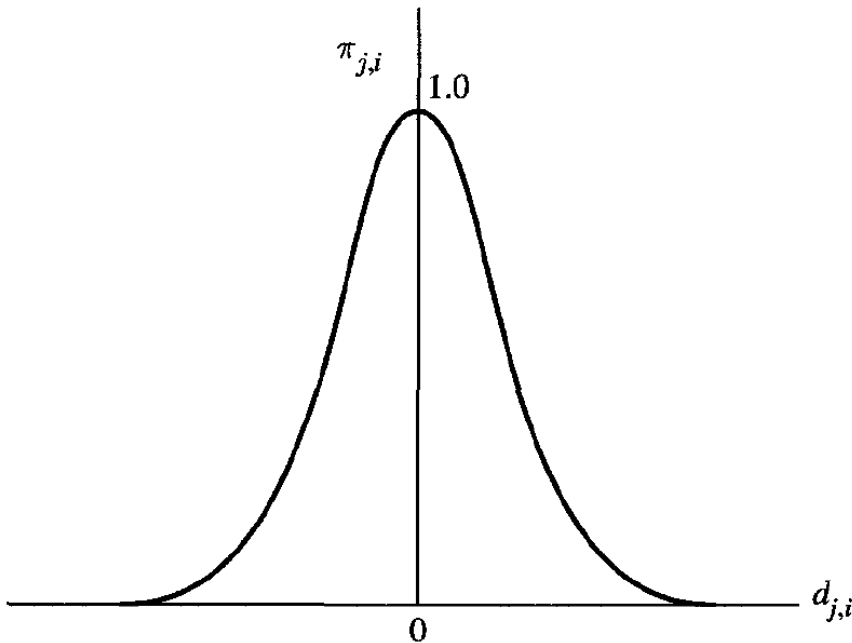
$\Lambda_i(n) = 2$ یعنی سلول برنده و ۲۴ سلول اطراف آن

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت گوسی،



نگاشت خودسازمانده (SOM)

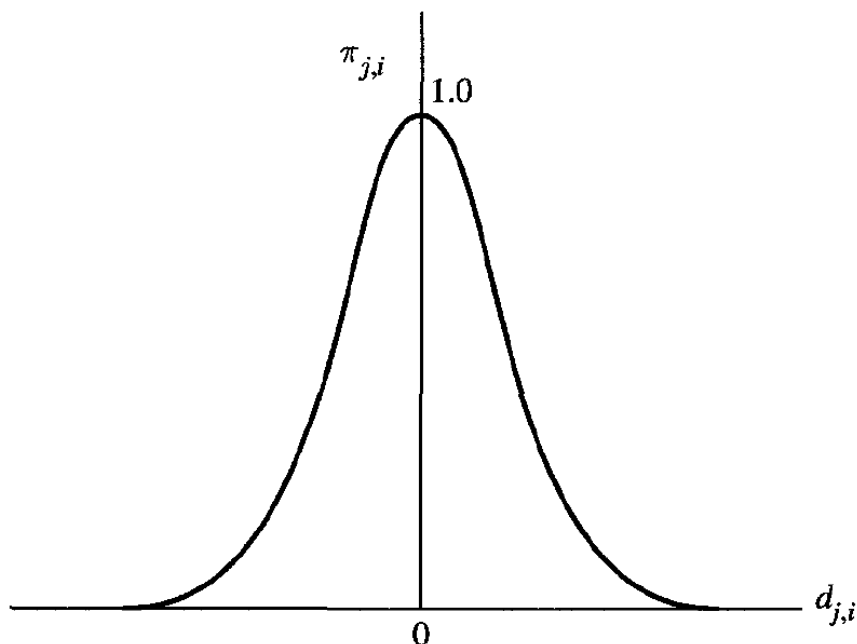
انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت گوسی،

$$\pi_{j,i} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right)$$

$d_{j,i}$ فاصله سلول j از سلول برنده i



نگاشت خودسازمانده (SOM)

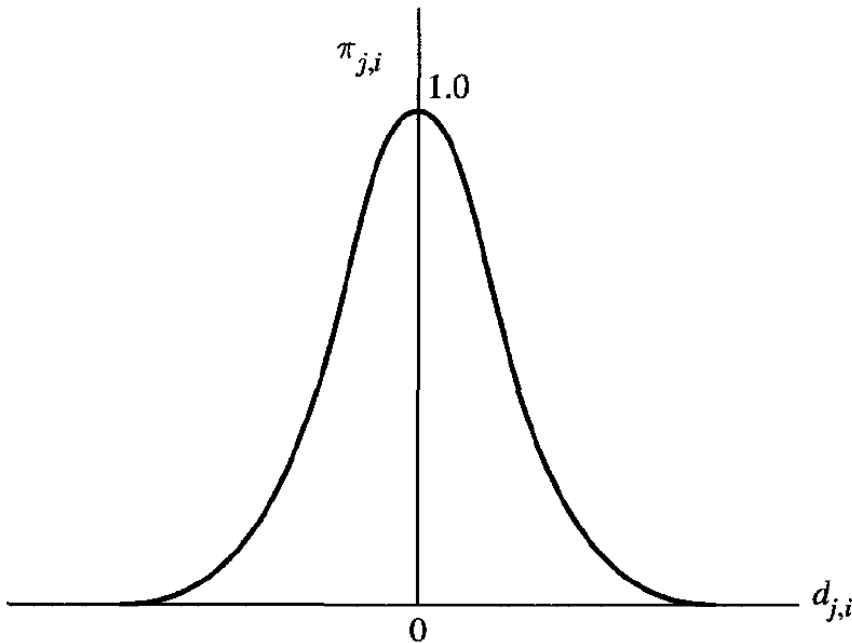
انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(x)}(n)$

- در حالت گوسی،

$$\pi_{j,i} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right)$$

$d_{j,i}$ فاصله سلول j از سلول برنده i



- به عنوان مثال، می توان پهنای این تابع (عرض همسایگی) را در فاز مرتب شدن به صورت زیر تغییر داد:

$$\sigma(n) = \sigma_0 \exp(-n/\tau)^2$$

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

- که در این صورت، تنظیم وزن ها در الگوریتم SOM به صورت زیر انجام می گیرد:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) \pi_{j,i(\mathbf{x})}(n) (\mathbf{x}(n) - \mathbf{w}_j(n))$$

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

- که در این صورت، تنظیم وزن ها در الگوریتم SOM به صورت زیر انجام می گیرد:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) \pi_{j,i(\mathbf{x})}(n) (\mathbf{x}(n) - \mathbf{w}_j(n))$$

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

- که در این صورت، تنظیم وزن ها در الگوریتم SOM به صورت زیر انجام می گیرد:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) \pi_{j,i(\mathbf{x})}(n) (\mathbf{x}(n) - \mathbf{w}_j(n))$$

- در فاز همگراشدن، باید برای مدت نسبتاً طولانی، ضریب آموزش و پهنای تابع همسایگی را تقریباً ثابت نگه داشت.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

- که در این صورت، تنظیم وزن ها در الگوریتم SOM به صورت زیر انجام می گیرد:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) \pi_{j,i(\mathbf{x})}(n) (\mathbf{x}(n) - \mathbf{w}_j(n))$$

- در فاز همگراشدن، باید برای مدت نسبتاً طولانی، ضریب آموزش و پهنای تابع همسایگی را تقریباً ثابت نگه داشت.

- با انتخاب مناسب مقدار اولیه و تغییر مناسب ضریب آموزش و پهنای تابع همسایگی، می توان اطمینان حاصل کرد که هیچ سلولی در شبکه بیهوده نخواهد بود.

نگاشت خودسازمانده (SOM)

انتخاب پارامترهای ضریب آموزش و تابع همسایگی:

b- تابع همسایگی $\Lambda_{i(\mathbf{x})}(n)$

- که در این صورت، تنظیم وزن ها در الگوریتم SOM به صورت زیر انجام می گیرد:

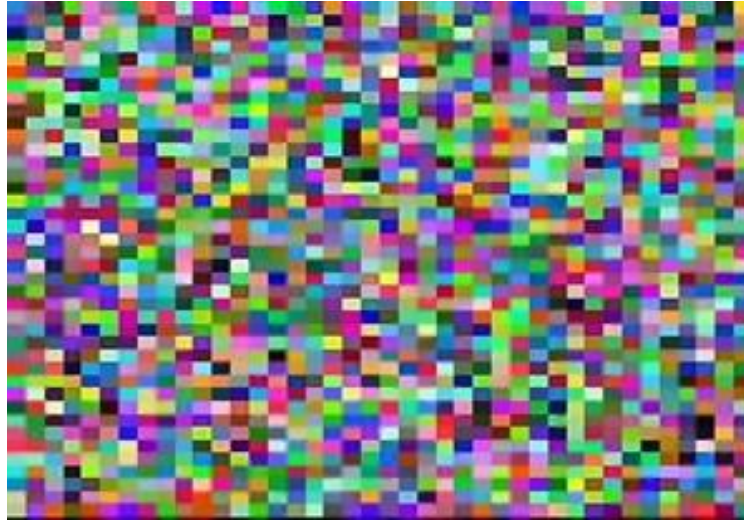
$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) \pi_{j,i(\mathbf{x})}(n) (\mathbf{x}(n) - \mathbf{w}_j(n))$$

- در فاز همگراشدن، باید برای مدت نسبتاً طولانی، ضریب آموزش و پهنای تابع همسایگی را تقریباً ثابت نگه داشت.

- با انتخاب مناسب مقدار اولیه و تغییر مناسب ضریب آموزش و پهنای تابع همسایگی، می توان اطمینان حاصل کرد که هیچ سلولی در شبکه بیهوده نخواهد بود.

نکته قابل توجه در الگوریتم SOM: آموزش اضافی در این الگوریتم، هیچ مشکلی به وجود نمی آورد (برخلاف مثلاً الگوریتم پس-انتشار خطا در MLP)

نگاشت خودسازمانده (SOM)



نگاشت خودسازمانده (SOM)

