

SOM²

شبکه‌ی عصبی مصنوعی جلسه‌ی چهاردهم



دانشگاه شهید بهشتی

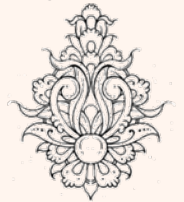
دانشکده‌ی مهندسی برق و کامپیوتر

پاییز ۱۳۹۱

احمد محمودی ازناوه

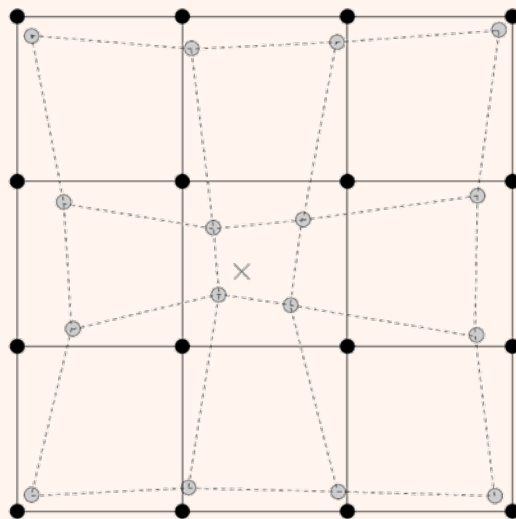
فهرست مطالب

- مروری بر جلسه‌ی پیش



SOM (SOFM)

- به نوعی می‌توان گفت تقلید بهتر از شبکه‌ی عصبی انسان، منجر به ارائه‌ی SOM توسط kohonen شد.
- در این شبکه علاوه بر این که افزون بر نرون برنده نرون همسایه‌ی آن هم آموزش می‌بینند.



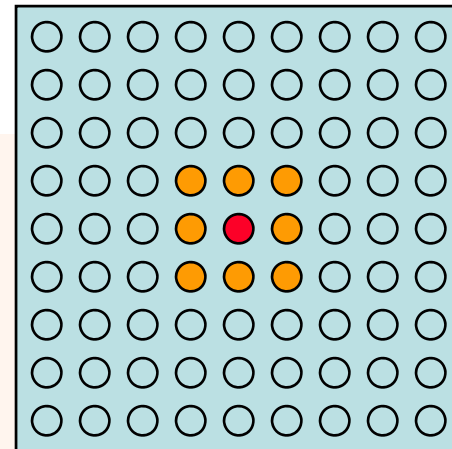
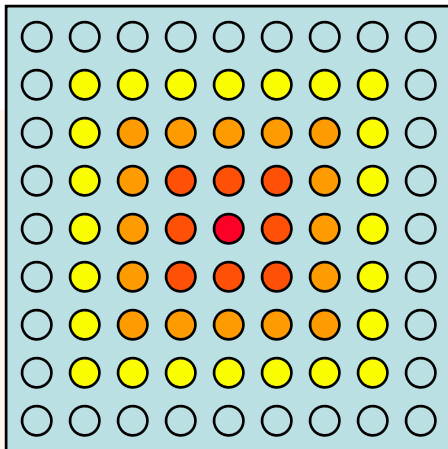
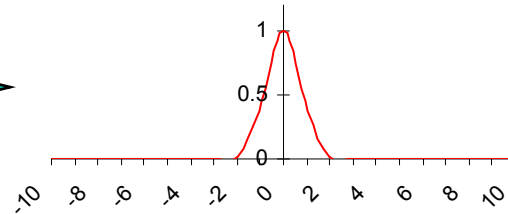
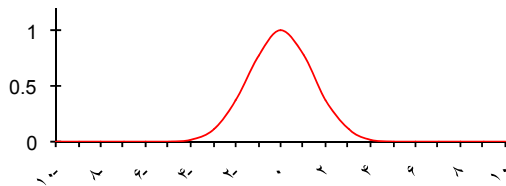
Marc M. Van Hulle



تابع همسایگی

- تابع همسایگی از شعاع همسایگی بزرگ شروع کند و به تدریج تمرکز تنها روی واحد برنده باقی می ماند.

$$h_i(d_{ij}) = \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right)$$



تابع همسایگی

- تعداد همسایگان هم‌روند با زمان قابل تغییر می‌کند.

$$h_i(d_{ij}) = \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right)$$

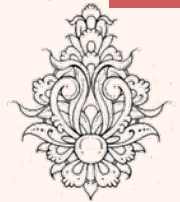
- اگر نرون i برنده باشد d_{ij} همان فاصله‌ی نرون j از نرون برنده (i) است.

- انحراف معیار تابع گاوسی را متغیر در نظر

می‌گیریم:

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) \quad n=0,1,2,\dots$$

$$h_{j,i(x)}(n) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2(n)}\right) \quad n=0,1,2,\dots$$



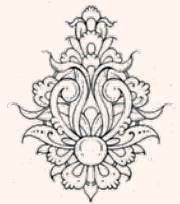
به روزسانی وزن‌ها

- بر طبق روابط به دست آمده خواهیم داشت:

$$w_j(n+1) = w_j(n) + \eta(n) h_{ij(x)}(n) (x - w_j(n))$$

- نرخ یادگیری را نیز به صورت نمایی متغیر در نظر گرفته می‌شود.

$$\eta(n) = \eta_0 \exp\left(-\frac{n}{\tau_2}\right)$$



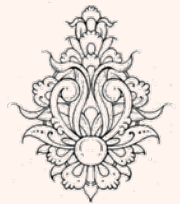
به روزسانی وزن‌ها (ادامه...)

$$w_j(n+1) = w_j(n) + \eta(n) h_{ij(x)}(n) (x - w_j(n))$$

$$h_{j,i(x)}(n) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2(n)}\right) \quad n=0,1,2,\dots$$

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) \quad n=0,1,2,\dots$$

$$\eta(n) = \eta_0 \exp\left(-\frac{n}{\tau_2}\right) \quad n=0,1,2,\dots$$



به روزسانی وزن‌ها (ادامه...)

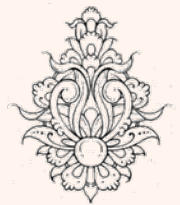
مقادیر η_0 ، σ_0 ، τ_0 و τ_1 به صورت تجربی در بازهای زیر خواهند بود:

$$\eta_0 = 0.1 \quad \tau_2 = 1000$$

این مقدار تدریجی کاهش می‌یابد اما
نهایتاً بالاتر از ۰.۰۱ خواهد بود

- مقداری که برای σ_0 در نظر گرفته می‌شود برابر با شعاع مشبک است.
- بنابراین این مقدار وابسته به چیدمان نرون هاست.
- برای τ_1 مقدار زیر را در نظر می‌گیرند:

$$\tau_1 = \frac{1000}{\log \sigma_0}$$



فازهای یادگیری

- در SOM هر بردار که برنده باشد وزنهای خود و همسایگانش به نسبت همسایگی، تنظیم می‌شوند.

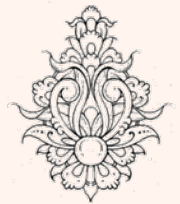
- Ordering

– هدف

- مکان جغرافیایی واحدها مشخص می‌شود.
- شعاع همسایگی به اندازه‌ی تمامی واحدها شروع و سپس رو به کاهش می‌گذارد
- معمولاً تعداد تکرارها ۵۰۰ برابر تعداد واحدهای فروجی است

– تعداد گروه‌ها اهمیت دارد.

Ordering وارد جزئیات نمی‌شود. به‌گونه‌ای فرم
زمن یا **Coarse** را نشان می‌دهد



فازهای یادگیری (ادامه...)

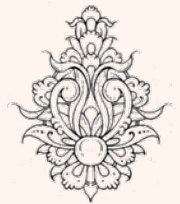
- Convergence(tuning)

– هدف

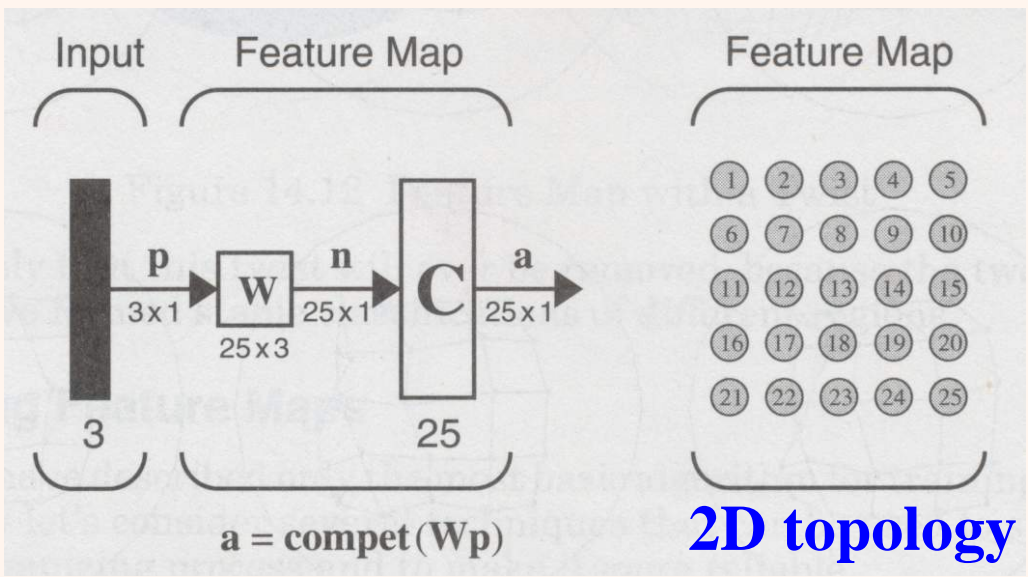
- جزییات تقسیم‌بندی گروه‌ها مشخص می‌شود
- شعاع همسایگی به خود واحد برنده محدود خواهد شد.
- در این حالت جزییات هر واحد و نه همسایه‌ها اهمیت دارد.
- در طی تکرارهای متوالی n کوچک و کوچک‌تر می‌شود.

• نکته:

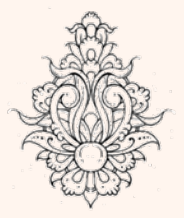
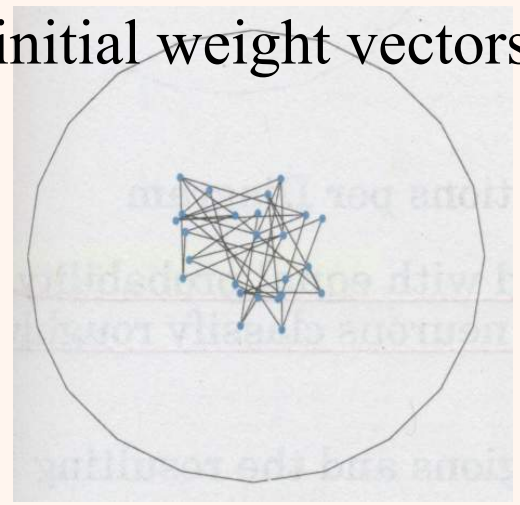
- بردارهای وزن اولیه مقادیر کوچکی انتخاب می‌شوند.
- اگر خیلی پراکنده باشند مساله‌ی واحد مرده پیش می‌آید.



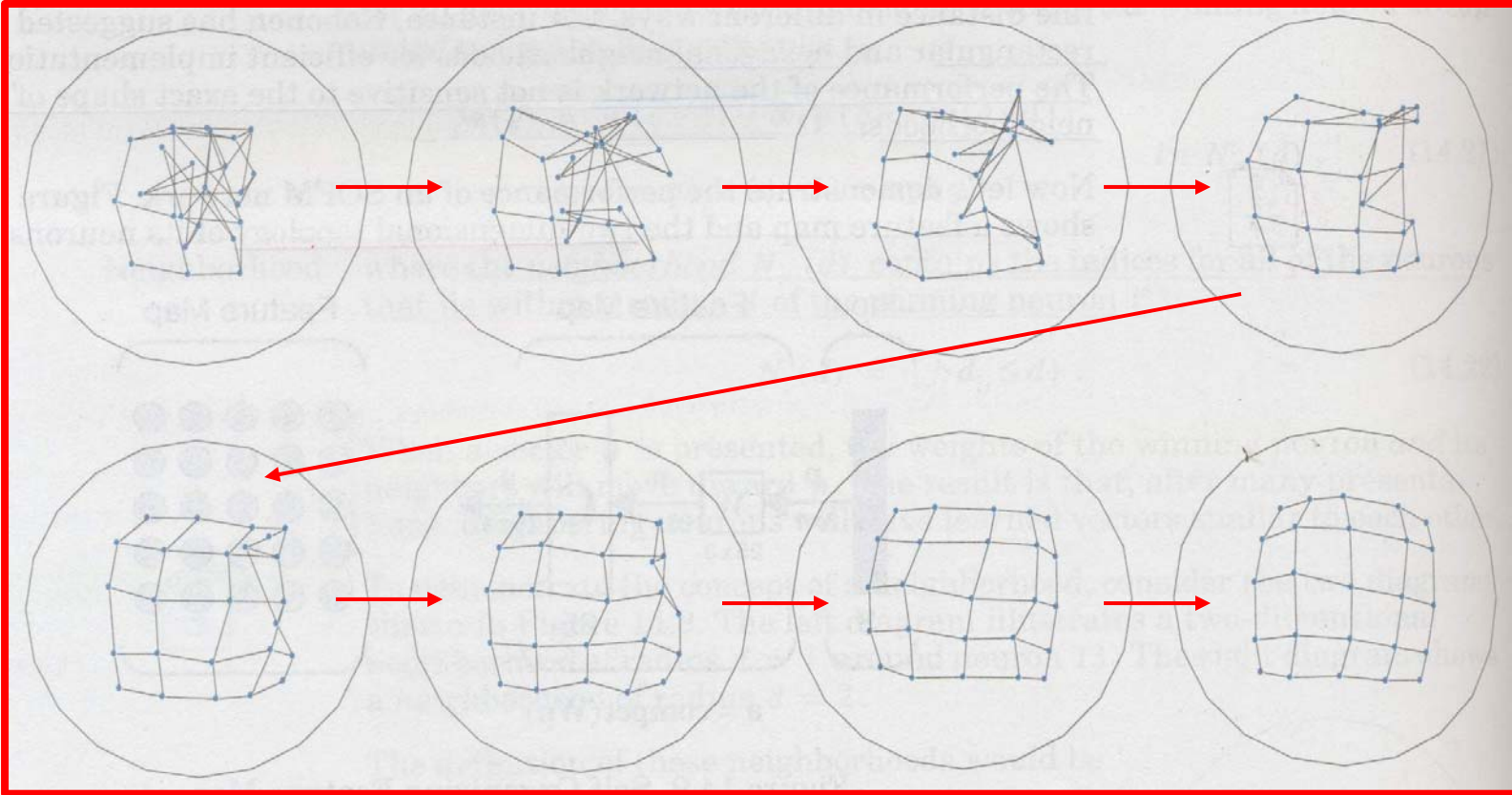
SOM (SOFM)



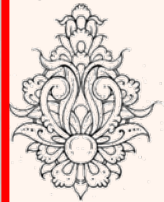
initial weight vectors



SOM (SOFM)



250 iterations per diagram

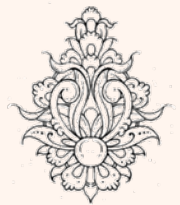


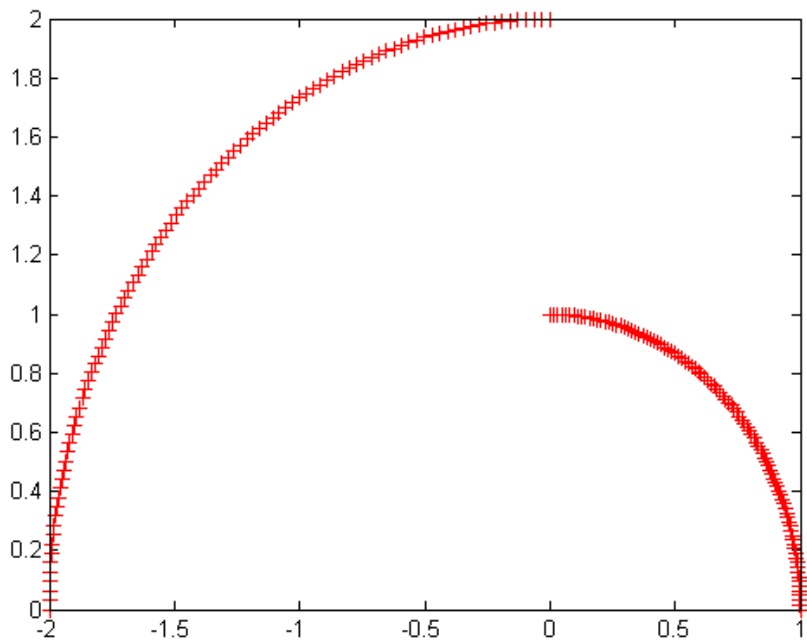
ژانسیکانه
سپهبدی
بهشتی

مثال

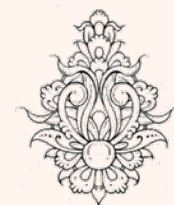
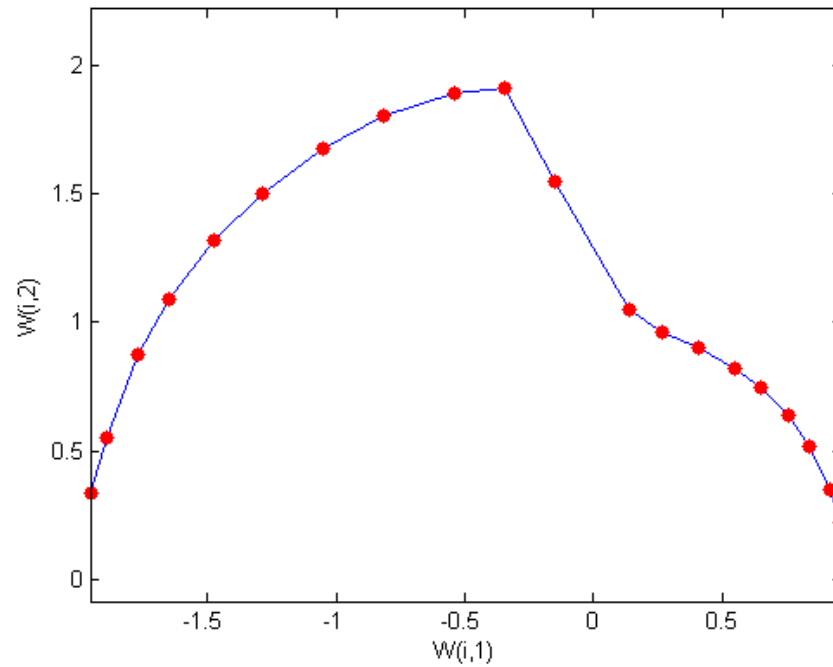
```
angles = 0:0.5*pi/99:0.5*pi;  
P1 = [sin(angles); cos(angles)];  
P2 = [-sin(angles); cos(angles)];  
P=[P1,2.*P2]  
plot(P(1,:),P(2,:),'+r')
```

```
net = newsom([0 1;0 1],[20]);  
net.trainParam.epochs = 10;  
net = train(net,P);  
figure;  
plotsom(net.iw{1,1},net.layers{1}.distances)  
p = [1;0];  
a = sim(net,p)
```



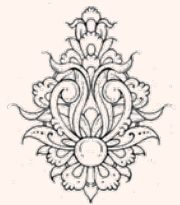
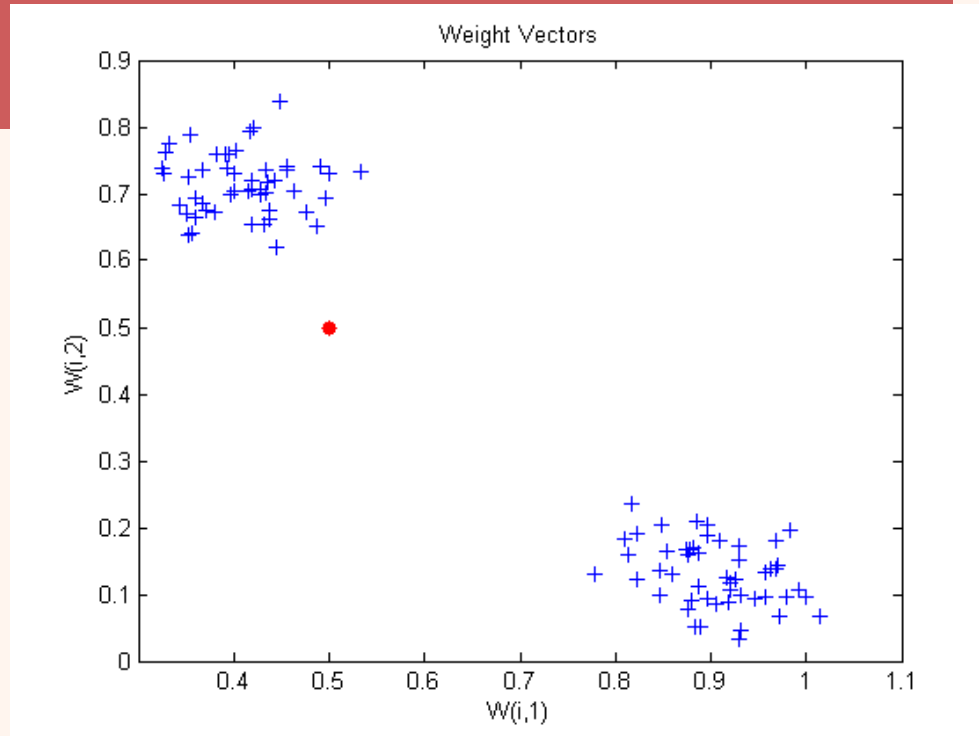


Weight Vectors

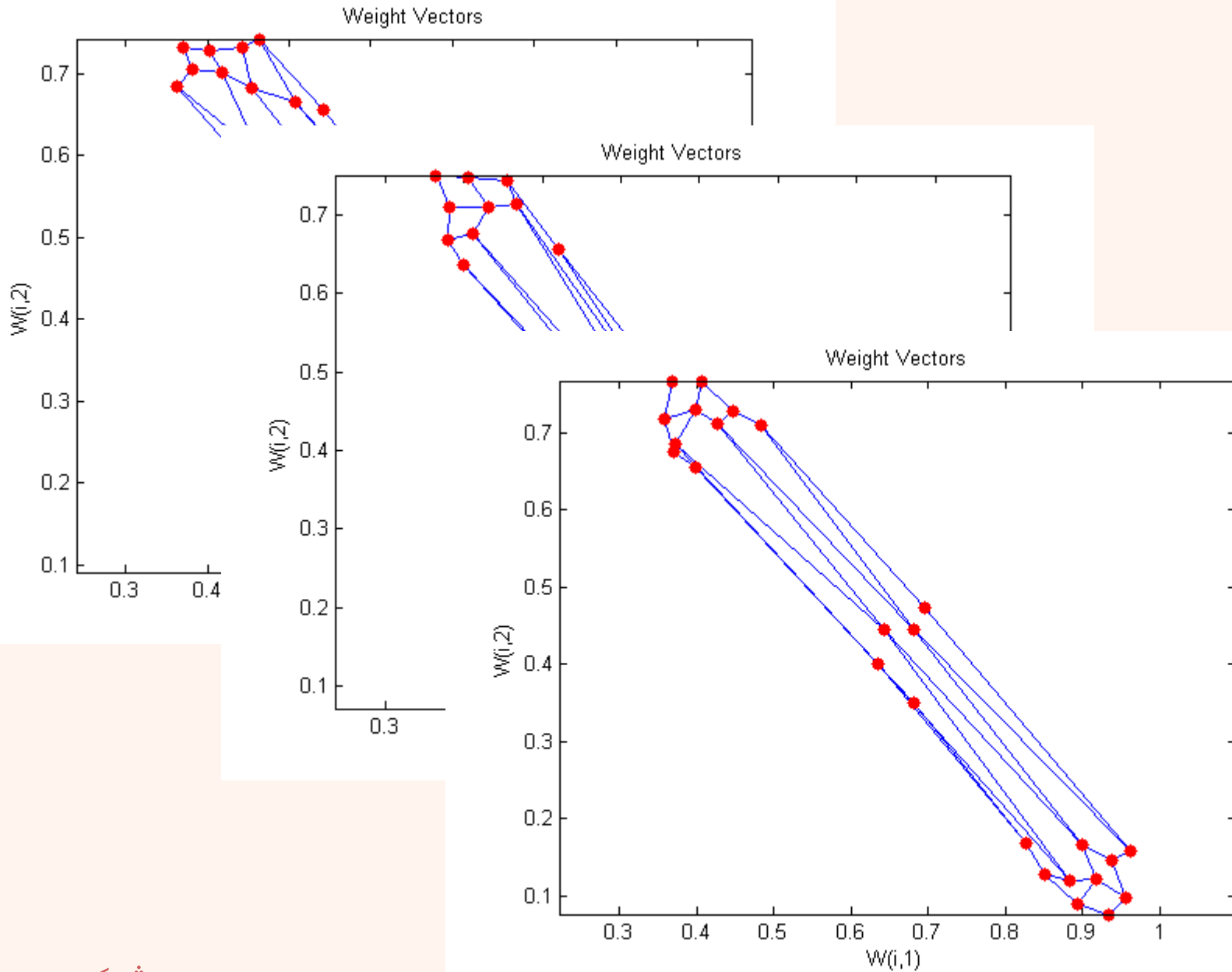


مثال

```
figure;  
plot(P(1,:),P(2,:),'+b')  
net = newsom([0 1; 0 1],[5 5],'gridtop');  
hold on;  
plotsom(net.iw{1,1},net.layers{1}.distances)  
net.trainParam.epochs = 10;  
net = train(net,P);  
figure;  
plotsom(net.iw{1,1},net.layers{1}.distances)  
p = [0.5;0.3];  
a = sim(net,p)
```



ادامه‌ی مثال



مثال

