

不完全是你在找什麼？你可能想嘗試：

- 一個近乎自組織的標籤雲
- Kohonen的C ++自組織映射及其在計算機視覺領域的應用

[突出顯示](#)

13,504,202名成員

1.2K jash.liao

[用品](#) [Q&A](#) [forums](#) [lounge](#)

Self-Organizing Maps

[跟隨](#)

## 自組織特徵映射 (Kohonen 映射)

**Bashir Magomedov**，2006年11月7日

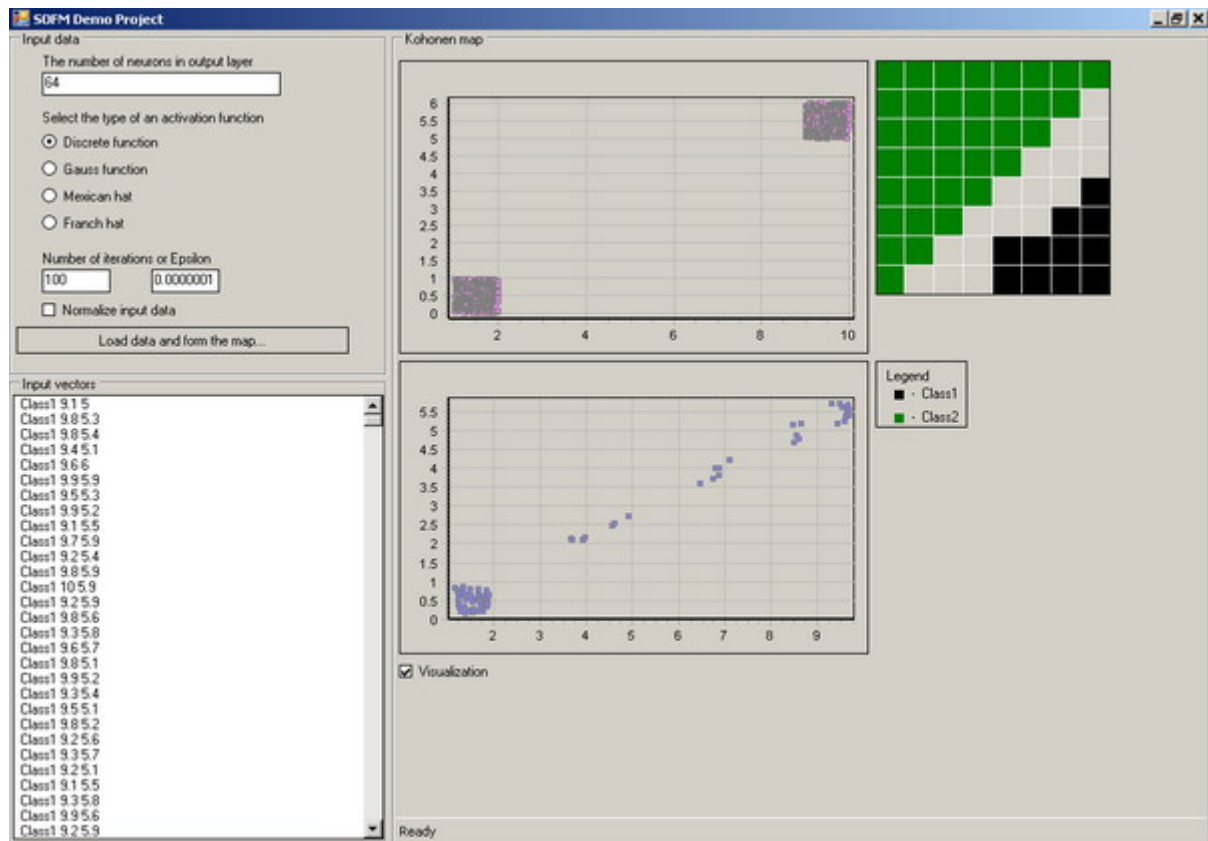
4.73 (44票) 率：

該文章介紹了自組織特徵映射。理論和代碼實現提供。



你的電子郵件地址好嗎？您已註冊我們的新聞通訊，但您的電子郵件地址未經確認，或者長時間未得到確認。請[點擊這裡](#)發送確認郵件，以便我們確認您的電子郵件地址並重新發送您的簡報。或者，您可以 [更新您的訂閱](#)。

[下載源文件 - 12.8 KB](#)[下載演示項目 - 787 KB](#)



## 詞彙表

- **SO (F) M** - 自組織（特徵）地圖
- **(A) NN** - （人工）神經網絡

## 介紹

自組織特徵映射是競爭神經網絡，其中神經元以二維網格（最簡單的情況下）表示特徵空間。根據學習規則，在多維空間中彼此相似的向量在二維空間中將是相似的。**SOFM**通常用於可視化 $n$ 維空間，但其主要應用是數據分類。

## 數據分類

假設我們有一組描述某些對象的 $n$ 維向量（例如汽車）。每個向量元素都是一個對象的參數（例如在汽車的情況下 - 寬度，高度，重量，引擎類型，功率，油箱容積等）。每個這樣的參數對於不同的對象都是不同的。如果您需要通過僅查看這些向量來確定汽車的類型，那麼使用**SOFM**可以輕鬆完成。

## SOFM的體系結構

**SOFM NN**由兩層神經元組成（圖1）。第一層實際上不是神經元層，它只接收輸入數據並將其傳輸到第二層。讓我們考慮最簡單的情況，第二層的神經元組合成二維網格。其他結構，如三維球體，圓柱體等等，都不屬於本文的範圍。第三層的每個神經元與第二層的每個神經元連接。第二層中的神經元數量可以任意選擇，並且隨著任務的不同而不同。第二層的每個神經元都有自己的**權重向量**其維度等於輸入層的維度。神經元通過鄰域關係連接到相鄰的神經元，這決定了地圖的拓撲或結構。這種鄰域關係由稱為**拓撲鄰域**的特殊函數分配（見下文）。

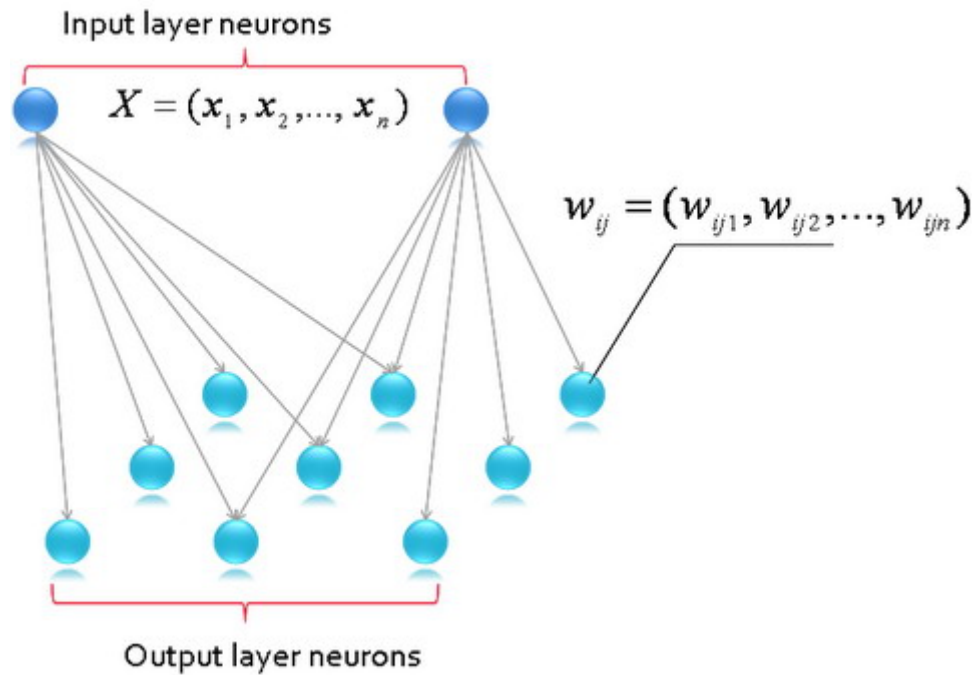


圖1. SOFM NN的體系結構

## 學習規則

在功能開始時，第二層神經元的所有權向量被設置為隨機值。之後，選擇一組學習向量中的一些輸入向量並將其設置為NN的輸入。在這一步，輸入矢量和所有神經元矢量之間的差異計算如下：

$$D_{ij} = \|X^l - W_{ij}\| = \sqrt{(x_1 - w_{ij1})^2 + \dots + (x_n - w_{ijn})^2}$$

其中， $i$ 和 $j$ 是輸出層中神經元的指數。之後，神經網絡選擇**贏家神經元**，即權重向量與輸入向量最相似的神經元：

$$D(k_1, k_2) = \min_{i,j} D_{i,j}$$

這裡， $k_1$ 和 $k_2$ 是贏家神經元的指數。現在，我們需要對獲勝者和所有相鄰神經元的權向量進行修正。神經元的鄰域由拓撲鄰域函數確定，如下所示：

$$h(\rho, t) = \exp\left(\frac{\rho^2}{2\sigma^2(t)}\right)$$

這裡， $r$ 是與贏家神經元的距離：

$$\rho = \sqrt{(k_1 - i)^2 + (k_2 - j)^2}$$

$s$ 是一個決定鄰里空間的功能。在運作開始時，它幾乎涉及電網的整個空間，但隨著時間的推移， $s$ 的值下降。由於吸引函數等於1， $r$ 等於零。這裡顯示的是拓撲鄰域函數的最簡單形式，但在實際任務中，經常使用它的修改：

$$h(\rho, t) = \exp\left(-\frac{\rho^2}{\sigma^2(t)}\right)\left(1 - \frac{2}{\sigma^2(t)}\rho^2\right)$$

- "墨西哥帽子";

$$h(\rho) = \begin{cases} 1, & |\rho| \leq a, \\ -\frac{1}{3}, & a < |\rho| \leq 3a, \\ 0, & |\rho| > 3a, \end{cases}$$

- "法國帽子" -  $a$ 是假定不變的小修飾

在下一步中，在計算每個神經元的拓撲鄰域函數之後，所有神經元的權重更新如下：

$$W_{ij}(t+1) = W_{ij}(t) + \alpha(t)h(\rho, t)(X^l(t) - W_{ij}(t))$$

這裡 ( $t$ ) 是一個隨時間減少的學習率函數。如果神經元是獲勝者或者與獲勝者相鄰，則其權重向量被更新，否則保持不變。在每個步驟中，神經網絡確定其權重向量與輸入向量最相似的神經元，並對其進行修正，並對其和鄰居的權重向量進行修正，使它們更接近輸入向量（圖2）。

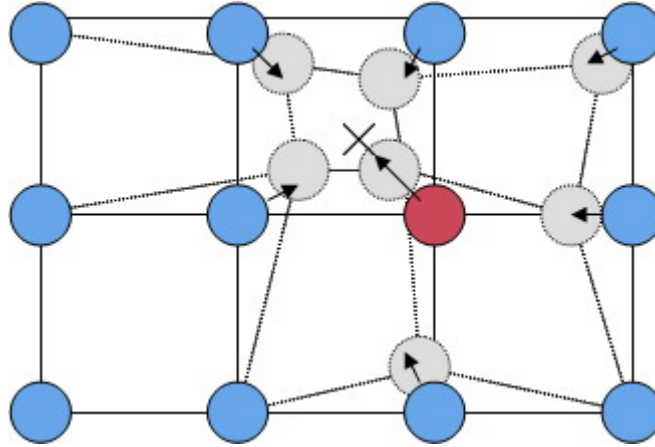


圖2. 將勝者神經元及其鄰居更新為標有X的輸入向量。實線和虛線分別對應於更新之前和之後的情況

通常，將來自訓練集的每個輸入向量呈現給NN，並且學習繼續一些固定數目的周期，或者當輸入和權重向量之間的差異達到某個 $\epsilon$ 值時。相鄰神經元之間的差異隨著時間而減少，因此它們被組織為與學習集中的一個類相對應的組（地圖）。

## 玩演示

在我們開始之前，我需要對演示項目可以處理的數據說幾句話。它必須是表格數據。該表的每一行代表一個對象。行中的項目是數據集的變量。重要的是每個數據樣本都有相同的一組變量。因此，表格的每一列都包含變量的所有值。行中的最後一個變量對應於類名；如果你不知道它，只需添加一個額外的空格字符。所以，讓我們開始吧。

1. 運行演示項目 - **SOFMTest.exe**。
2. 設置第二層中的神經元數量等於100。將迭代次數設置為10。epsilon的值必須小於0.01。選擇離散拓撲鄰域函數並保持“Normalize input data”複選框未被選中。
3. 首先，我們來看看兩個類的例子。按下“加載數據並形成地圖”按鈕並選擇2classtest.data文件。
4. 地圖建築將在進行中。您可以在頂部圖形上看到數據分佈（圖3a），以及在學習過程中修改神經元權重的過程（圖3b）。
5. 學習完成後，您將看到顯示分類過程如何實現的色彩圖（圖4）。

注意：您可以取消選中可視化複選框來加速計算。

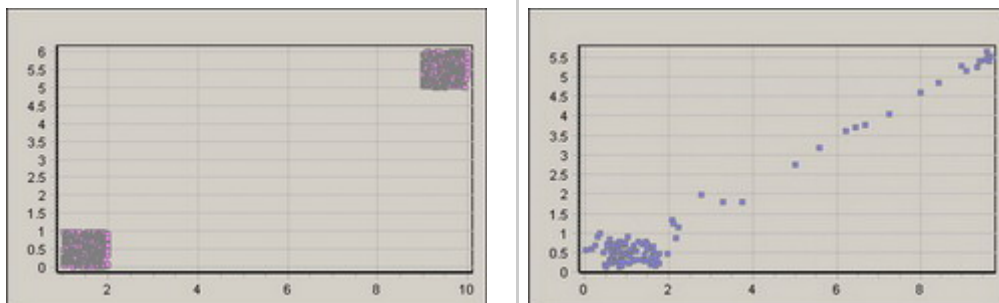


圖3. 兩級示例。數據集中的每個對象由兩個參數組成， $x$ 軸對應於第一個參數，而 $y$ 軸對應於第二個參數。a) 輸入數據分佈。b) 重量變化過程。

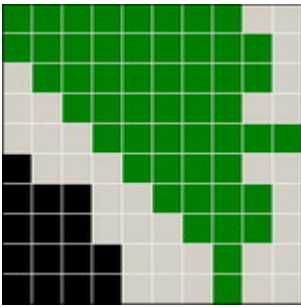


圖4.為兩個例子獲得的地圖。每個矩形對應於來自第二層的神經元。黑色神經元 - 第一類，綠色神經元 - 第二類。

此外，您可以嘗試SOFM如何在標準測試數據集上工作 - 這是眾所周知的Fisher Iris數據集。數據集由150個鳶尾花的測量數據組成：50個Iris-Setosa，50個Iris-Versicolor和50個Iris-Virginica（圖5）。

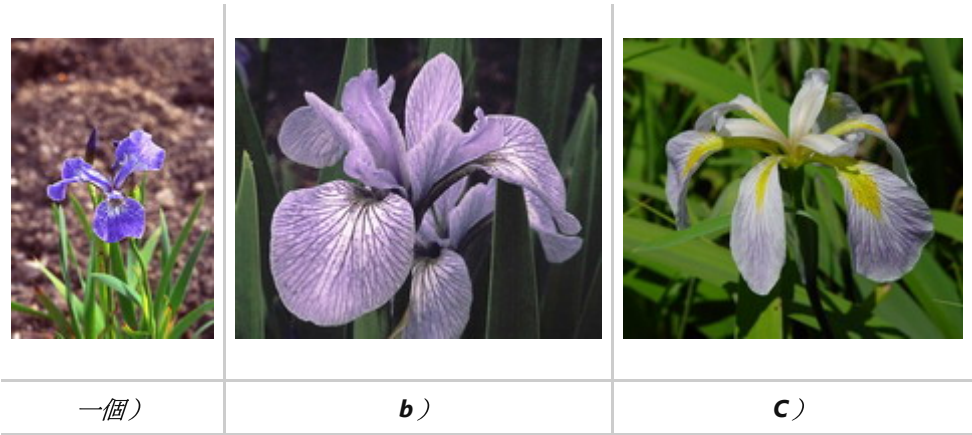


圖5.鳶尾花。a) Setosa b) Versicolor c) Virginica。

將第二層神經元的數量設置為64，將“Normalize input data”複選框設置為檢查位置（所有其他參數保持不變），然後按下“加載數據和表格映射”按鈕。然後，選擇iris.data文件。經過一些計算（您可以取消勾選“可視化”複選框以更快地執行），將構建地圖（圖6）。它上面的每種顏色都對應於一個Iris類。

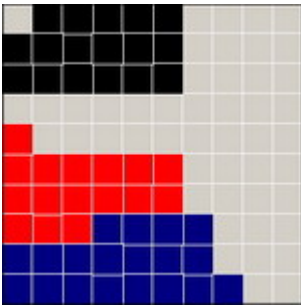


圖6.從Iris數據示例獲得的地圖。每個矩形對應於來自第二層的神經元。黑色神經元 - 第一類（Setosa），紅色神經元 - 第二類（Versicolor），藍色神經元 - 第三類（Virginica）。

## 潛入代碼

讓我們來看看它是如何工作的。在SOFM的對象模型中有兩個主要的類（圖7）：**Neuron**和**NeuralNetwork**。

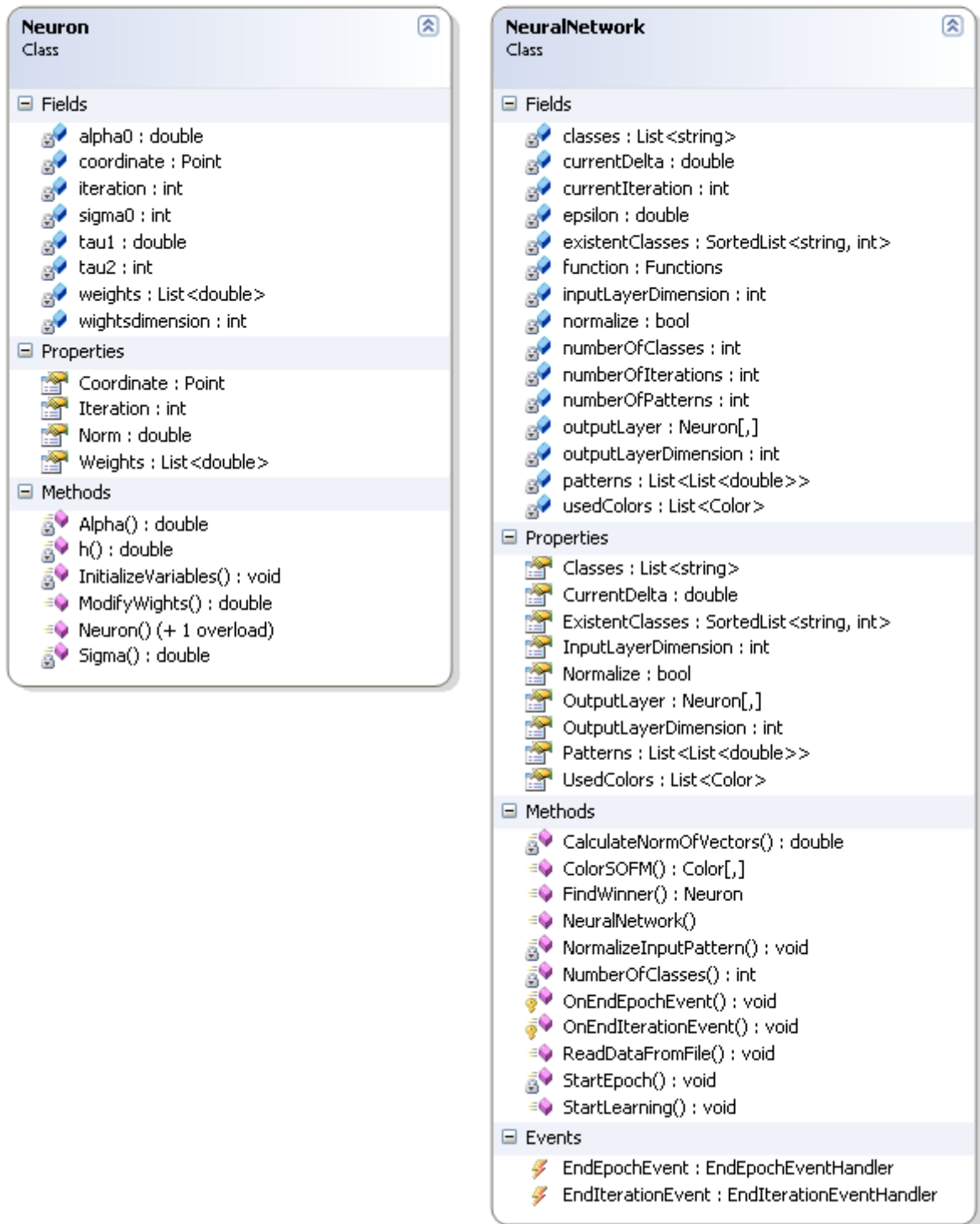


圖7.類圖。SOFM.Neuron和SOFM.NeuralNetwork。

每個Neuron班級都有一個Coordinate領域，它顯示了神經元在二維貪婪中的位置。該Weights字段是泛型類型的權向量，包含一些double值。的Neuron類可以執行ModifyWeights操作，這修改根據所選擇的拓撲鄰域函數（的神經元的權重向量h）類型，距離的贏家，和當前迭代的數目。

的NeuralNetwork類包含神經元的一個兩維網格，並且可以執行許多操作的：

隱藏 複製代碼

```
public NeuralNetwork(int m, int numberOfIterations, double epsilon, Functions f)
// Creates NN, with m x m neurons in second layer, and defined number of Iterations,
// epsilon and function of topological neighborhood.
public void ReadDataFromFile(string inputDataFileName)
// Reads data from inputDataFileName
public void StartLearning()
// Starts constructing of the map.
```



來自學習集的所有模式呈現為NN輸入 **numberOfIterations** 時間或者權值向量修改較小 (**currentEpsilon** <= **Epsilon**)。

[隱藏](#) [複製代碼](#)

```
public void StartLearning()
{
    int iterations = 0;
    while (iterations <= numberOfIterations && currentEpsilon > epsilon)
    {
        List<List<double>> patternsToLearn = new List<List<double>>(numberOfPatterns);
        foreach (List<double> pArray in patterns)
            patternsToLearn.Add(pArray);
        Random randomPattern = new Random();
        List<double> pattern = new List<double>(inputLayerDimension);
        for (int i = 0; i < numberOfPatterns; i++)
        {
            pattern = patternsToLearn[randomPattern.Next(numberOfPatterns - i)];

            StartEpoch(pattern);

            patternsToLearn.Remove(pattern);
        }
        iterations++;
        OnEndIterationEvent(new EventArgs());
    }
}
```

私人 **StartEpoch()** 方法響應尋找優勝者神經元並開始通過所有 **outputLayer** 神經元進行權重修改的過程。

[隱藏](#) [複製代碼](#)

```
private void StartEpoch(List<double> pattern)
{
    Neuron Winner = this.FindWinner(pattern);
    currentEpsilon = 0;
    for (int i = 0; i < outputLayerDimension; i++)
        for (int j = 0; j < outputLayerDimension; j++)
        {
            currentEpsilon +=
                outputLayer[i, j].ModifyWights(pattern, Winner.Coordinate,
                                                  currentIteration, function);
        }
    currentIteration++;
    currentEpsilon = Math.Abs(currentEpsilon /
                              (outputLayerDimension * outputLayerDimension));
    EndEpochEventArgs e = new EndEpochEventArgs();
    OnEndEpochEvent(e);
}
```

所以，一個時代是向NN輸入呈現其中一種模式的過程，並且當來自訓練集的所有模式呈現給NN輸入時，迭代是一組時期。每當紀元或迭代結束時，就會引發相應的事件。

[隱藏](#) [複製代碼](#)

```
protected virtual void OnEndEpochEvent(EndEpochEventArgs e)
{
    if (EndEpochEvent != null)
        EndEpochEvent(this, e);
}

protected virtual void OnEndIterationEvent(EventArgs e)
{
    if (EndIterationEvent != null)
        EndIterationEvent(this, e);
}
```

該 **ColorSOFM()** 方法返回彩色矩陣，其中每個元素對應於來自輸出層的神經元。地圖繪畫的顏色是隨機選擇的。

## 如何使用SOFM

要在程序中使用SOFM類，首先必須添加對`sofm.dll`程序集的引用。然後，您需要將以下行添加到該`using`部分：

[隱藏](#) [複製代碼](#)

```
using SOFM;
```

接下來，您必須創建`SOFM.NeuralNetwork`類的一個實例，如下所示：

[隱藏](#) [複製代碼](#)

```
SOFM.NeuralNetwork nn = new NeuralNetwork(NumberOfCards,  
                                           NumberOfIterations, Epsilon, function);
```

然後，添加訂閱者到`NeuralNetwork`事件（如果需要）：

[隱藏](#) [複製代碼](#)

```
nn.EndEpochEvent += new EndEpochEventHandler(nn_EndEpochEvent);  
nn.EndIterationEvent += new EndIterationEventHandler(nn_EndIterationEvent);
```

並且，定義`Normalize`實例的屬性（以`data`以規範化或不輸入數據）。然後，從文件中讀取數據：

[隱藏](#) [複製代碼](#)

```
nn.ReadDataFromFile(ofd.FileName);
```

並開始學習過程：

[隱藏](#) [複製代碼](#)

```
nn.StartLearning();
```

學習完成後，彩色地圖可以獲得如下：

[隱藏](#) [複製代碼](#)

```
System.Drawing.Color[,] colorMatrix = nn.ColorSOFM();
```

就這樣。

## 遺言

只有當來自不同類別的輸入向量不高度相關（不太相似）時，SOFM才能將輸入數據集劃分為類。這裡只提供了SOFM的基本概念。有關更詳細的說明，請參閱參考資料部分的一本書。

## 參考

在演示項目中使用Steema TeeChart控件進行圖表繪製。從Delphi遷移到.NET是一個很好的控制，lite版本是免費的。它可以從Steema Software [站點](#)下載。

有一些關於SOFM的額外研究的鏈接：

- [T. Kohonen](#)。自組織地圖 - 本書由SOFM的創建者撰寫。
- <http://www.cs.bham.ac.uk/~jlw/sem2a2/Web/Kohonen.htm> - 很好的例子。
- <http://www.google.ru/search?hl=ru&q=Kohonen+maps&lr=> - Google搜索也很有用。

## 執照

本文以及任何關聯的源代碼和文件均根據[GNU通用公共許可證 \(GPLv3\)](#)

## 分享



## 關於作者



### Bashir Magomedov

軟件開發人員 (高級)

英國

跟隨  
這位會員

工作：匯豐銀行 (<http://www.hsbc.co.uk/>)。

Regalia：CS，MCAD，MCPD博士：Web開發人員，MCTS：.Net Framework 2.0，3.5。

興趣：編程，人工智能，C#，.NET，HTML5，ASP.NET，SQL，LINQ。

婚姻狀況：已婚，女兒

博客：<http://www.magomedov.co.uk>

## 您也可能對。。。有興趣...

[DevOps的解決方案藍圖](#)

[AAMVA條形碼駕駛執照使用LEADTOOLS進行識別和創建](#)

[Kohonen的C++自組織映射及其在計算機視覺領域的應用](#)

[針對亞馬遜網絡服務 \(AWS\) 的MQTT發布](#)

[使用GPS和C#進行映射](#)

[使用英特爾®數學核心庫和Arduino創建](#)

## 評論和討論

添加評論或問題



Search Comments



[首頁](#) [上一頁](#) [下一頁](#)

### SOMTEST2

KuidoKylm 10-Jul-13 3:27



基於Bashir工作的修改程序

<http://mihkel.org/failid/Data%20Mining.docx>

<http://mihkel.org/failid/SOM.7z>

[回復](#) · [電子郵件](#) · [查看主題](#)

5.00 / 5 (2票)



---

## 皮膚分割

**lajpaal** 29-Mar-12 8:18



任何機構在SOM matlab中都有皮膚分割的想法？

[回复](#) · [電子郵件](#) · [查看主題](#)



---

## 節點運動

**Gobsek** 20-Oct-10 6:42



Hello!

What is the algorithm (formula) for node movements (toward to BMU) ?

Can you please refer how do you recalculate topology of neighbor nodes.

Regards.

[Reply](#) · [Email](#) · [View Thread](#)

5.00/5 (1 vote)



---

## My vote of 5

**henrywang0423** 11-Oct-10 0:35



One of the best articles I have once met. Thank you so much for your contribution. I of course gave you 5.

[Reply](#) · [View Thread](#)



---

## Self-organizing map

**je01621** 23-Nov-09 16:16



Project title: Intelligent Humanoid Behavior Robot using self-organizing map

Our software project is to make the robot move through voice commands(SAPI voice tools) eg. walk, turn left and turn right. It uses kinematics, gait analysis and self-organizing map specifically extended Kohonen's model. Kohonen's model is used to produce set of angles for the different servomotors. Our Robot is biped each leg has 5 servomotors. There are 10 degrees of freedom all in all. My problem was how to implement the Kohonen map using the values of x, y, and z coordinates as input.. Anyone has an idea? I really need help..Thanks... 😊

[Reply](#) · [Email](#) · [View Thread](#)



---

## college project

**prashantoct02** 21-Aug-09 0:37



Hi,

Was useful reading the article. Planning to implement SOMs in Semi Digital Watermarking in our college project. Request your thoughts on that.

Prashant  
(India)

[Reply](#) · [Email](#) · [View Thread](#)



---

## SOM --> RSOM

**Member 6038196** 5-Apr-09 8:01

▲ Thanks for the article. A great read. I've successfully implemented a SOM in my program. In order to make it an recurrent SOM do I simply add a leaky integrator, or are there other things I need to do?  
▼ Thanks

[Reply](#) · [Email](#) · [View Thread](#)



---

## Image Classification work anksw 19-Feb-09 13:54

▲ Hello Bashir,  
▼

We are a BPO based at New Delhi, India. We are looking for some solutions which can help us in our image classification work. We have millions of electronic images which are primarily legal documents of US real estate, like Deeds, Mortgages etc. These documents have variety of formats, which is difficult to classify. We want to identify different formats available, and then create a learning sample based on it, to further segregate them from millions of images.

We have basically two requirements:

1. An unsupervised learning mechanism which can help us to identify the unique pattern images from amongst mixed images.
2. Image/Layout classification mechanism which can take input as unique formats (identified from first), and then map each available image to any of the known formats.

Please let me know if you can help us or provide guidelines in achieving this task.

PS : Please let me know your email id where I can discuss with you further.

Regards,  
Ankit Jain  
ankitj@competentsoftware.com

[Reply](#) · [Email](#) · [View Thread](#)



---

## private int iteration; Henk Meijerink 26-Jan-09 14:41

▲ Hi Bashir,  
▼

I love your article and was pleased to rate it a 5.

In class Neuron your private int field 'iteration' seems to always have the value 1. It is never incremented or set to any other value.

Is that what you intended?

The field is only used in the private double method: h() and in the read-only property Iteration.  
The only other mention of 'iteration' in class Neuron is a formal parameter in the public double method: ModifyWeights(), where it actually hides the field.

Thanks, Henk.

[Reply](#) · [Email](#) · [View Thread](#)



---

## Need help for SOM using C#... kulingo 2-Jul-08 3:37

▲ Hi Mr. Bashir, I sent a mail for SOM using C#. Please check your "gmail" account.  
▼

[Reply](#) · [Email](#) · [View Thread](#)



**Thanks for this great article! [modified]****Florian.Witteler 13-Mar-08 5:23**

I have two little suggestions. It would be great, if you have time to edit your post.

1. On a Mac, the font "Symbol" is not available.

In the Chapter 'Learning rule' it would be great, if you don't use the Symbol font, but the greek letters in unicode. I posted all letters below, because Macs (and I think most of the UNIX machines) display it like this:  
e.g.: "...", here  $r$  is a distance to the winner-neuron"

[here](#) is a list of all greek letters in unicode (UTF8)

This should work on all platforms.

2. In the formula-graphic for the gaussian-curve, you forgot the initial minus.

Kind regards,

Florian

*modified on Wednesday, March 12, 2008 5:52 PM*

[Reply](#) · [Email](#) · [View Thread](#)

**formate of the input data****mralfarra 21-Jul-07 21:06**

hi,

please could you give me explanation about the formate of the input data ...

I have a samples in the formate of vectors , each vector represent one sample ... how can I use them in this s.w ?

thanx

[Reply](#) · [Email](#) · [View Thread](#)

**Re: formate of the input data****Bashir Magomedov 23-Jul-07 0:08**

Good day. Thank you for your message. I'm actually on vacations now and can't answer to you difenetly (I just have no access to my source codes machine). But you should consider sample files from demo project, and use the same data structure.



Thank you for your attention

[Reply](#) · [Email](#) · [View Thread](#)

**Re: formate of the input data****mralfarra 12-Aug-07 21:51**

thank you .... 😊

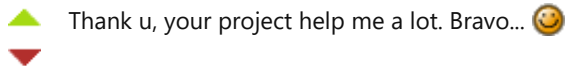


[Reply](#) · [Email](#) · [View Thread](#)



**Thanks a lot,**

**Le Tien 22-Jun-07 4:31**



Thank u, your project help me a lot. Bravo... 😊

[Reply](#) · [Email](#) · [View Thread](#)



---

### Re: Thanks a lot, Bashir Magomedov 22-Jun-07 14:35



You are welcome!



---

Thank you for your attention

[Reply](#) · [Email](#) · [View Thread](#)



---

### TeeChart.Lite.dll stano 13-Apr-07 9:08



The code looks good, but I'm curious as to what TeeChart.lite.dll is?

Could you just provide the code without these third party tools?

Thanks,

Paul.

[Reply](#) · [Email](#) · [View Thread](#)



---

### Re: TeeChart.Lite.dll Bashir Magomedov 13-Apr-07 12:49



Goof day, Paul!

Thank you for your note. The implementation of Neural network is fully detached from any vizualization features and provided in separated assembly named SOFM.DLL. So if you want to get only NN code, just download the sources and examine SOFM project (There are no other projects in source archive file).

About Teechart. It is used only in demonstration program for some visualizations, i.e. to show Hamming distance between patterns and others.

Good luck!

---

Thank you for your attention

[Reply](#) · [Email](#) · [View Thread](#)




---

### Excellent article! Martin Welker 26-Feb-07 20:51



You got my 5!

Martin

 : Vote for me![Reply](#) · [Email](#) · [View Thread](#)

## Missing data files

**Progress 8-Nov-06 22:31**

The data files for the demo are missing.

[Reply](#) · [Email](#) · [View Thread](#)

5.00/5 (1 vote)



## Re: Missing data files

**Bashir Magomedov 9-Nov-06 14:15**

哎呀。🙄 對我感到羞恥🙄。抱歉。  
固定！👀



感謝您的關注

[回复](#) · [電子郵件](#) · [查看主題](#)

## 源代碼丟失

**TBermudez 7-Nov-06 22:26**

源代碼鏈接已損壞。

[回复](#) · [電子郵件](#) · [查看主題](#)

## 回复：源代碼丟失

**Bashir Magomedov 7-Nov-06 22:28**

發布過程中。請等待5-10分鐘。謝謝



感謝您的關注

[回复](#) · [電子郵件](#) · [查看主題](#)

## 回复：源代碼丟失

**Daywalker213 12-Jan-11 13:14**

謝謝你這麼好的一篇文章..我正在計算一個jpg圖像中的血細胞。你能指導我在這個方向嗎？

[回复](#) · [電子郵件](#) · [查看主題](#)[刷新](#)

1

[一般](#) [新聞](#) [建議](#) [問題](#) [錯誤](#) [答案](#) [笑話](#) [讚美](#) [Rant](#) [Admin](#)



