PNN神經網路Grafcet建模和硬體合成

陳慶瀚

MIAT實驗室

2018-07-01

機率神經網路（Probabilistic Neural Network）是由 D. F. Specht在1988年提出，機率神經網路的架構基本上，是一個四層神經元結構的網路模型，屬於前向式的神經網路架構的一種，機率神經網路主要的理論基礎建立在於貝氏決策上。

運用機率神經網路的四層網路架構在於任意維度輸出的分類應用問題上，可以快速且有效地解決，在於輸入向量大小上的問題上，由於網路結構上的優點，並沒有限定一定是連續值或必須是二進位值，對於面對不同型式的問題上多了許多的方便性。而且機率神經網路在面臨因為系統外界環境因素改變，而需要加入新的分類資料時，僅需對新進之資料定義新分類資料的權值而無需像其他類型的網路架構改變全部的網路權值。

由於這一種網路學習速度十分得快速（嚴格來說學習所需的時間為零，因為其網路連結權值採一次設定，只是直接從訓練範例中載入所需數據，無迭代過程），對於錯誤的資訊具有相當的容忍性，面臨稀疏的樣本空間時也可根據問題直接調整參數，因此頗受重視。

## 1. PNN原理和演算法

## (1). 貝氏分類器（Bayes Classifier）

機率神經網路的主要架構建立在以貝氏分類法則（Bayes’ Methed）為基礎的分類器上。

假設一分類問題具有K個類別：

 (1)

此一分類問題的分類規則是由M維的特徵向量

 (2)

所決定，即在此M維樣本空間中，各分類的機率密度函數為特徵向量的函數：

 (3)

而貝氏分類器的決策公式：

 對所有的  (4)

為第k類的機率密度函數

則代表應為第k類，但被誤判的評估函數值

是第k類的事前機率（prior probability）

在理論上利用上面貝氏決策公式可解決分類的問題，但是實際應用中會有一個問題。因為在於一般情況下，我們並無法事先了解要訓練的資料真實的機率密度函數，但是為了要利用貝氏分類器解決問題，所以在使用貝氏分類器時，對於每一個類別的訓練資料，必須假設一個機率密度函數，在一般的情況下，為了使用上的方便假設該機率密度函數具有固定的形式，例如常態分佈。如此一來，便能夠利用已知的訓練資料來估計出機率密度函數的參數，方便在應用貝氏分類器上。

## (2). Parzen視窗法

因為貝氏分類器對於每個類別需要建立個別的機率密度函數，在實務應用上，通常很難事先決定機率密度函數，而且在一般的情況中，訓練資料多多少少都會存在有資料太少或是資料不足以建立完整的機率密度函數的現象，尚且有可能出現有部分資料不正確的問題，所以決定機率密度函數並不是十分容易的事情。

針對貝氏分類器的缺點，Parzen在1962年提出Parzen視窗法可以解決貝氏分類器的問題[....]。在圖1表示著如何利用Parzen法則來使用一個特徵估測一個類別，圖中的橫軸標示特徵值的座標，對於在訓練資料中的每一筆樣本建立一個以樣本的特徵值為中心的高斯曲線，最後把所有建立的曲線疊加成一個屬於該類別的機率密度函數.。

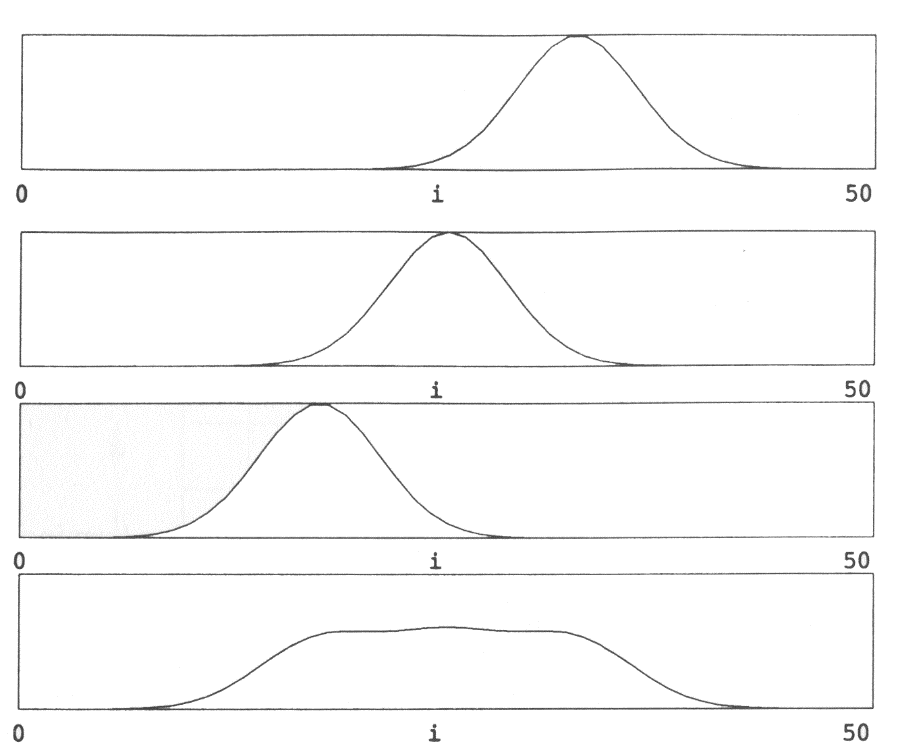


圖1. 三個不同的高斯函數與其疊加

Parzen的方法中表示隨著樣本數目的增加，根據樣本建立的曲線會愈接近真實的機率密度函數，可是這個理論存在一個缺點，由於這一項理論並無法精確的預估出真正需要多少個樣本數目，才能建立滿足要求的機率密度函數精確度，所以為了要建立一個好的分類器，最好可以隨時更新最佳的樣本參數。

實際問題中，訓練資料的向量空間通常都是大於一維。對於一個二維的特徵向量空間，二維的高斯函數可以表示成圖2, 將每一點的數值加總起來，便可以成為一個利用訓練資料估測出來的機率密度函數。同理，我們可將維度擴展到N維，如此一來這一個方法便可用到任意維度的特徵向量問題上。

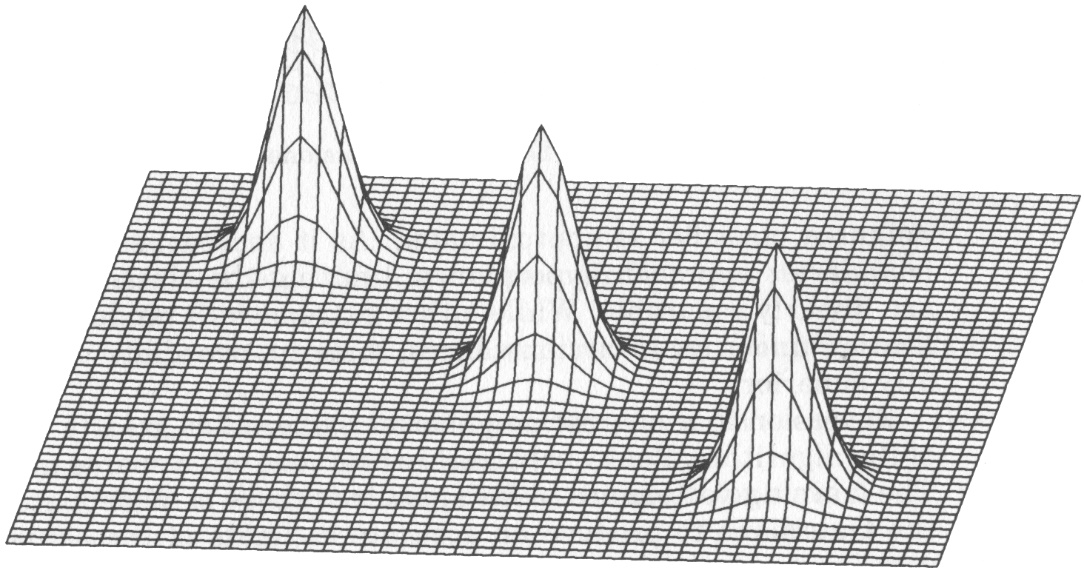


圖2. 三個斯函數，當

而將整個N維的機率密度函數利用數學形式表示則：

 (5)

 當分類A在X點位置的機率密度函數值

 訓練向量的個數

 訓練向量中的個數

 平滑參數

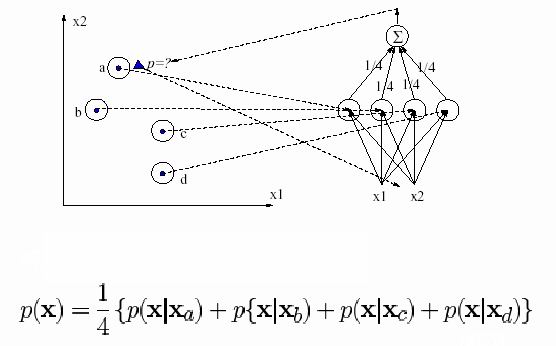
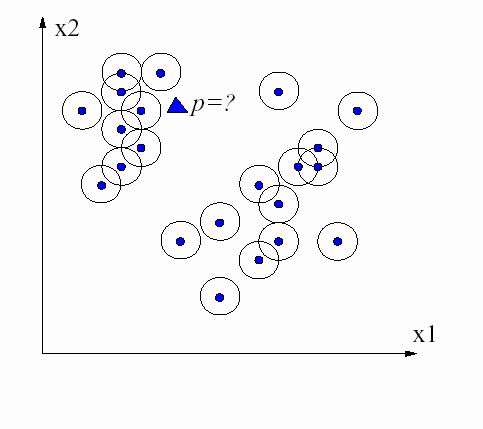
 在A分類中的訓練向量個數

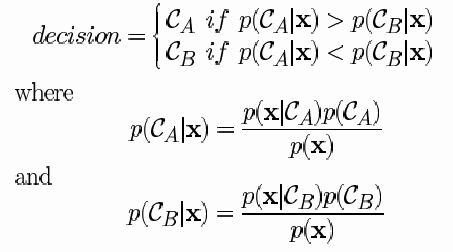
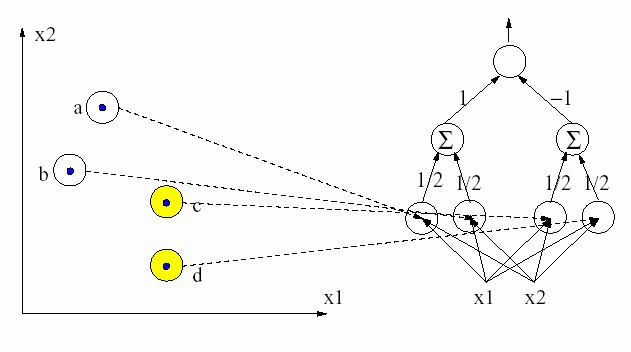
 測試分類向量

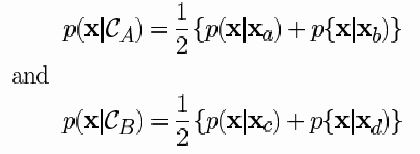
 在分類A中第訓練資料

 向量轉置

**(3). 估測機率密度函式**







## 2. PNN神經網路的設計

機率神經網路的設計將貝氏分類器的觀念引進到類神經網路的模型中，讓這一個新的神經網路模型不僅具有貝氏貝類器許多的優點，且為了改進貝氏分類器的機率密度函數不易建立的缺點。這是因為機率神經網路針對機率密度函數作了三個假設：

* 各分類的機率密度函數型態相同。
* 此共同的機率密度函數為高斯分佈，即常態分佈。
* 各分類的高斯分佈機率密度函數的變異矩陣為對角矩陣，且各對角元素的值相同，值為。

因為有了以上三個簡單的限制，而使得機率神經網路在應用上減少了貝氏分類器建構上的問題，增加了許多的便利性。

圖3是一個機率神經網路神經網路的架構。

|  |  |
| --- | --- |
|  | 圖3. 機率神經網路神經網路 |
| （a）系統架構； |
| （b）特徵單元(Pattern Unit)  （c） 輸出單元（output unit） |

特徵單元的作用是為了得到輸入向量與各別權重向量W的乘積

然後對作非線性的轉換，但不同於其他倒傳遞型類神經網路架構中常用的sigmoid轉換函數，在機率神經網路架構中是採用 函數。

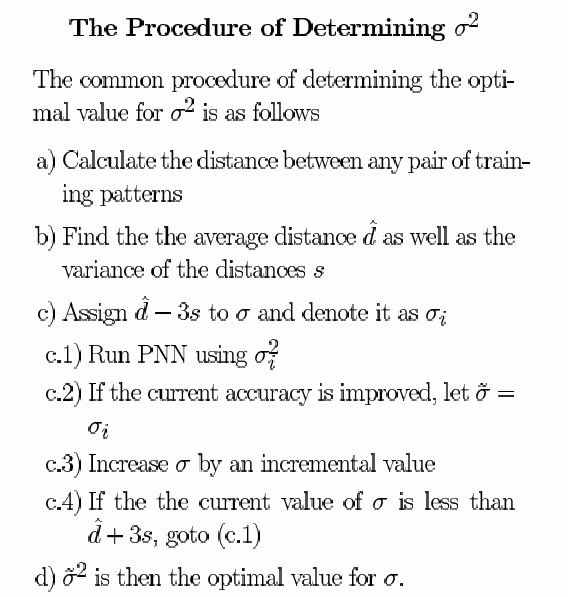
若X和都已正規化到單位長度則函數可以簡化為

 (6)

總和單元把各個從特徵單元得到的值加總起來

輸出單元為二輸入的神經元，目的在作結果的輸出決策圖中的是為了調整訓練資料中各分類的原始訓練個數不均的問題。

**決定**



## 3. PNN的訓練和使用

訓練階段

|  |  |
| --- | --- |
| 1 | void PNN::Training(ifstream &in) |
| 2 | { |
| 3 | int i,j; |
| 4 | in>>nb\_in>>nb\_out>>nb\_hidden; |
| 5 | W\_xh.Initialize(nb\_in,nb\_hidden); |
| 6 | W\_hy.Initialize(nb\_hidden,nb\_out); |
| 7 | for(i=0;i<nb\_hidden;i++) |
| 8 | { |
| 9 | for(j=0;j<nb\_in;j++)in>>W\_xh.m[j][i]; |
| 10 | for(j=0;j<nb\_out;j++)in>>W\_hy.m[i][j]; |
| 11 | } |
| 12 | } |

使用階段

|  |  |
| --- | --- |
| 1 | void PNN::Using(f1D &input,f1D &inf\_output) |
| 2 | { |
| 3 | float neth; |
| 4 | int i,j; |
| 5 | /\*----------- Basis function layer ----------\*/ |
| 6 | for(i=0;i<nb\_hidden;i++) |
| 7 | { |
| 8 | neth=0; |
| 9 | for(j=0;j<nb\_in;j++) |
| 10 | { |
| 11 | cout<<W\_xh.m[j][i]<<endl; |
| 12 | neth+=(input.m[j]-W\_xh.m[j][i])\*(input.m[j]-W\_xh.m[j][i]); |
| 13 | } |
| 14 | inf\_output.m[i]=exp(-neth/(2\*sigma\*sigma)); |
| 15 | } |
| 16 | /\*----------- summation layer ----------\*/ |
| 17 | // ...... |
| 18 | /\*----------- decision layer ----------\*/ |
| 19 | // ...... |
| 20 | } |

4 PNN神經網路硬體加速器設計

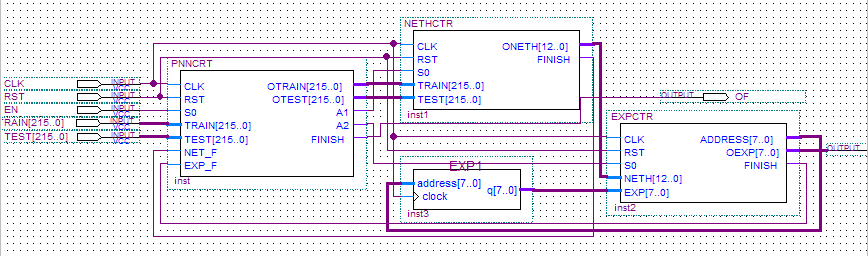
(1)機率密度函數模組

圖1是機率密度函數模組的GRAFCET模型。此模組主要在求隱藏層神經元輸出的歸屬值。步驟為先計算每個輸入值的維度和隱藏層神經元的維度(即為訓練資料的維度)差之絕對值，並累加其絕對值，並經過機率密函數公式的指數運算，所運算出的結果即為隱藏層神經元輸出歸屬值，圖中J為計數器，代表輸入值維度與隱藏層神經元維度的個數。此一機率密度函數模組的Grafcet模型採用循序架構，在stage1是計算輸入的24個人臉特徵值向量與PNN特徵單元的距離，stage2則計算正規化和EXP函數查表。



圖1、PDF模組的GRAFCET建模

根據MIAT方法論的合成規則，我們可以將1的GRAFCET模型合成到硬體電路，圖2(a)為電路架構，其模擬波形如圖2(b)。



主控制器

EXP函數模組

絕對值加法模組

(a)電路架構



(b)波形模擬

圖2、PDF模組合成電路和波形模擬(a)電路方塊(b)波形模擬

考量電路性能的提升，我們將圖2的循序GRAFCET模型重新設計為管線(pipelining)架構。圖3顯示將圖1中的stage1和stage2做管線化建模。圖4是新的波型模擬。



圖3、PDF模組管線化Grafcet建模

圖4、PDF管線化模組合成之VHDL電路功能波形模擬

我們比較循序架構和管線架構所使用的硬體資源和效能如表1所示。

表1、PDF模組電路使用的硬體資源和效能

|  |  |  |
| --- | --- | --- |
|  | 循序架構 | 管線化架構 |
| Total Logic Elements | 1,034 / 33,216 | 1,042 / 33,216 |
| Cycles/per inference | 54 | 50 |
| Most Critical Path Delay | 12.916 ns | 12.840 ns |
| Performance | 77.42 MHz | 77.88 MHz |

(2) 決策模組

PNN分類器決策模組的基本概念如同競爭式學習法則，尋找樣本群(即註冊的生物特徵)和輸入特徵最為相近似的類別。圖5為決策模組Grafcet建模，其主要功能在於找出特徵單元所輸出的最大機率密度函數值(PDF)，然後輸出最後分類結果。決策模組合成電路和波形模擬如圖6。



圖5、決策模組Grafcet模型



(a)電路方塊



(b)波形模擬

圖6、決策模組合成電路和波形模擬(a)電路方塊(b)波形模擬

此一電路所需硬體資源及效能如表2。

表2、決策模組電路效能及合成面積

|  |  |
| --- | --- |
|  | 決策模組 |
| Total Logic Elements | 48 / 33,216 |
| Cycles/per Decision | 2 |
| Most Critical Path Delay | 2.155 ns |
| Performance | 464.04 MHz |

(3) 完整的PNN分類器

首先將PNN分類器分割成2個子模組，即機率密度函數計算模組與決策模組，分割後的兩模組，可以再以IDEF0繼續分割其內部功能子模組，直到獲得基本的組合邏輯建構單元(building block)為止。以決策模組為例，我們再度分割成暫存器模組和比較器模組兩個子模組。圖5.11是PNN分類器的階層式模組架構。



圖7、PNN分類器的階層式模組架構

以IDEF0形式所描述的階層化模組，將交由GRAFCET來建立每個模組的離散事件模型，也就是將每個模組視為一個獨立動態系統進行行為建模。圖8是最上層的PNN分類器模組的GRAFCET模型，其中G1和G2兩個Step分別致動(activate)兩個sub-GRAFCET：機率密度函數(PDF)模組和決策(Decision)模組。



圖8、PNN分類器GRAFCET建模

圖9為系統整合硬體架構圖，其中包括主控制器、Net模組、PDF模組和決策模組，所需要硬體資源及效能如表3所示。



(a)電路方塊



(b)波形模擬

圖5.13、系統整合模組合成電路和波形模擬(a)電路方塊(b)波形模擬

表5.3、系統整合模組電路效能及合成面積

|  |  |
| --- | --- |
|  | 系統整合模組 |
| Total Logic Elements | 1,100 / 33,216 |
| Cycles/per Decision | 50 |
| Most Critical Path Delay | 12.764 ns |
| Performance | 78.35 MHz |