

IBM SPSS Neural Networks
26

IBM

附註

使用此資訊和支援的產品之前，請先閱讀 第 15 頁的『注意事項』中的資訊。

產品資訊

若新版本未聲明，則此版本便適用於 IBM® SPSS Statistics 26.0.0 版以及之後發行的所有版本和修正。

目錄

神經網路(W)	1	半徑式函數	9
Introduction to Neural Networks (神經網路簡介)	1	分割	11
什麼是神經網路?	1	架構	11
神經網路結構	2	輸出	12
多層感知器	3	儲存	12
分割	4	匯出	13
架構	5	選項	13
訓練	6	注意事項	15
輸出	7	商標	16
儲存	8	索引	17
匯出	8		
選項	8		

神經網路(W)

下列神經網路功能包含在 SPSS® Statistics Premium Edition 或「神經網路」選項中。

Introduction to Neural Networks (神經網路簡介)

神經網路因為功能強大、彈性和易用，所以是許多預測性資料採礦應用程式較常使用的工具。預測性神經網路在基礎程序複雜的應用程式特別有用，例如：

- 預測客戶需求以便將製作模式和交付成本合理化。
- 預測直效郵遞行銷回應的機率，以決定郵寄名單中的哪些家庭應該寄送優惠訊息。
- 為申請人評分，以決定申請人之擴充信用的風險。
- 在保險請求給付資料庫中偵測詐欺交易。

用於預測性應用程式的神經網路，像是多層認知 (MLP) 和半徑式函數 (RBF) 網路，會因為模型預測結果可與目標變數的已知數值相比較而加以監督。「預測及決策樹狀結構」可讓您調適 MLP 和 RBF 網路，並儲存所產生的模型以供評分。

什麼是神經網路？

術語神經網路適用於鬆相關的系列模型，以大型參數空間和靈活的結構為特點，從腦功能研究遞減。隨著系列增長，大部分新模型設計用於非生化應用程式，儘管許多關聯的術語仍反映其來源。

神經網路的特定定義視其使用領域而改變。雖然，沒有單個定義能夠適當地涵蓋整個系列的模型；但就目前而言，請考量下列說明¹：

神經網路是一個巨大的平行配送處理器，它具有儲存經驗知識並提供使用的自然傾向。它類似於大腦的下面兩方面：

- 透過學習過程從網路獲得知識。
- 中間神經元連線強度（稱作突觸加權）用來儲存知識。

有關此定義為何過於約束的討論，請參閱²。

為了將神經網路與使用此定義的傳統統計方法區分，未說明的只有實際的定義文字。例如，傳統的線性回歸模型可透過最小平方方法獲得知識，並將該知識儲存在迴歸係數中。在此處，它就是神經網路。實際上，您可以辨論此線性回歸是某些神經網路的特例。但是，線性回歸具有嚴格的模型結構，且在學習資料之前強制作出了一組假設。

相對而言，上方的定義對於模型結構與假設的需求最小。因此，神經網路可以結合眾多統計模型，而不需要提前對應變數和自變數之間的關係作出特定假設。此關係形式在學習過程中判定。如果應變數和自變數之間的線性關係恰當，則神經網路的結果應該近似線性回歸模型的結果。如果非線性關係更為恰當，則神經網路會自動近似「正確的」模型結構。

1. Haykin, S. 1998. *Neural Networks: A Comprehensive Foundation*, 2nd ed. New York: Macmillan College Publishing.

2. Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

此彈性的犧牲是無法簡單說明神經網路的突觸加權。因此，如果您嘗試說明產生應變數和自變數之間關係的基礎過程，則最好使用更傳統的統計模型。但是，如果模型的說明性不是那麼重要，則通常可以使用神經網路來更快取得良好的模型結果。

神經網路結構

雖然神經網路在模式結構和假設上的要求很低，但若能瞭解一般網路架構還是很有用。多層認知 (MLP) 或半徑式函數 (RBF) 網路是將目標變數（也稱為輸出）的預測誤差降至最低的預測變數（也稱為輸入或自變數）的函數。

請考慮在產品隨附的資料集 *bankloan.sav* 中，您想要從一堆貸款申請人找出可能的違約者。套用在這個問題上的 MLP 或 RBF 網路，就是可將預測預設值的誤差降至最低的測量函數。以下圖形有助於瞭解此函數的相關形式。

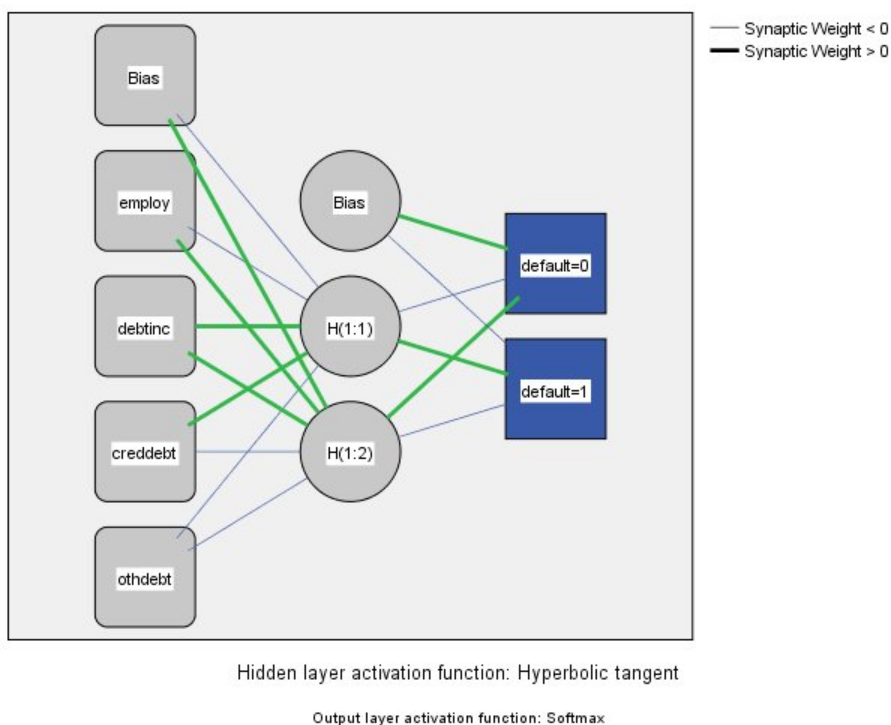


圖 1. 具有一個隱藏階層的前饋架構

此結構稱為前饋架構，因為在網路中的連線未經任何反饋迴圈便從輸入階層向前流至輸出階層。在本圖中：

- 輸入階層包含預測變數。
- 隱藏階層包含無法觀察的節點或單元。每一個隱藏單元的值都是預測變數的某種函數；此函數的精確形式部分取決於網路類型，部分取決於使用者可控制的規格。
- 輸出階層包含回應值。由於預設值的歷程是含有兩個類別的類別變數，因此將它重新編碼為兩個指標變數。每一個輸出單元都是隱藏單元的某種函數。同樣地，此函數的精確形式部分取決於網路類型，部分取決於使用者可控制的規格。

MLP 網路可容許第二個隱藏階層；在此狀況下，第二個隱藏階層的每一個單元都是在第一個隱藏階層中該單元的函數，而每一個回應值都是在第二個隱藏階層中該單元的函數。

多層感知器

「多層感知器」(MLP) 程序會依據預測變數值，為一個或多個依 (目標) 變數產生預測模型。

範例。 下列是兩個使用 MLP 程序的情況：

銀行放貸人員必須能辨識具有哪些特質的人可能會拖欠貸款，並使用這些特質來識別好和壞的信用風險。使用過去客戶作為樣本，她可以使用過去客戶的保留樣本來訓練多層感知器和驗證分析，然後使用網路來將準客戶分類為好的與壞的信用風險。

醫院系統著重於追蹤成本和入院接受心肌梗塞 (MI，或「心臟病」) 治療之病患的住院日數。取得這些量數的正確估計值可讓管理人員在病患接受治療的同時，正確地管理可用的空床。使用接受 MI 治療之病患的治療記錄樣本，管理人員可訓練網路來預測成本和住院日數。

資料考量

因變數。 因變數可以是：

- 名義. 當變數值代表實質上並未等級化的種類時 (例如，有員工工作的公司部門)，則此變數可視為名義。名義變數的範例包括區域、郵遞區號及宗教團體。
- 序數. 當變數值代表實質上已等級化的種類時 (例如，服務滿意度從非常不滿意到非常滿意分級)，則此變數可視為序數。序數變數的範例包括代表滿意度或信賴程度的態度分數以及偏好等級分數。
- 尺度. 如果某一變數可視為尺度 (連續)，表示它的值代表含有實際意義矩陣的已排序種類，因此適合比較值之間的距離。尺度變數的範例包括以年份表示的年齡及以千元為單位的收入。

本程序假設已指定給所有應變數適當的測量層級，但您可以在來源變數清單的變數上按一下滑鼠右鍵，並選取蹦現功能表上的測量層級，暫時變更變數的測量層級。

變數清單中各變數旁的圖示可識別測量層級和資料類型：

預測值變數。 可指定預測值為因素 (類別) 或共變量 (尺度)。

類別變數編碼。 本程序在整個程序期間，會使用 one-of-c 編碼來暫時記錄類別預測變數與因變數。如果一個變數有多個 c 類別，則變數會儲存為 c 向量，其中第一個類別標示為 (1,0,...,0)，第二個類別標示為 (0,1,0,...,0)，...，最後一個類別標示為 (0,0,...,0,1)。

此編碼方法會增加加權鍵結值的數目，這會導致訓練變慢；然而，更多「精簡」的編碼方法通常會產生不適合的神經網路。如果您的網路訓練進行的非常慢，您可以將類似的類別組合在一起，或捨棄具有極少類別的觀察值，以嘗試減少類別預測值中的類別個數。

所有的 one-of-c 編碼都是以訓練資料為基礎，即使已定義測試或保留樣本也是如此 (請參閱第 4 頁的『分割』)。因此，如果測試或保留樣本所包含的觀察值之預測值類別不在訓練資料中，則這些觀察值不會用於程序或評定中。如果測試或保留樣本所包含的個案之因變數類別不在訓練資料中，則這些個案不會用於程序，但可能用於評分中。

調整。 尺度因變數和共變量會依預設進行調整，以改善網路訓練。所有的調整都是以訓練資料為基礎，即使已定義測試或保留樣本也是如此 (請參閱第 4 頁的『分割』)。也就是視調整的類型而定，僅使用訓練資料來計算平均數、標準差、共變量或因變數的最小值或最大值。如果您指定變數來定義分割，很重要的是這些共變量或因變數必須在訓練樣本、測試樣本和保留樣本之間有類似的分配。

次數加權。 此程序會忽略次數加權。

複製結果。 如果您要精確地複製結果，除了使用相同的程序設定外，請為亂數產生器使用相同的初始化值、相同的資料順序和相同的變數順序。此問題的詳細資料如下：

- **亂數產生器。** 程序會在隨機指派分割、將加權鍵結值初始化隨機分成子樣本、將自動架構選項隨機分成子樣本的期間內使用亂數產生器，以及使用用於加權初始化和自動架構選擇的模擬退火演算法。未來若要重新產生相同的隨機化結果，請先使用與亂數產生器的相同初始化值，再執行每個「多層感知器」程序。
- **觀察值順序。** 線上和小型批次訓練方法（請參閱 第 6 頁的『訓練』）明確地依存於觀察值的順序；然而，即使是批次訓練也依存於觀察值的順序，因為加權鍵結值的初始化包含從資料集中抽出子樣本。

若要將順序效應降到最低，請以隨機方式排列觀察值。若要驗證某個解決方案的穩定性，您也許會想要取得幾種不同的解決方案，其觀察值皆以不同的隨機順序排列。在檔案極大的情況下，可進行多次運算，以不同的隨機順序排列一個觀察值的樣本。

- **變數順序。** 結果可能會受到因素和共變異數清單中的變數順序所影響，因為當變數順序變更時，會指派不同的初始值樣式。如同個案順序的影響，您可以嘗試不同的變數順序（只要在因素和共變異數清單中進行拖放）以評估給定解答的穩定性。

建立多層感知器網路

在功能表上，選擇：

分析 > 神經網路 > 多層感知器...

1. 選取至少一個因變數。
2. 選取至少一個因素或共變量。

您可以選擇性地在「變數」標籤上變更調整共變量的方法。選項為：

- **標準化。** 減去平均數，然後除以標準差 $(x - \text{mean})/s$ 。
- **常態化。** 減去最小值，然後除以範圍 $(x - \text{min})/(\text{max} - \text{min})$ 。常態化的值介於 0 和 1 之間。
- **調整後常態化。** 減去最小值，然後除以範圍的調整版本 $[2*(x - \text{min})/(\text{max} - \text{min})] - 1$ 。調整後常態化的值介於 -1 和 1 之間。
- **無。** 沒有調整共變量。

具有未知測量層級的欄位

當資料集中一或多個變數（欄位）的測量層級不明時，就會顯示「測量層級」警示。由於測量層級會影響此程序的結果計算，因此所有變數皆必須具有已定義的測量層級。

掃描資料

讀取作用中資料集的資料，並且針對目前具有未知測量層級的任何欄位指派預設的測量層級。若為大型資料集，則讀取時可能需要一些時間。

手動指派

列出測量層級不明的所有欄位。您可以將測量層級指派給那些欄位。您也可以「資料編輯器」的「變數清單」窗格中指派測量層級。

由於測量層級是此程序的重要項目，因此您在所有欄位皆擁有已定義的測量層級之前，無法執行此程序。

分割

分割區資料集。 此組別指定將作用中資料集區隔成訓練、檢定和保留樣本的方法。訓練樣本由用來訓練神經網路的資料記錄所組成；在資料集中有某些比例的觀察值必須指定為訓練樣本以取得模式。檢定樣本是一組獨立的資料記錄，用來追蹤訓練期間的錯誤以避免訓練過度。建議您建立訓練樣本，且當檢定樣本小於訓練樣本

時，一般來說網路訓練更加高效。保留樣本是另一個獨立的資料記錄集，用來存取最終神經網路；由於保留觀察值並未用來建置模型，保留樣本的錯誤為模型的預測能力提供「誠實」的估計。

- **根據觀察值的相對數字隨機指派觀察值。** 指定隨機指派給每一個樣本（訓練、檢定和保留）的觀察值的相對數字（比例）。% 欄報告根據您所指定的相對數字，將要指派給每一個樣本的觀察值的百分比。

例如，指定 7、3、0 作為訓練、檢定和保留樣本的相對數字，對應於 70%、30% 和 0%。指定 2、1、1 作為相對數字對應於 50%、25% 和 25%；1、1、1 對應於將資料集劃分為訓練、檢定和保留三部分相等。

- **使用分割變數來指定觀察值。** 指定一數值變數將作用中資料集的每一個觀察值指定給訓練、檢定或保留樣本。將變數中含有正值的觀察值指定給訓練樣本，將值為 0 的觀察值指定給檢定樣本，或將值為負數指定給保留樣本。含有系統遺漏值的觀察值會從分析中排除。任何分割變數的使用者遺漏值永遠視為有效。

附註：使用分割變數將無法保證在程序連續執行時會產生相同的結果。請參閱主要多層感知器主題中的「複製結果」。

架構

「架構」標籤是用來指定網路的架構。程序可自動選取「最佳」的架構，或是您可以指定自訂的架構。

自動架構選項會建立包含一個隱藏階層的網路。指定隱藏階層中所允許的單位最小和最大數目，則自動架構選項會計算隱藏階層中「最佳」的單位數目。自動架構選項會使用隱藏和輸出階層的預設啟動函數。

自訂架構選擇讓您可專業地控制隱藏和輸出階層，而且在您事先知道想要的架構或需要扭曲自動架構選項的結果時最為有用。

隱藏階層

隱藏階層包含無法觀察的網路節點（單元）。每個隱藏單元皆為輸入加權總和的函數。此函數為啟動函數，而加權值是以估計演算法來決定。若網路包含第二個隱藏階層，則在此第二個階層中的每個隱藏單元皆為第一個隱藏階層中的單元加權總和函數。在兩個階層中會使用相同的啟動函數。

隱藏階層的數目。一個多層感知器可以有一個以上隱藏階層。

啟動函數。啟動函數會將階層中的單元加權總和「鏈結」至後續階層中單元的值。

- **雙曲正切。** 此函數格式如下： $\gamma(c) = \tanh(c) = (e^c - e^{-c}) / (e^c + e^{-c})$ 。必須使用實數引數並將它們轉換至範圍 (-1, 1)。當使用自動架構選項時，此為隱藏階層中所有單位的啟動函數。
- **Sigmoid。** 此函數格式如下： $\gamma(c) = 1 / (1 + e^{-c})$ 。必須使用實數引數並將它們轉換至範圍 (0, 1)。

單位數。您可以明確指定每個隱藏階層中的單位數目，也可以讓估計演算法自動決定。

輸出階層

輸出階層包含目標變數（因變數）。

啟動函數。啟動函數會將階層中的單元加權總和「鏈結」至後續階層中單元的值。

- **單位。** 此函數格式如下： $\gamma(c) = c$ 。必須使用實數引數並以原樣傳回。當使用自動架構選項時，如果有任何尺度因變數，則此為輸出階層中所有單位的啟動函數。
- **Softmax。** 此函數格式如下： $\gamma(c_k) = \exp(c_k) / \sum_j \exp(c_j)$ 。必須使用實數引數的向量並將它轉換至其元素落在 (0, 1) 範圍且總和為 1 的向量。Softmax 只有在所有因變數為類別時才能使用。當使用自動架構選項時，如果所有因變數為類別，則此為輸出階層中所有單位的啟動函數。
- **雙曲正切。** 此函數格式如下： $\gamma(c) = \tanh(c) = (e^c - e^{-c}) / (e^c + e^{-c})$ 。必須使用實數引數並將它們轉換至範圍 (-1, 1)。

- **Sigmoid**。此函數格式如下： $\gamma(c) = 1/(1+e^{-c})$ 。必須使用實數引數並將它們轉換至範圍 (0、1)。

調整尺度因變數。只有在已選取至少一個尺度因變數時，才能使用這些控制項。

- **標準化**。減去平均數，然後除以標準差 $(x - \text{mean})/s$ 。
- **常態化**。減去最小值，然後除以範圍 $(x - \text{min})/(\text{max} - \text{min})$ 。常態化的值介於 0 和 1 之間。如果輸出階層使用 Sigmoid 啟動函數，這會是尺度因變數所需的調整方法。修正選項指定可套用為修正調整公式的一個小數目 ϵ ；此修正可確保所有調整的因變數值會在啟動函數的範圍內。尤其是數值 0 和 1 會定義 Sigmoid 函數的範圍限制，但卻不在該範圍內，且當 x 為最小值和最大值時，數值 0 和 1 會出現在未修正的公式中。修正的公式為 $[x - (\text{min} - \epsilon)] / [(\text{max} + \epsilon) - (\text{min} - \epsilon)]$ 。請指定一個大於或等於 0 的數值。
- **調整後常態化**。減去最小值，然後除以範圍的調整版本 $[2 * (x - \text{min}) / (\text{max} - \text{min})] - 1$ 。調整後常態化的數值介於 -1 和 1 之間。如果輸出階層使用雙曲正切啟動函數，這會是尺度因變數所需的調整方法。修正選項指定可套用為修正調整公式的一個小數目 ϵ ；此修正可確保所有調整的因變數值會在啟動函數的範圍內。尤其是數值 -1 和 1 會定義雙曲正切函數的範圍限制，但卻不在該範圍內，且當 x 為其最小值和最大值時，數值 -1 和 1 會出現在未修正的公式中。修正的公式為 $\{2 * [(x - (\text{min} - \epsilon)) / ((\text{max} + \epsilon) - (\text{min} - \epsilon))] - 1\}$ 。請指定一個大於或等於 0 的數值。
- **無**。沒有調整尺度因變數。

訓練

「訓練」標籤是用來指定訓練網路的方式。訓練類型和最佳化演算法會決定可用的訓練選項。

訓練類型。練習類型會決定網路處理記錄的方式。選取以下任一訓練類型：

- **批次**。僅在傳輸所有訓練資料記錄之後更新加權鍵結值；也就是，批次訓練使用訓練資料集中的所有記錄資訊。因為批次訓練可直接降低總錯誤數，因此為較常使用的選項；然而，批次訓練需要更新加權數次直到達到其中一個中止規則，因此可能需要進行多次資料傳輸。這對「較小」的資料集而言最為實用。
- **線上**。在每一訓練資料記錄之後更新加權鍵結值；也就是，線上訓練一次使用一個記錄資訊。線上訓練會持續取得記錄並更新加權直到達到其中一個中止規則。若所有的記錄全都使用過，但沒有達到任何一個中止規則時，則程序會再次利用資料記錄持續進行。對於具有相關預測值的「較大」資料集而言，線上訓練優於批次訓練；也就是，若有許多記錄和輸入，且其數值並非彼此獨立，則比起批次訓練而言，線上訓練可較快速地取得合理的答案。
- **小型批次**。將訓練資料記錄分成大約相等大小的組別，並在傳輸一個組別後更新加權鍵結值；也就是，小型批次會使用記錄組別的資料。如有需要，程序會再次利用該資料組別。小型批次是批次和線上訓練之間的折衷方式，且最適合用於「中型」資料集。程序可自動決定每一小型批次的訓練記錄數目，或是您可以指定一個大於 1 但小於或等於儲存在記憶體體的觀察值最大數目的整數。您可以在「選項」標籤上，設定可儲存在記憶體體上的觀察值數目上限。

最佳化演算法。這是用於估計加權鍵結值的方法。

- **尺度化共軛梯度**。調整共軛梯度使用方法的假設僅適用於批次訓練類型，因此此方法不適用於線上或小型批次訓練。
- **梯度下降**。此方法必須和線上或小型批次訓練一起使用；也可和批次訓練一起使用。

訓練選項。訓練選項可讓您微調最佳化演算法。一般不太需要變更這些設定，除非網路在估計時遇到問題。

尺度化共軛梯度演算法的訓練選項包括：

- **初始 Lambda (λ)**。尺度化共軛梯度演算法之 Lambda (λ) 參數的初始值。指定一個大於 0 且小於 0.000001 的數值。
- **初始 Sigma**。尺度化共軛梯度演算法之 Sigma 參數的初始值。指定一個大於 0 且小於 0.0001 的數值。

- **區間中心和區間偏移。** 區間中心 (a_0) 和區間偏移 (a) 定義區間 $[a_0 - a, a_0 + a]$ ，在這個區間內使用模擬退火演算法時隨機產生加權向量。模擬退火演算法是在最佳化演算法應用期間，為達到尋找整體最小值之目的，作為破壞局部最小值之用。此方法適用於加權初始化和自動架構選項。為區間中心指定一個數值，並為區間偏移指定一個大於 0 的數值。

梯度下降演算法的訓練選項包括：

- **初始學習率。** 梯度下降演算法之學習率的初始值。較高的學習率表示網路將訓練更快，但可能造成系統不穩定。指定一個大於 0 的數值。
- **學習率下界。** 梯度下降演算法學習率下界。此設定僅適用於線上和小型批次訓練。指定一個大於 0 但小於初始學習率的數值。
- **動量。** 梯度下降演算法的初始動量參數。動量項目可有助於預防太高學習率所造成的系統不穩定。指定一個大於 0 的數值。
- **學習率縮減 (以週期為單位)。** 當梯度下降和線上或小型批次訓練一起使用時，會需要週期的數目 (p) 或訓練樣本的資料傳輸以將初始學習率降低至學習率的下界。此項目可讓您控制學習率減少因素 $\beta = (1/p - K) * \ln(\eta_0 / \eta_{low})$ ，其中 η_0 為初始學習率、 η_{low} 為學習率的下界，而 K 為訓練資料集中小型批次的總數目 (或線上訓練的訓練記錄數目)。請指定一個大於 0 的整數。

輸出

網路架構。 顯示有關神經網路的摘要資訊。

- **說明。** 顯示有關神經網路的資訊，包括因變數、輸入和輸出單位的數目、隱藏階層和單位的數目，以及啟動函數。
- **圖。** 以無法編輯的圖表來顯示網路結構圖。請注意，隨著共變量和因素層級的數目增加，結構圖也會越來越難解讀。
- **加權鍵結值。** 顯示係數估計值，此估計值會顯示已知階層中的單位和下一階層中的單位之間的關係。即使將作用中資料集分割成訓練、測試和保留資料，加權鍵結值仍會以訓練樣本為基礎。請注意，加權鍵結值的數目可以變得相當大，因此這些加權一般不適用於解讀網路結果。

網路效能。 顯示用於判斷模式是否「良好」的結果。附註：此組別中的圖表是以合併訓練和測試樣本為基礎，或是如果沒有測試樣本時，僅以訓練樣本為基礎。

- **模型摘要。** 以分割和整體方式顯示神經網路結果的摘要，包括錯誤、不正確預測的相對錯誤或百分比、用於中止訓練的中止規則和訓練時間。

當識別函數、Sigmoid 函數或雙曲正切啟動函數套用在輸出階層時，錯誤為平方和誤差。當 Softmax 啟動函數套用在輸出階層時，錯誤為交叉熵誤差。

根據因變數測量層級來顯示不正確預測的相對錯誤或百分比。如果任一因變數具有尺度測量層級，則會顯示平均整體相對錯誤 (相對於平均數模式)。如果所有的因變數為類別，則會顯示不正確預測的平均百分比。也會顯示個別因變數之不正確預測的相對錯誤或百分比。

- **分類結果。** 以分割和整體方式來顯示每個類別因變數的分類表。每個表會提供每個因變數類別之正確或不正確分類的觀察值數目。也會報告已正確分類之總觀察值的百分比。
- **ROC 曲線。** 顯示每個類別因變數的 ROC (接收器作業特性) 曲線。也會顯示可提供每個曲線下之區域的表格。對一個特定的因變數來說，ROC 會為每個類別顯示一個曲線。如果因變數有兩種類別，則每個曲線會視討論中的類別為正向狀態和另一個類別進行比較。如果因變數有兩種以上的類別，則每個曲線會視討論中的類別為正向狀態和其他的類別整合進行比較。
- **累積收益圖。** 顯示每個類別因變數的累積收益圖。為每個因變數類別顯示一條曲線的方法和 ROC 曲線的方法相同。

- **增益圖。** 顯示每個類別因變數的增益圖。為每個因變數類別顯示一條曲線的方法和 ROC 曲線的方法相同。
- **觀察圖表的預測。** 顯示每個因變數的依觀察值預測圖表。如果是類別因變數，則會顯示每個回應值類別之預測虛擬機率的集群盒形圖，其中以觀察回應值類別作為集群變數。如果是尺度因變數，則會顯示散佈圖。
- **預測圖表的殘差。** 顯示每個尺度因變數的依預測值殘差圖表。殘差與預測值之間應該沒有可見的樣式。僅為尺度因變數產生此圖表。

觀察值處理摘要。 顯示觀察值處理摘要表，其中摘要出分析中包含和排除的觀察值數、總數，以及是依訓練、測試和保留樣本包含和排除。

自變數重要性分析。 執行敏感度分析，此分析會計算在決定神經網路時每個預測值的重要性。分析是以合併訓練和測試樣本為基礎，或是如果沒有測試樣本時，僅以訓練樣本為基礎。這會建立可顯示每個預測值之重要性和常態化重要性的表格和圖表。請注意，如果有大量的預測值或觀察值，則敏感度分析的計算會很昂貴且耗時。

儲存

「儲存」標籤是用來將預測值儲存為資料集中的變數。

- **儲存每個因變數的預測值或類別。** 此動作會儲存尺度因變數的預測值和類別因變數的預測類別。
- **儲存每個因變數的預測虛擬機率或類別。** 此動作會儲存類別因變數的預測虛擬機率。對於前 n 個類別，系統會為每個類別儲存個別的變數，其中 n 是在「要儲存的類別」行中指定。

已儲存變數的名稱。 自動名稱產生會確保您能保留所有的工作。自訂名稱可讓您捨棄/取代先前執行的結果，而不必先刪除在「資料編輯器」中儲存的變數。

機率和虛擬機率

具有 Softmax 啟動函數和交叉熵錯誤的類別因變數會有每個類別的預測值，而類別中的每個預測值為觀察值歸入類別的機率。

具有平方和誤差的類別因變數會有每個類別的預測值，但無法將預測值解讀為機率。程序會儲存這些預測虛擬機率，即使任一機率小於 0 或大於 1，或是特定的因變數總和不是 1。

ROC、累積收益圖和增益圖（請參閱第 7 頁的『輸出』）是根據虛擬機率而建立。當任一虛擬機率小於 0 或大於 1，或是特定變數的總和不是 1 時，首先會將機率調整在 0 和 1 之間，且總和為 1。虛擬機率是透過除以其總和來進行調整。例如，如果觀察值具有三類別因變數之 0.50、0.60 和 0.40 的預測虛擬機率，則每個虛擬機率會除以總和 1.50 以得到 0.33、0.40 和 0.27。

如果任一虛擬機率為負數，則在進行上述的調整前，會將最低的絕對值增加到所有虛擬機率。例如，如果虛擬機率為 -0.30、0.50 和 1.30，則首先將 0.30 加到每個值以取得 0.00、0.80 和 1.60。接著，將每個新數值除以總和 2.40 以取得 0.00、0.33 和 0.67。

匯出

「匯出」標籤是用來將每個因變數的加權鍵結估計值儲存到 XML (PMML) 檔案。您可以使用這個模型檔案，將模式資訊套用到其他資料檔中以進行評分工作。如果已定義分割檔，則此選項無法使用。

選項

使用者遺漏值。 因素必須有有效值，以將觀察值納入分析。這些控制項可讓您決定是否要在因素和類別因變數中，將使用者遺漏值視為有效值。

中止規則。 這些是決定何時停止訓練神經網路的規則。訓練會透過至少一個資料傳輸繼續執行。可依據下列準則來中止訓練，並依照下列順序檢查準則。在遵循的中止規則定義中，步驟會對應至線上和小型批次方法的資料傳輸，以及對應至批次方法的疊代。

- **最大步驟數目 (不包含錯誤縮減)。** 檢查錯誤縮減前所允許步驟數目。如果在指定的步驟數目後沒有出現錯誤縮減，則訓練會中止。指定一個大於 0 的整數。您也可以指定用於計算錯誤的資料樣本。**自動選擇**會使用測試樣本 (如果存在)，否則會使用訓練樣本。請注意，批次訓練可保證在每個資料傳輸後訓練樣本錯誤會縮減；因此只在測試樣本存在時此選項才適用於批次訓練。**訓練及檢定資料兩者**會為這些樣本檢查錯誤；只在測試樣本存在時才可以使用此選項。

附註：在每個完整資料傳輸後，線上和小型批次訓練需要進行額外的資料傳輸，以計算訓練錯誤。這個額外的資料傳輸會讓訓練明顯變慢，因此通常會建議您提供測試樣本，並選取不論何種情況下「**自動選擇**」。

- **最大訓練時間。** 選擇是否要指定演算法執行的最大分鐘數。指定一個大於 0 的數值。
- **最大訓練週期。** 允許的最大週期數 (資料傳輸)。如果超過最大週期數，訓練會中止。請指定一個大於 0 的整數。
- **訓練錯誤中的最小相對變更。** 如果與先前步驟相比時，訓練錯誤中的相對變更小於準則值時，則訓練會中止。指定一個大於 0 的數值。如果是線上或小型批次訓練，只有在使用測試資料計算錯誤時才會忽略此準則。
- **訓練錯誤比例中的最小相對變更。** 如果與虛無模型錯誤率相比，訓練錯誤率小於準則值時，則訓練會中止。虛無模型會預測所有因變數的平均值。指定一個大於 0 的數值。如果是線上或小型批次訓練，只有在使用測試資料計算錯誤時才會忽略此準則。

要儲存在記憶體中的最大觀察值數目。 此項目可控制下列多層感知器演算法中的設定。請指定一個大於 1 的整數。

- 在自動架構選項中，用來判斷網路架構的樣本大小是 $\min(1000, memsize)$ ，其中 $memsize$ 是可儲存在記憶體中的最大觀察值數目。
- 在可自動計算小型批次數目的小型批次訓練中，小型批次的數目是 $\min(\max(M/10, 2), memsize)$ ，其中 M 是訓練樣本中的觀察值數目。

半徑式函數

「半徑式函數」(RBF) 程序會依據預測變數值，為一個或多個依 (目標) 變數產生預測模型。

範例。 某電信公司根據服務使用方式來切割客戶數量，並將客戶分成四個組別。使用人口資料來預測組別成員的 RBF 網路可讓公司替個別準客戶自訂報價。

資料考量

因變數。 因變數可以是：

- **名義。** 當變數值代表實質上並未等級化的種類時 (例如，有員工工作的公司部門)，則此變數可視為名義。名義變數的範例包括區域、郵遞區號及宗教團體。
- **序數。** 當變數值代表實質上已等級化的種類時 (例如，服務滿意度從非常不滿意到非常滿意分級)，則此變數可視為序數。序數變數的範例包括代表滿意度或信賴程度的態度分數以及偏好等級分數。
- **尺度。** 如果某一變數可視為尺度 (連續)，表示它的值代表含有實際意義矩陣的已排序種類，因此適合比較值之間的距離。尺度變數的範例包括以年份表示的年齡及以千元為單位的收入。

本程序假設已指定給所有因變數適當的測量層級，但您可以在來源變數清單的變數上按一下滑鼠右鍵，並選取蹦現功能表上的測量層級，暫時變更變數的測量層級。

變數清單中各變數旁的圖示可識別測量層級和資料類型：

預測值變數。 可指定預測值為因素（類別）或共變量（尺度）。

類別變數編碼。 本程序在整個程序期間，會使用 one-of-c 編碼來暫時記錄類別預測變數與因變數。如果一個變數有多個 c 類別，則變數會儲存為 c 向量，其中第一個類別標示為 (1,0,...,0)，第二個類別標示為 (0,1,0,...,0)，...，最後一個類別標示為 (0,0,...,0,1)。

此編碼方法會增加加權鍵結值的數目，這會導致訓練變慢；但是，更多「精簡」的編碼方法通常會產生不適合的神經網路。如果您的網路訓練進行的非常慢，您可以將類似的類別組合在一起，或捨棄具有極少類別的觀察值，以嘗試減少類別預測值中的類別個數。

所有的 one-of-c 編碼都是以訓練資料為基礎，即使已定義測試或保留樣本也是如此（請參閱第 11 頁的『分割』）。因此，如果測試或保留樣本所包含的觀察值之預測值類別不在訓練資料中，則這些觀察值不會用於程序或評定中。如果測試或保留樣本所包含的個案之因變數類別不在訓練資料中，則這些個案不會用於程序，但可能用於評分中。

調整。 尺度因變數和共變量會依預設進行調整，以改善網路訓練。所有的調整都是以訓練資料為基礎，即使已定義測試或保留樣本也是如此（請參閱第 11 頁的『分割』）。也就是視調整的類型而定，僅使用訓練資料來計算平均數、標準差、共變量或因變數的最小值或最大值。如果您指定變數來定義分割，很重要的是這些共變量或因變數必須在訓練樣本、測試樣本和保留樣本之間有類似的分配。

次數加權。 此程序會忽略次數加權。

複製結果。 如果您要精確地複製結果，除了使用相同的程序設定外，請為亂數產生器使用相同的初始化值和相同的資料順序。此問題的詳細資料如下：

- **亂數產生器。** 在隨機指派分割期間，此程序會使用亂數產生器。未來若要重新產生相同的隨機化結果，請先使用與亂數產生器的相同初始化值，再執行每個「半徑式函數」程序。請參閱以取得逐步的指示。
- **觀察值順序。** 結果也會依存於資料順序，因為使用了 TwoStep 集群演算法來判斷半徑式函數。

若要將順序效應降到最低，請以隨機方式排列觀察值。若要驗證某個解決方案的穩定性，您也許會想要取得幾種不同的解決方案，其觀察值皆以不同的隨機順序排列。在檔案極大的情況下，可進行多次運算，以不同的隨機順序排列一個觀察值的樣本。

建立半徑式函數網路

在功能表上，選擇：

分析 > 神經網路 > 半徑式函數...

1. 選取至少一個因變數。
2. 選取至少一個因素或共變量。

您可以選擇性地在「變數」標籤上變更調整共變量的方法。選項為：

- **標準化。** 減去平均數，然後除以標準差 $(x - \text{mean})/s$ 。
- **常態化。** 減去最小值，然後除以範圍 $(x - \text{min})/(\text{max} - \text{min})$ 。常態化的值介於 0 和 1 之間。
- **調整後常態化。** 減去最小值，然後除以範圍的調整版本 $[2*(x - \text{min})/(\text{max} - \text{min})] - 1$ 。調整後常態化的值介於 -1 和 1 之間。
- **無。** 沒有調整共變量。

具有未知測量層級的欄位

當資料集中一或多個變數（欄位）的測量層級不明時，就會顯示「測量層級」警示。由於測量層級會影響此程序的結果計算，因此所有變數皆必須具有已定義的測量層級。

掃描資料

讀取作用中資料集的資料，並且針對目前具有未知測量層級的任何欄位指派預設的測量層級。若為大型資料集，則讀取時可能需要一些時間。

手動指派

列出測量層級不明的所有欄位。您可以將測量層級指派給那些欄位。您也可以在「資料編輯器」的「變數清單」窗格中指派測量層級。

由於測量層級是此程序的重要項目，因此您在所有欄位皆擁有已定義的測量層級之前，無法執行此程序。

分割

分割區資料集。 此組別指定將作用中資料集區隔成訓練、檢定和保留樣本的方法。**訓練樣本**由用來訓練神經網路的資料記錄所組成；在資料集中有某些比例的觀察值必須指定為訓練樣本以取得模式。**檢定樣本**是一組獨立的資料記錄，用來追蹤訓練期間的錯誤以避免訓練過度。建議您建立訓練樣本，且當檢定樣本小於訓練樣本時，一般來說網路訓練更加高效。**保留樣本**是另一個獨立的資料記錄集，用來存取最終神經網路；由於保留觀察值並未用來建置模型，保留樣本的錯誤為模型的預測能力提供「誠實」的估計。

- **根據觀察值的相對數字隨機指派觀察值。** 指定隨機指派給每一個樣本（訓練、檢定和保留）的觀察值的相對數字（比例）。% 欄報告根據您所指定的相對數字，將要指派給每一個樣本的觀察值的百分比。

例如，指定 7、3、0 作為訓練、檢定和保留樣本的相對數字，對應於 70%、30% 和 0%。指定 2、1、1 作為相對數字對應於 50%、25% 和 25%；1、1、1 對應於將資料集劃分為訓練、檢定和保留三部分相等。

- **使用分割變數來指定觀察值。** 指定一數值變數將作用中資料集的每一個觀察值指定給訓練、檢定或保留樣本。將變數中含有正值的觀察值指定給訓練樣本，將值為 0 的觀察值指定給檢定樣本，或將值為負數指定給保留樣本。含有系統遺漏值的觀察值會從分析中排除。任何分割變數的使用者遺漏值永遠視為有效。

架構

「架構」標籤是用來指定網路的架構。程序會建立包含一個隱藏「半徑式函數」階層的神經網路；一般而言，不需要變更這些設定。

隱藏階層的單位數。 選擇隱藏單位數的方法有三種。

1. **在自動計算的範圍內搜尋最佳單位數。** 程序會自動計算範圍的最小值和最大值，並且在該範圍內搜尋最佳的隱藏單位數。

如果定義了檢定樣本，則程序會使用檢定資料準則：最佳隱藏單位數目是測試資料中產生錯誤最少的數目。如果未定義檢定樣本，則程序會使用 Bayesian 資訊準則 (BIC)：最佳隱藏單位數目是訓練資料中產生最小 BIC 的數目。

2. **在指定範圍內搜尋最佳單位數。** 您可以提供自己的範圍，且程序將在該範圍內搜尋「最佳」的隱藏單位數。與先前的作法相同，範圍中最佳的隱藏單位數是使用測試資料準則或 BIC 來決定。
3. **使用指定的單位數。** 您可以覆寫使用的範圍，並直接指定特定的單位數。

隱藏階層的啟動函數。 隱藏階層的啟動函數為半徑式函數，其會將階層中的單位「連結」至後續階層中的單位數值。對輸出階層而言，啟動函數為識別函數；因此輸出單位僅會是隱藏單元的加權總和。

- **常態化半徑式函數。** 使用 Softmax 啟動函數，以讓所有隱藏單位的啟動常態化成總和為 1。
- **一般半徑式函數。** 使用指數啟動函數，以讓隱藏單位的啟動成為 Gaussian 突出物般的輸入函數。

隱藏單位中的重疊。重疊因素是套用至半徑式函數寬度的乘數。自動計算的重疊因素數值是 $1+0.1d$ ，其中 d 是輸入單位數（所有因素的類別數目與共變量數目的總和）。

輸出

網路架構。 顯示有關神經網路的摘要資訊。

- **說明。** 顯示有關神經網路的資訊，包括因變數、輸入和輸出單位的數目、隱藏階層和單位的數目，以及啟動函數。
- **圖。** 以無法編輯的圖表來顯示網路結構圖。請注意，隨著共變量和因素層級的數目增加，結構圖也會越來越難解讀。
- **加權鍵結值。** 顯示係數估計值，此估計值會顯示已知階層中的單位和下一階層中的單位之間的關係。即使將作用中資料集分割成訓練、測試和保留資料，加權鍵結值仍會以訓練樣本為基礎。請注意，加權鍵結值的數目可以變得相當大，因此這些加權一般不適用於解讀網路結果。

網路效能。 顯示用於判斷模式是否「良好」的結果。附註：此組別中的圖表是以合併訓練和測試樣本為基礎，或是如果沒有測試樣本時，僅以訓練樣本為基礎。

- **模型摘要。** 以分割和整體方式顯示神經網路結果的摘要，包括錯誤、不正確預測的相對錯誤或百分比和訓練時間。

錯誤為平方和誤差。此外，根據因變數測量層級來顯示不正確預測的相對錯誤或百分比。如果任一因變數具有尺度測量層級，則會顯示平均整體相對錯誤（相對於平均數模式）。如果所有的因變數為類別，則會顯示不正確預測的平均百分比。也會顯示個別因變數之不正確預測的相對錯誤或百分比。

- **分類結果。** 顯示每個類別因變數的分類表。每個表會提供每個因變數類別之正確或不正確分類的觀察值數目。也會報告已正確分類之總觀察值的百分比。
- **ROC 曲線。** 顯示每個類別因變數的 ROC（接收器作業特性）曲線。也會顯示可提供每個曲線下之區域的表格。對一個特定的因變數來說，ROC 會為每個類別顯示一個曲線。如果因變數有兩種類別，則每個曲線會視討論中的類別為正向狀態和另一個類別進行比較。如果因變數有兩種以上的類別，則每個曲線會視討論中的類別為正向狀態和其他的類別整合進行比較。
- **累積收益圖。** 顯示每個類別因變數的累積收益圖。為每個因變數類別顯示一條曲線的方法和 ROC 曲線的方法相同。
- **增益圖。** 顯示每個類別因變數的增益圖。為每個因變數類別顯示一條曲線的方法和 ROC 曲線的方法相同。
- **觀察圖表的預測。** 顯示每個因變數的依觀察值預測圖表。如果是類別因變數，則會顯示每個回應值類別之預測虛擬機率的集群盒形圖，其中以觀察回應值類別作為集群變數。如果是尺度因變數，則會顯示散佈圖。
- **預測圖表的殘差。** 顯示每個尺度因變數的依預測值殘差圖表。殘差與預測值之間應該沒有可見的樣式。僅為尺度因變數產生此圖表。

觀察值處理摘要。 顯示觀察值處理摘要表，其中摘要出分析中包含和排除的觀察值數、總數，以及是依訓練、測試和保留樣本包含和排除。

自變數重要性分析。 執行敏感度分析，此分析會計算在決定神經網路時每個預測值的重要性。分析是以合併訓練和測試樣本為基礎，或是如果沒有測試樣本時，僅以訓練樣本為基礎。這會建立可顯示每個預測值之重要性和常態化重要性的表格和圖表。請注意，如果有大量的預測值或觀察值，則敏感度分析的需要大量計算且耗時。

儲存

「儲存」標籤是用來將預測值儲存為資料集中的變數。

- **儲存每個因變數的預測值或類別。** 此動作會儲存尺度因變數的預測值和類別因變數的預測類別。

- **儲存每個因變數的預測虛擬機率。** 此動作會儲存類別因變數的預測虛擬機率。對於前 n 個類別，系統會為每個類別儲存個別的變數，其中 n 是在「要儲存的類別」行中指定。

已儲存變數的名稱。 自動名稱產生會確保您能保留所有的工作。自訂名稱可讓您捨棄或取代先前執行的結果，而不必先刪除在「資料編輯器」中儲存的變數。

機率和虛擬機率

無法將預測虛擬機率解讀為機率，因為「半徑式函數」程序使用平方和誤差與輸出階層的識別啟動函數。程序會儲存這些預測虛擬機率，即使任一機率小於 0 或大於 1，或是特定的因變數總和不是 1。

ROC、累積收益圖和增益圖（請參閱 第 12 頁的『輸出』）是根據虛擬機率而建立。當任一虛擬機率小於 0 或大於 1，或是特定變數的總和不是 1 時，首先會將機率調整在 0 和 1 之間，且總和為 1。虛擬機率是透過除以其總和來進行調整。例如，如果觀察值具有三類別因變數之 0.50、0.60 和 0.40 的預測虛擬機率，則每個虛擬機率會除以總和 1.50 以得到 0.33、0.40 和 0.27。

如果任一虛擬機率為負數，則在進行上述的調整前，會將最低的絕對值增加到所有虛擬機率。例如，如果虛擬機率為 0.30、.50 和 1.30，則首先將 0.30 加到每個值以取得 0.00、0.80 和 1.60。接著，將每個新數值除以總和 2.40 以取得 0.00、0.33 和 0.67。

匯出

「匯出」標籤是用來將每個因變數的加權鍵結估計值儲存到 XML (PMML) 檔案。您可以使用這個模型檔案，將模式資訊套用到其他資料檔中以進行評分工作。如果已定義分割檔，則此選項無法使用。

選項

使用者遺漏值。 因素必須有有效值，以將觀察值納入分析。這些控制項可讓您決定是否要在因素和類別因變數中，將使用者遺漏值視為有效值。

注意事項

本資訊係針對 IBM 在美國所提供之產品與服務所開發。IBM 可能會以其他語言提供本資料。但是，您可能需要具有該語言的產品或產品版本，才能存取該產品。

IBM 可能並未在其他國家提供在本文件中討論到的產品、服務或功能。有關目前在 貴地區可供使用的產品與服務相關資訊，請洽您當地的 IBM 服務代表。對於 IBM 產品、程式或服務的任何參考，目的並不是要陳述或暗示只能使用 IBM 產品、程式或服務。任何功能相等且未侵犯 IBM 智慧財產權的產品、程式或服務皆可使用。但是，評估及確認任何非 IBM 產品、程式或服務的操作之責任應由使用者承擔。

IBM 可能有一些擁有專利或專利申請中的項目包含本文件所描述的內容。本文件的提供並不表示授與您對於這些專利的權利。您可以將書面的授權查詢寄至：

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

對於與雙位元組 (DBCS) 資訊相關的授權查詢，請與貴國的 IBM 智慧財產部門聯絡，或將查詢郵寄至：

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

International Business Machines Corporation 只依「現況」提供本出版品，不提供任何明示或默示之保證，其中包括且不限於不侵權、可商用性或特定目的之適用性的隱含保證。有些地區不允許特定交易中明示或默示的保固聲明，因此，此聲明或許對您不適用。

此資訊內容可能包含技術失準或排版印刷錯誤。此處資訊會定期變更，這些變更將會納入新版的聲明中。IBM 可能會隨時改善和 / 或變更此聲明中所述的產品和 / 或程式，恕不另行通知。

本資訊中任何對非 IBM 網站的敘述僅供參考，IBM 對該網站並不提供任何保證。該「網站」的內容並非此 IBM 產品的部分內容，使用該「網站」需自行承擔風險。

IBM 可能會以任何其認為適當的方式使用或散佈您提供的任何資訊，無需對您負責。

意欲針對達成以下目的而擁有本程式相關資訊之程式被授權人：(i) 在獨立建立的程式與其他程式 (包括本程式) 之間交換資訊及 (ii) 共用已交換的資訊，應聯絡：

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

在適當條款與條件之下，包括某些情況下 (支付費用)，或可使用此類資訊。

在本文件中描述的授權程式及其適用之所有授權材料皆由 IBM 在與我方簽訂之 IBM 客戶合約、IBM 國際程式授權合約或任何相等等效合約中提供。

本文件中引用的效能資料及用戶範例僅供敘述之目的。特定配置及作業條件下的實際效能結果可能不同。

本文件所提及之非 IBM 產品資訊，取自產品的供應商，或其發佈的聲明或其他公開管道。IBM 並未測試過這些產品，也無法確認這些非 IBM 產品的執行效能、相容性或任何對產品的其他主張是否完全無誤。有關非 IBM 產品的功能問題應直接洽詢該產品供應商。

關於 IBM 未來方針或意圖的所有聲明僅代表目標或目的，得依規定未另行通知即變更或撤銷。

此資訊包含用於日常企業運作的資料和報表範例。為了儘可能提供完整說明，範例中包含了人名、公司名稱、品牌名稱和產品名稱。這些名稱全為虛構，如與實際人員或企業之名稱有所雷同，純屬巧合。

著作權授權：

本資訊含有原始語言之範例應用程式，用以說明各作業平台中之程式設計技術。貴客戶可以為了研發、使用、銷售或散布符合範例應用程式所適用的作業平台之應用程式介面的應用程式，以任何形式複製、修改及散布這些範例程式，不必向 IBM 付費。這些範例並未在所有情況下完整測試。故 IBM 不保證或默示保證這些樣本程式之可靠性、服務性或功能。這些程式範例以「現狀」提供，且無任何保證。IBM 對因使用這些程式範例而產生的任何損害概不負責。

這些範例程式或任何衍生成果的每份複本或任何部分，都必須依照下列方式併入著作權聲明：

© IBM 2019. 本程式之若干部分係衍生自 IBM 公司的範例程式。

© Copyright IBM Corp. 1989 - 20019. All rights reserved.

商標

IBM、IBM 標誌及 ibm.com 是 International Business Machines Corp. 在世界許多管轄區註冊的商標或註冊商標。其他產品及服務名稱可能是 IBM 或其他公司的商標。IBM 商標的最新清單可在 Web 的 "Copyright and trademark information" 中找到，網址為 www.ibm.com/legal/copytrade.shtml。

Adobe、Adobe 標誌、PostScript 以及 PostScript 標誌為 Adobe Systems Incorporated 於美國和 / 或其他國家的註冊商標或商標。

Intel、Intel 標誌、Intel Inside、Intel Inside 標誌、Intel Centrino、Intel Centrino 標誌、Celeron、Intel Xeon、Intel SpeedStep、Itanium 和 Pentium 為 Intel Corporation 或其分公司於美國和其他國家的商標或註冊商標。

Linux 為 Linus Torvalds 於美國和 / 或其他國家的註冊商標。

Microsoft、Windows、Windows NT 和 Windows 標誌為 Microsoft Corporation 於美國和 / 或其他國家的商標。

UNIX 為 The Open Group 於美國和其他國家的註冊商標。

Java 和所有以 Java 為基礎的商標及標誌是 Oracle 及（或）其子公司的商標或註冊商標。

索引

索引順序以中文字，英文字，及特殊符號之次序排列。

〔三劃〕

小型批次訓練
在多層感知器中 6

〔四劃〕

中止規則
在多層感知器中 8

〔五劃〕

半徑式函數 9
分割 11
網路架構 11
模型匯出 13
輸出 12
選項 13
儲存變數至作用中資料集 12

〔六劃〕

多層感知器 3
分割 4
訓練 6
網路架構 5
模型匯出 8
輸出 7
選項 8
儲存變數至作用中資料集 8
收益圖
半徑式函數中 12
在多層感知器中 7

〔七劃〕

批次訓練
在多層感知器中 6
保留樣本
半徑式函數中 11
在多層感知器中 4

〔九劃〕

架構
神經網路 2

〔十劃〕

神經網路
架構 2
訓練樣本
半徑式函數中 11
在多層感知器中 4

〔十一劃〕

啟動函數
半徑式函數中 11
在多層感知器中 5

〔十二劃〕

測試樣本
半徑式函數中 11
在多層感知器中 4

〔十四劃〕

網路架構
半徑式函數中 11
在多層感知器中 5
網路訓練
在多層感知器中 6
網路結構圖
半徑式函數中 12
在多層感知器中 7

〔十五劃〕

增益圖
半徑式函數中 12
在多層感知器中 7
線上訓練
在多層感知器中 6

〔十六劃〕

輸出階層
半徑式函數中 11
在多層感知器中 5
遺漏值
在多層感知器中 8

〔十七劃〕

隱藏階層
半徑式函數中 11
在多層感知器中 5

R

ROC 曲線(O)
半徑式函數中 12
在多層感知器中 7



Printed in Taiwan