**AI-FML代理人區域學習機制應用於圍棋機器人**

|  |  |
| --- | --- |
| 蔡易霖  國立臺南大學  資訊工程學系  m10759011@gm2.nutn.edu.tw | 李健興 國立臺南大學  資訊工程學系  leecs@mail.nutn.edu.tw |

# 摘要

在本研究中，我們提出了一個具有區域學習機制的AI-FML代理人應用於圍棋機器人。AI-FML代理人具有三種智慧應用於機器人，包括感知智能、認知智能和計算智能。此外我們將區域學習機制嵌入到AI-FML代理人中。執行區域學習的方法包括三個步驟。首先訓練初始全域模型，接著為每個已識別的模型進行區域訓練，最後使用剩下的訓練資料更新全域模型。本篇論文採用Google DeepMind Master 60 盤圍棋做為訓練集和測試集。從實驗結果可以發現具有區域學習機制的AI-FML代理人可以提高圍棋機器人在回歸模型上的性能。

**關鍵詞**：AI-FML、代理人、區域學習、模糊機器學習、圍棋。

## 前言

如今，機器學習已在現實生活中的許多領域和行業中被使用，例如醫學診斷、影像處理、回歸…等。但是訓練高效能的機器學習模型通常是一個迭代的過程，它依賴於經驗和嘗試錯誤。有時當模型表現不令人滿意時，我們需要採取一些補救措施來增強其性能[1]。我們可以使用一些策略來增強機器學習的性能，例如使用單個多神經元的模型、單個多層網路的模型、並行連接多個簡單基底模型或串聯連多個簡單的基底模型[1、2]。 在本文中，我們提出了一種具有區域學習機制的AI-FML代理，以提高機器學習的性能。Mendel所提出的區域學習想法如下：「設想一位雕塑家正在雕刻一個人物，在他第一次雕刻後，他會檢查整個雕像，並且注意需要對哪些地方進行改進。他會將重點擺到需要被改善的部分，接著將雕像的精緻部分與雕像的其餘部分結合在一起。持續不斷的修補，直到對整個雕像感到滿意為止。」區域學習中的每個區域都類似於雕塑品中需要更多補修的部分[1]。根據[1]，區域學習包括三個步驟，描述如下：(1)首先使用所有訓練數據訓練初始全域模型。(2)從初始的全域模型中找出區域，並為每個區域訓練一個區域模型。(3)使用訓練集更新全域模型。Wu和Jerry [1]將區域定義為輸入域中的連接多面體。例如，一維輸入域中是一個直線區段，而二維輸入域可以是一個矩形，或是一個橢圓形…等。但是一般來說，尋找區域的位置是一項非常艱鉅的任務，並非每個問題都可以輕鬆實現。區域學習並行和串聯連接多個簡單模型，以提高學習性能。Mendel在論文中[2]介紹如何使用基於規則的模糊系統執行區域學習，因為在模糊系統中易於初始化區域的位置。

對於數據科學家來說，回歸、集成式學習和深度學習是重要的機器學習方法[9]。Jang [11]在1993年提出了一種基於自適應網絡的模糊推論系統(ANFIS)。ANFIS整合了機器學習模型和模糊邏輯原理，並且在實際應用中取得兩者的好處。它的推論系統對應於一組模糊規則，這些規則具有學習能力以近似非線性函數，並被認為是一種通用的評估器[11]。在我們之前的論文[3]中，我們使用AlphaGo Master [14] 60盤棋局作為實驗數據集，並基於FML基因學習(GFML)、XGBoost集成式學習和七層深度模糊神經網絡(DFNN)訓練預測勝率，以使Darkforest AI 機器人 [12]預測的勝率更接近ELF Open Go AI機器人 [13]。在本文中，我們進一步將FML、區域學習和基於自適應網絡的模糊推理系統(ANFIS)與深度學習相結合，以訓練模型來預測AlphaGo Master 60場比賽的勝率。實驗結果顯示，引入區域學習機制比我們在以前的論文中所提出的方法具有更好的性能[3]。

本論文架構如下：第2節介紹了基於ANFIS的區域學習機制的結構。第3節介紹了圍棋資料集基於ANFIS的區域學習機制。第4節提出了一種基於區域學習的AI-FML代理人，應用用於機器人，以及預測圍棋比賽的勝率。第5節為實驗結果。最後，第6節為本論文之結論。

### 基於ANFIS區域學習機制架構

* 1. **區域學習機制**

在傳統ANFIS的設計階段，使用訓練資料來優化輸入的隸屬函數與輸入參數，並使用所有訓練資料來優化評估指標[1、11]。基於文獻[1]中區域學習的概念，本論文提出了一種具有區域學習機制的AI-FML代理人，從全域的模糊規則系統設計開始，然後找到評估指標最大的區域作為區域。圖1顯示了用於預測Master 60 盤棋局勝率的區域學習架構，其描述如下：(1)我們使用所有訓練資料來訓練全域模型。(2)我們找出第1手(M1)到P1手(MP1)的輸入區域會引起較大的損失，因此我們使用屬於該區間的訓練資料來訓練一個區域模型(PM1)以減少整體學習損失。接著我們採用第P1+1 (MP1+1)手到第P2 (MP2)手, …,第N (MN)手到第PN (MPN)手落於區間內的資料來訓練區域模型1 (PM1), 區域模型 2 (PM2), …, 區域模型 L (PML) (3)最後再使用區域1, 2, …, L未使用的剩餘訓練資料更新全域的模糊規則系統。​​圖1以第2盤棋局為例，假設L = 3，MP1 = 20，MP2 = 40和MP3 = 90。



**圖1. 預測Master 60 盤棋局勝率的區域學習架構**

* 1. **基於自適應網絡的模糊推理系統機制**

基於ANFIS的區域學習機制已嵌入到AI-FML代理人中，我們將其應用於預測圍棋資料的勝率。另外採用ANFIS算法訓練區域模型以提高性能指標[1，11]。 ANFIS是基於Takagi–Sugeno模糊推論系統的人工神經網絡模型之一，並於1990年初被提出[10，11]。在本論文中，我們使用PyTorch來實作基於ANFIS的區域學習模型框架。ANFIS的架構包括前鑑部和後鑑部兩部分，其架構由五層網路所組成，如圖2[11]。其描述如下: (1)第一層/模糊化層：我們使用隸屬函數來計算輸入值所對應的歸屬值。 (2)第二層/規則層: 負責通過乘積為輸入信號來生成規則的觸發強度。(3)第三層/正規化層：正規化觸發強度，將每個值除以所有的觸發強度總和。(4)第四層/推論層: 將第三層的正規化值與後鑑部參數相乘，並將結果發送到第五層。 (5)第五層/解模糊化層: 將所有輸出信號的總和計算，並生成最終輸出。



**圖2. 自適應網絡的模糊推理系統五層架構[11]**

1. **圍棋資料基於ANFIS區域學習機制**
   1. **IEEE WCCI 2020資料集簡介**

在本論文中，我們採用2016年12月和1月的AlphaGo Master 60盤棋局作為實驗數據，這些資料被指定為IEEE CIS旗艦會議上所舉行的競賽數據[3]。參賽者可以從60場比賽中選擇40場比賽作為訓練資料，將其餘20場比賽作為測試資料。並建構模糊推論系統的知識庫和規則庫。表1為第1盤棋局的第1步到第20步的資料。每個棋局都包含Darkforest AI Bot和EFL OpenGo AI Bot預測的數據。MoveNo是手數編號，但MoveNo僅列出「奇數」編號(即1、3、5 ...)，因為每一行對應一對黑棋與白棋。因此MoveNo 1的行對應於黑色的第1手(即B1)和白色的第1手(即W2)。MoveNo 145對應於黑色第145手(即B145)和白色第146手(即W146)。如果最後的MoveNo為「奇數」，則最後一行的白色資訊將為空。訓練資料從第1局到第40局共有3758筆數據。測試資料從第41局到第60局共有1880筆數據。

**表1. 第1局第1步到第20步圍棋數據**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Move  No | DBSN | DWSN | DBWR | DWWR | DBTMR | DWTMR | EBWR | EWWR |
| 1 | 3863 | 2274 | 0.52 | 0.48 | 0 | 1 | 0.49 | 0.50 |
| 3 | 9283 | 7866 | 0.51 | 0.48 | 0.5 | 1 | 0.45 | 0.54 |
| 5 | 11395 | 6798 | 0.51 | 0.47 | 0.66 | 1 | 0.45 | 0.57 |
| 7 | 4499 | 10703 | 0.51 | 0.46 | 0.75 | 1 | 0.49 | 0.52 |
| 9 | 7388 | 20017 | 0.52 | 0.46 | 0.8 | 1 | 0.47 | 0.53 |
| 11 | 20098 | 9693 | 0.53 | 0.46 | 0.83 | 1 | 0.47 | 0.52 |
| 13 | 20017 | 14595 | 0.53 | 0.47 | 0.85 | 1 | 0.48 | 0.49 |
| 15 | 6786 | 4892 | 0.52 | 0.45 | 0.87 | 0.875 | 0.48 | 0.53 |
| 17 | 20017 | 6432 | 0.53 | 0.46 | 0.88 | 0.7778 | 0.48 | 0.53 |
| 19 | 5267 | 13267 | 0.51 | 0.48 | 0.9 | 0.8 | 0.48 | 0.50 |
| 說明 | | | | | | | | |
| (1) *DBSN*和*DWSN* 表示黑森林系統黑與白的模擬次數  (2) *DBWR*和*DWWR* 表示黑森林系統黑與白的預測勝率  (3) *DBTMR*和*DWTMR* 表示黑森林系統黑與白的匹配率  (4) EBWR和EWWR表示ELF OpenGo系統黑與白的預測勝率 | | | | | | | | |

* 1. **圍棋資料分區前處理**

Jang [11]在1993年論文中提到：「模糊條件語句是IF A THEN B形式的表達式，其中A和B是具有適當的隸屬函數的模糊集的標籤」和「fuzzy if-then規則通常用於記錄了不精確的推論模式，這些模式對人類在不確定和不精確的環境中做出決策扮演著重要的角色」。

有多種用於劃分輸入域的分區函數，例如，明確分區函數或Type-1分區函數[2]。實值變數的明確分區由多個不重疊的相鄰區域所組成，這些區域是實數的間隔，其中每個區域中的隸屬程度為1，在該區域之外為0。此外實值變數的一階不確定性分區由多個重疊區間組成，其中我們可以確定重疊的開始和結束位置，因此每個重疊中的歸屬程度是介於0到1之間。一階不確定性分區被視為與其最近鄰域重疊的非矩形型Type-1分區函數。如圖3本實驗採用64個高斯函數的Type-1分區，用於圍棋數據預處理。



**圖3. 64個高斯函數的Type-1分區函數**

此外實驗中有兩個用於資料預處理的標準。第一個標準是「如果Darkforest的最終預測結果是錯誤的，則該局的資料集將不會用於訓練集或測試集。」例如我們在訓練集中將不採用第15盤、第25盤和第29盤的資料集。

第二個標準是「如果ELF OpenGo和Darkforest之間的白色或黑色預測勝負差異超過40％，則該局的資料將不會在訓練集或測試集中採用。」 假設一盤對局中有100手，但ELF OpenGo和Darkforest預測有40個手數不同，則該局將從資料集中刪除。基於此標準，我們將第8盤、第32盤和第36盤從訓練數據集中刪除。在實驗測試集中我們刪除了第44盤、第47盤、第56盤和第57盤。 此外，我們還檢查資料集中的缺失值。如果有一局的第1手與最後一手都是相同的黑棋或白棋，我們會將該局的最後一筆資料移除。經過檢查後，我們從訓練集中刪除了19筆資料，從測試集中刪除了6筆資料。最後我們在此實驗中保留了3143筆訓練資料，測試集中保留1239筆資料。

### 基於區域學習AI-FML機器人圍棋應用

* 1. **ANFIS於AI-FML代理人圍棋應用**

自適應神經模糊推理系統（ANFIS）架構是流行於早期模糊系統中所使用的方法。在本論文中我們將ANFIS使用Pytorch框架進行實作。PyTorch是Facebook AI研究團隊發布的一個深度學習框架，它能夠實現張量計算和動態神經網絡，以及自動計算每個張量中的梯度微分。並且具有很好的靈活性與易用性，因而廣受機器學習研究者歡迎。此外它還提供神經網路、優化器與損失函數的函式庫。在實驗中我們採用Python 3.7.3、Anaconda 4.7.12 (發行版)、PyTorch 1.5.0+cpu。硬體使用Intel Xeon E5-2667 v4 3.20 GH處理器、16 GB的DDR4記憶體並運行於Ubuntu 16.04.5 LTS (Xenial Xerus)作業系統。基於區域學習的AI-FML代理人中的模糊規則核心系統是ANFIS，它由以下功能所組成：(1)含有數據的知識庫。(2)規則庫。(3)規則推論的運算子。(4)模糊化過程。(5)去模糊化過程[11]。圖4顯示了機器人應用於AI-FML代理人的ANFIS架構，該架構透過以下步驟建立：(1)在前鑑部使用隸屬函數檢索模糊變數的輸入值，以計算每個語意的隸屬程度。(2)結合前鑑部的隸屬程度以獲取每個規則的權重。(3)根據權重產生每個規則的後鑑部。(4)匯總結果以產生最終輸出[ 11]。人類輸入資料，AI-FML代理人將解模糊後的值輸出給機器人，例如Kebbi Air，Palro或Zenbo，以實現人類和機器人在教室中學習的目標[3]。



**圖4. ANFIS於AI-FML代理人架構**

在本研究中，我們使用64個Type-1高斯分區函數對六個輸入模糊變數進行分區，包括DBSN、DWSN、DBWR、DWWR、DBTMR和DWTMR。舉例來說，如果*x*是AI-FML代理人區域學習的輸入值DBSN，有64條規則，如下所示：

規則1：IF *x*為分區1，Then *y* = *y*1 (*x*)

規則2：IF *x*為分區2，Then *y* = *y*2 (*x*)

⋮

規則64：IF *x*為分區64，Then *y* = *y*64 (*x*)

其中*y*1(*x*)，*y*2(*x*)，…，*y*64(*x*)是*x*的不同函數。在分區 P(1|*x*) 中，假設只有規則1被觸發，因此模糊系統輸出為y = y1(*x*)。另外在分區 P(2|*x*) 中，如果規則1和規則2均被觸發，因此模糊系統的輸出是 *y*1(*x*) 和 *y*2(*x*) 在分區1和2的隸屬程度加權平均值。

* 1. **圍棋資料基於區域學習AI-FML代理人**

基於區域學習的AI-FML代理人可以將1-Patch、2-Patch和3-Patch應用於電腦圍棋資料集。圖5(a)是1-Patch的學習模型，此模型僅有一個Patch專門來預測第1手到第40手。其餘的手數交由全域模型進行預測。圖5(b)是2-Patch的學習模型，此模型經由兩個區域進行訓練。在此模型中我們將前40手切成兩等份，分別為第1手到第20手，以及第21手到第40手。其餘的手數交由全域模型進行預測。圖5(c)是3-Patch的學習模型，區域1與2分別為第1手到第20手，以及第21手到第40手。區域3為第41手到第90手。其餘的手數交由全域模型進行預測。

|  |
| --- |
| (a) |
| (b) |
| (c)  **圖5. 區域學習模型架構 (a) 1-Patch (b) 2-Patch (c) 3-patch** |

### 實驗結果

* 1. **實驗1: AI-FML代理人ANFIS全域學習**

在實驗1中，我們採用具有Adam優化器的ANFIS和學習速率為0.001的梯度下降法來更新神經網絡中的所有參數。表2說明了在訓練2000個迭代後使用平均絕對誤差(MAE)、均方誤差(MSE)和均方根誤差(RMSE)的評估指標數值。這種方法的性能比我們先前論文[3]中使用Adam優化器的DNN機器學習的方法還來的更好。此外實驗 1減少了學習參數量和學習時間。圖6顯示了實驗1中第39局和第58局的回歸分析。虛線標記的曲線為EBWR(AL)/ EWWR(AL)是實驗1的實驗結果。圖中的虛線和實線分別是Darkforest和ELF OpenGo的預測勝率，黑色和紅色曲線分別代表黑棋和白棋。如圖6(a)所示，虛線曲線雖然有趨近於ELF OpenGo所預測的勝率，但前5手和第56附近的手數仍具有較大的方差。從圖6(b)可以發現，除了前40手以外，虛線曲線的預測趨勢與實線類似。由於上述這些情況，我們在實驗2中實現了區域學習的概念。

**表2. 基於MAE、MSE和RMSE的損失評估**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 訓練集 | 驗證集 | 測試集 |
| MAE | 0.0683 | 0.0876 | 0.1197 |
| MSE | 0.0158 | 0.0286 | 0.0438 |
| RMSE | 0.1258 | 0.1692 | 0.2093 |

|  |
| --- |
|  |
| (a) |
|  |
| (b) |

**圖6. 實驗1迴歸分析 (a) 第39盤 (b) 第58盤**

* 1. **實驗2: 3個區域的AI-FML代理人**

在實驗2中，我們採用具有三個區域機制的區域學習AI-FML代理人。我們按照以下方式劃分三個區域：區域1為第1手到第20手，區域2為第21手到第40手，以及區域3為第41手到第90手。首先我們先學習1000代的全域模型，然後再學習100代的區域1、區域2和區域3。圖7(a)、(b)、(c) 和 (d) 分別顯示了全域模型、區域1、區域2和區域3在訓練期間的訓練集和驗證集的歷史學習曲線。我們也可以觀察到圖7 (a)、(b)和(d)比圖7(c)有更好的擬合曲線。也許這是因為第20手之後，棋局處於中間階段，處於不確定的情況下，這使得在學習模型時更難適應實際情況。在實驗2中，我們設計了兩個子實驗，描述如下：(1)實驗2.1：我們僅使用經過訓練1000代的全域模型來預測訓練集。 (2)實驗2.2：我們使用訓練後的全域模型、區域1、區域2和區域3來預測訓練集，並根據每一手所相對應的預測模型。圖8顯示了每一手的平均損失曲線。橙色實線和紫色虛線是實驗2.1和實驗2.2的結果。損失值與實驗2.1相比，實驗2.2著重在區域1、區域2和區域3有下降的趨勢。

|  |
| --- |
| (a) |
| (b) |
| (c) |
| (d) |

**圖7. (a) 全域模型、(b) 區域 1、(c) 區域 2 、(d) 區域 3的學習曲線**



**圖8. 實驗2中每一手的MSE曲線**

### 結論

在本論文中我們展示了具有區域學習機制的AI-FML代理人應用於圍棋機器人。驗實驗中執行了各種區域學習模型，包括全域模型、一個區域、兩個區域與三個區域的模型。此外區域學習可以在機器人應用AI-FML代理人中被實現。我們採用Google DeepMind Master 60盤棋局作為訓練集和測試集。從實驗結果可以發現具有區域學習機制的AI-FML代理人可以提高圍棋機器人在回歸模型上的性能。

**參考文獻**

1. D. Wu and J. M. Mendel, “Patch learning,” *IEEE Transactions on Fuzzy Systems*, Jul. 2019. (DOI: 10.1109/TFUZZ.2019.2930022)
2. J. M. Mendel, “Adaptive variable-structure basis function expansions: Candidates for machine learning,” *Information Science*, vol. 496, pp. 124-149, 2019.
3. C. S. Lee, Y. L. Tsai, M. H. Wang, W. K. Kuan, Z. H. Ciou, and N. Kubota, “AI-FML agent for robotic game of Go and AIoT real-world co-learning applications,” *2020 World Congress on Computational Intelligence* (*IEEE WCCI 2020*), Glasgow, Scotland, UK, Jul. 19-24, 2020.
4. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, “Mastering the game of Go with deep neural networks and tree search,” *Nature*, vol. 529, pp. 484–489, Jan. 2016.
5. C. S. Lee, M. H. Wang, L. W. Ko, Y. Hsiu Lee, H. Ohashi, N. Kubota, Y. Nojima, and S. F. Su, “Human intelligence meets smart machine: a special event at the IEEE International Conference on Systems, Man, and Cybernetics 2018,” *IEEE Systems, Man, and Cybernetics Magazine*, vol. 6, no. 1, pp. 23-31, Jan. 2020.
6. C. S. Lee, M. H. Wang, Y. L. Tsai, L. W. Ko, B. Y. Tsai, P. H. Hung, L. A. Lin, and N. Kubota, “Intelligent agent for real-world applications on robotic edutainment and humanized co-learning,” *Journal of Ambient Intelligence and Humanized Computing*, 2019. (DOI: 10.1007/s12652-019-01454-4).
7. C. S. Lee, M. H. Wang, L. W. Ko, N. Kubota, L. A. Lin, S. Kitaoka, Y. T Wang, and S. F. Su, “Human and smart machine co-learning: brain-computer interaction at the 2017 IEEE International Conference on Systems, Man, and Cybernetics,” *IEEE Systems, Man, and Cybernetics Magazine*, vol. 4, no. 2, pp. 6-13, Apr. 2018.
8. C. S. Lee, M. H. Wang, S. C. Yang, P. H. Hung, S. W. Lin, N. Shuo, N. Kubota, C. H. Chou, P. C. Chou, and C. H. Kao, “FML-based dynamic assessment agent for human-machine cooperative system on game of Go,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 25, no. 5, pp. 677-705, 2017.
9. J. Castanon, “10 machine learning methods that every data scientist should know,” May 2019
10. 維基百科, 【自適應神經模糊推理系統】 <https://en.wikipedia.org/wiki/Adaptive_neuro_fuzzy_inference_system>.
11. J. S. R. Jang, “ANFIS: Adaptive-Network-based Fuzzy Inference Systems,” IEEE Transaction on Systems, Man, and Cybernetics, vol. 23, pp. 665-685, May 1993.
12. Y. Tian and Y. Zhu, “Better computer Go player with neural network and long-term prediction,” Feb. 2016.
13. Y. Tian, J. Ma, Q. Gong, S. Sengupta, Z. Chen, J. Pinkerton, and C. L. Zitnick, “ELF Open Go: An analysis and open reimplementation of AlphaZero,” May 2019.
14. DeepMind, AlphaGo Master series: 60 online games, Jan. 2020, <https://deepmind.com/alphago-master-series>.