Neuro-fuzzy networks for inference transparency

Thomas Sesmat[1,2], Jean-Loup Farges[1,*], Filipo Perotto[1], Gauthier Picard[1]
*farges@onera.fr

CentraleSupélec, Gif-sur-Yvette
- 2024 March 27th and 28th -

ONERA
THE FRENCH AEROSPACE LAB

[1]ONERA, [2]INSA

Neural net models

- Remarkable learning performance
  - classification
  - regression
- Lack of transparency of the inference process

Neuro-Fuzzy networks

- 1975: McCulloch-Pitts neuron with fuzzy output and use of a network with those neurons [1]
- Nineties: *Adaptive-Network-based Fuzzy Inference System* (ANFIS)[2]
- Since: a large number of approaches [3], variety of
  - architectures
  - fuzzyfication and defuzzyfication techniques
  - algorithms for training
- Efficient for classification and regression tasks:
  - high degree of accuracy
  - appropriate level of interpretability some areas

For applications needing a high development assurance level, level of interpretability of Neuro-Fuzzy approaches not high enough:

- verification of each parameter shall be possible
- consistence of logical inference shall be ensured:
  - conjunction and disjonction based on T-norm and associated T-conorm
  - no “If...then” fuzzy rules with abrupt consequences
- input space shall be covered

Objective : To contribute to improving neuro-fuzzy approches w.r.t. those issues

Zadeh's Fuzzy Logics

With Boolean connection weights

Conjunction: pseudo-minimum on $[0;1]$

$$\min_j \left\{ x_j w_j^{conj} + 1 - w_j^{conj} \right\}$$

$1 - w_j^{conj}$ term:

- 0 if jth membership degree is taken into account
- 1 if jth membership degree is not taken into account

Disjunction: actual maximum

$$\max_j \left\{ x_j w_j^{disj} \right\}$$

- x_j: jth membership degree
- $w_j^{conj}, w_j^{disj} \in \{0, 1\}$: weights for taking into account jth membership degree

Covering input space

Coupled sigmoïds

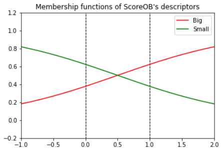
Meaningful linguistic descriptor $\Longrightarrow$ corresponding membership function shall be larger than a threshold, p, somewhere in the variable domain $[0; 1]$.
Describe a variable for any value $\Longrightarrow$ maximum in the set of corresponding membership functions shall be larger than p anywhere in the domain.

$$y^+(x) = \frac{1}{1 + e^{-w^+(x-x^*)+\ln\left(\frac{1}{p}-1\right)}}$$

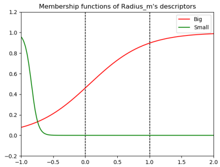
$$y^-(x) = \frac{1}{1 + e^{-w^-(x-x^*)+\ln\left(\frac{1}{p}-1\right)}}$$

- $0 < x^* < 1$ crossing value
- $w^+ > 0$ weight of increasing sigmoïd
- $w^- < 0$ weight of decreasing sigmoïd

Acceptable

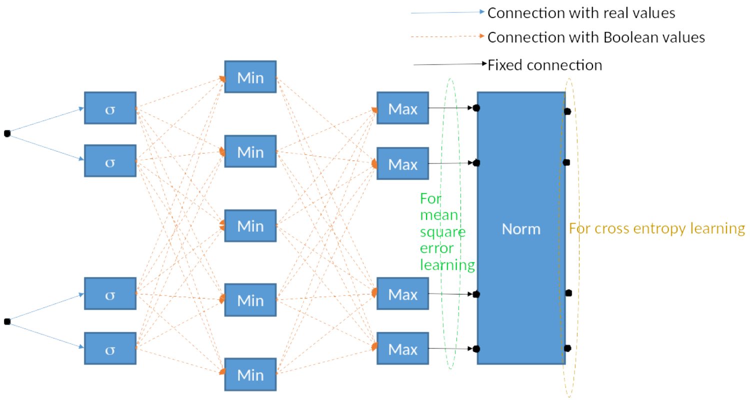


Not acceptable



Continuous and Boolean parameters in Neuro-Fuzzy networks

For identification of probability distributions and classification



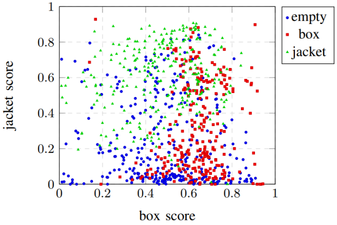
Interleaving a search on Boolean values with back-propagation

Algorithm

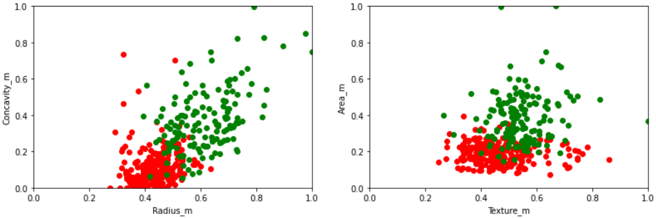
Input: Network - a Neuro-Fuzzy network with mixed parameters
Input: learning_budget - the number of Back-Propagation (BP) steps to be performed globally
Input: bp_steps - number of BP steps to be performed at each numerical weights optimization
Input: Data - the training sample of the data to be modeled
Output: Best_Network_Currently: a Neuro-Fuzzy network with adapted parameters

```
Networks.append(Network); Best_Network_Currently ← Network;
while learning_budget > 0 do
    if Best_Network_Currently.IsClose then
        All_Neighbours ← Open(Best_Network_Currently)\{Networks};
        for Network ∈ All_Neighbours do
            Network.Backpropagate(Data, steps); learning_budget ← learning_budget - steps;
        Best_Neighbours ← Select_On_Loss_Expectancy(All_Neighbours, learning_budget);
        Networks.append(Best_Neighbours); Best_Network_Currently.IsClose ← False;
    Best_Network_Currently.backpropagate(Data, steps);
    learning_budget ← learning_budget - steps;
    Best_Network_Currently ← Select_On_Loss_Expectancy(Networks, learning_budget);
```

Box, jacket or nothing [4]

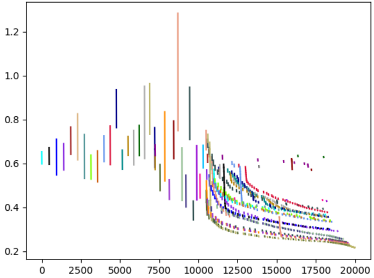


Malignant or benign (https://archive.ics.uci.edu/dataset/17)



Test on classification problems
Some results

Learning loss in function of back-propagation iteration



Typical results:

- x^*: 0.31 for box score, 0.08 for jacket score
- Rules: Box = jacket score small, Jacket = jacket score small and box score small, Empty = jacket score small

References

[1] S. C. Lee and E. T. Lee,
Fuzzy neural networks,
Mathematical Biosciences, 23 (1-2), pp. 151–177 (1975).

[2] J.-S. Jang,
ANFIS: adaptive-network-based fuzzy inference system,
IEEE transactions on systems, man, and cybernetics, 23 (3), pp. 665–685 (1993).

[3] P. V. de Campos Souza,
Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature,
Applied soft computing, 92, p. 106275 (2020).

[4] J.-L. Farges, G. Infantes, C. Lesire and A. Manecy,
Using POMDP with raw observations for detecting and recognizing objects of interest (2018).