

PROJECT ASSIGNMENT PROBLEM SOLVED BY NEURO-FUZZY INFERENCE SYSTEM

Zoran Gacovski
FON University – R. Macedonia



PROJECT ASSIGNMENT PROBLEM SOLVED BY NEURO-FUZZY INFERENCE SYSTEM

1. Introduction
2. Project assignment problem
3. Adaptive neuro-fuzzy inference system (ANFIS)
4. Simulation results
5. Conclusion

1. Introduction

- The job assignment problem (JAP) is non-trivial (NP-hard) problems, and very difficult to find a "reasonable solution in a reasonable time".
- This problem is omnipresent in and highly relevant to many industrial application domains like product manufacturing and workflow organization - scheduling tasks.
- The basic variant of the JAP = assigning jobs to executing staff such that the overall completion time is reduced, considering the abilities of the individual workers.
- It is an essential task to assign the right job to a right person. There are many factors that have impact on the assignment process (conflict criteria appear).
- In many companies – the managers are doing these assignments manually – without taking into consideration the personnel expertise, employees' preferences etc.
- Our work is a try to create an automated tool that will assist during the project assignment process, i.e. intelligent decision-making technique.

- An adaptive neuro-fuzzy inference system (ANFIS) which can learn to make human-like decisions and uses fuzzy membership functions for the soft constraints (input variables).
- For our study – three widely used soft constraints. The goal is to find the most appropriate decisions resulting from fuzzy rules, for matches between projects and employees.
- The use of fuzzy membership functions is convenient, because they express their decisions in terms of linguistic descriptions (low, medium, high) for the input constraints, as well as for the overall match between the projects and the employees.
- Manager decisions are highly correlated and virtually non-deterministic. However, some managers may provide fairly diverse decisions, due to their personal experience, not to mention emotions, subjective factors, mistakes etc..
- One manager can make 20% difference in decision even if the same data would be presented to him at a different time. Such indeterminate subjective component makes the optimization a very sophisticated task, very suitable for artificial intelligence solution.

2. Project assignment problem

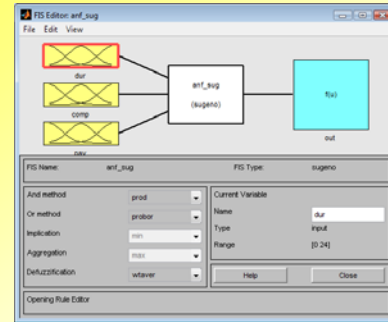
- $J = \{J_1, \dots, J_n\}$ - a set of jobs; $N = \{N_1, \dots, N_m\}$ - set of workers, /each job can be executed by at least one of the workers ($n, m \in \mathbb{N}$).
- The problem: to find an assignment of the jobs, such that the overall time for completing all jobs is minimal.
- For project assignment problem – we consider supervised learning methods. In large corporations – usually all projects and all employees are put in two separate databases.
- First automation of the decision process can be done if every employee is joined with all possible projects – that satisfy his hard constraints (preferences = desires).
- The data can be presented to a manager, who will make decisions (project offers), based on the following three widely used soft constraints (input variables):
 - Duration of the project (in weeks);
 - Type of the project (its complexity);
 - Pay-grade match (the earning that the employee will get from the project).

3. Adaptive Neuro-fuzzy Inference System

- The adaptive neuro-fuzzy inference system uses a given input/output data set, and constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using a back-propagation algorithm.
- Number of input membership functions: three membership functions (low, medium, high) for each of the three input variables model.
- Type of input membership functions: based on the properties of the input variables – we considered triangular membership functions, but trapezoid, Gaussian were tested as well.
- Type of the output membership function: single output, using weighted average defuzzification. All output membership functions had the same type -constant or linear.
- Rule base example:
 1. If (dur is low) and (comp is low) and (pay is low) then (out is mf1) (1)
 4. If (dur is low) and (comp is med) and (pay is low) then (out is mf4) (1)

- The output variable is "employee-project likelihood" and the number of output membership functions ranged from 2 to 81.
- For the testing set – 30 students were questioned to choose one out of 4 offered projects.
- In our case (3 input variables and 3 MFs) – there are $27=3^3$ fuzzy rules; through rule extraction the number of rules was decreased.
- Performance function: some of the widely used performance functions in neural networks are Sum of the squared error and Mean squared error.
- Optimization methods: back-propagation and hybrid (mixed least squared and back-propagation) methods were used as optimization methods.
- Data is partitioned into 3 sets: training, testing and cross-validation: the range of the training data set size was 50-90%. The cross-validation and the testing data sets each took half of the rest of the data (5-25%). Cross-validation is optional.
- Number of training epochs – in most runs it was set up to 50 epochs.

4. Simulation results – Matlab/ Fuzzy toolbox



Simulation results – ANFIS structure

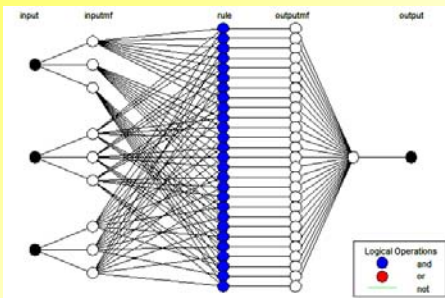


Fig. 1. ANFIS structure with the generated rules.

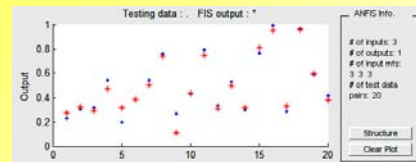


Fig. 2. Testing data vs. actual FIS output.

mf1	mf2	mf3
dur_low= [-9.6 0 9.6]	comp_low= [-4 0 4]	pay_low= [-40 0 40]
dur_med= [2.4 12 21.6]	comp_med= [1 5 9]	pay_med= [10 50 90]
dur_hi= [14.4 24 33.6]	comp_hi= [6 10 14]	pay_hi= [60 100 140]

Table 1 – Membership functions after training

	Before training	After training
Largest difference	mf6= [0.0004 0.0001 0.0056 5e-05]	mf6= [-0.00052 -0.00029 -0.0046 -5.83e-05]
Smallest difference	mf1= [0.017 0.016 0.021 0.0045]	mf1= [0.01678 0.01062 0.0221 0.004419]
Average difference	mf19= [0.00012 .0007 0.001 1e-05]	mf19= [0 0 0 0]

Table 2 – Comparison of output values before and after ANFIS training

5. Conclusions

- The project assignment problem is solved by an adaptive neuro-fuzzy inference system.
- ANFIS (adaptive neuro-fuzzy inference system) = method for fuzzy modeling that learns information about a data set, in order to compute the membership functions.
- 2 methods for generating the rule base – starting from empty rule base (command line version) and starting with initial rule base (GUI version).
- For the training and testing sets – we have used employees' polls about job preference.
- For future research purposes – the number of fuzzy rules will be reduced, but it should held good performance as the original set of rules.
- Finally, we have realized simulation in Matlab, and solved different examples of decision-making for the project assignment problem.