SIGMOID SUPPLEMENTED DECISION STRUCTURES FOR EVIDENCE SENSITIVITY LEARNING

Caelum Kamps, Rahim Jassemi-Zargani
Defense Research & Development Canada (DRDC)
Ottawa, ON, Canada

ABSTRACT

During decision making for classification it is desirable to predict an outcome when not all of the evidence is available. Consider medical diagnosis. If a doctor is trying to determine the cause of a patient's ailment, often they are presented with a subset of potential evidences for or against a particular diagnosis. As the doctor runs more tests and the patients symptoms evolve, the doctor becomes more confident in their evaluation. It is critical that the decision maker be as confident in their decision as possible with as few evidences as are available. The goal of this paper is to improve the ability to predict the final decision given only a subset of the total information. By exploiting interdependencies and probabilistic relationships between the evidences, the confidence of prediction of a decision making tool can be improved through machine learning. Given some complete set of evidences, the Analytical Hierarchy Process (AHP) provides a method of weighting the nodes in a decision structure to synthesize a decision that reflects the opinion of a subject matter expert (SME). By truncating the comparison matrices produced for the AHP, weights can be generated for decision structures that are lacking inputs, known as deficient decision structures. This paper proposes a method of sigmoid node supplementation to the standard decision structure. Using machine learning the parameters of these sigmoid nodes can be optimized so that the output of deficient decision structures can be vastly improved for prediction of the output of the complete decision structure. This method preserves the original weights derived through the AHP and thus the relative importance of evidences is maintained after learning is undergone. An example will illustrate the improved confidence in prediction that can be achieved by adjusting the sensitivity of the supplemented sigmoid nodes.

Keywords: *Machine Learning, Prediction, Uncertainty, Evidence Sensitivity*

1. Introduction

The AHP is useful for making decisions when all of the information is available. Compensating for lacking information and probabilistic relationships within the evidences (cues) is one of its challenges. [1] Given some complete decision structure, meaning all of the cues are available, the AHP provides a method of weight generation for each node. [2] When this decision making structure is deficient, meaning that it is missing some of the cues, then the same comparison matrices generated for the complete decision structure can be truncated to produce weights that maintain the relative importance of the available cues. These weights are consistent in their measurement of relative importance but do not consider the sensitivity of the known cues. Sensitivity appears in the form of relationships between the known and unknown cues as well as the sensitivity of the known cues to the output. For instance, if the value of an unknown cue is highly correlated with a known one, then the known cue is considered more sensitive compared to other known cues with no correlation. To compensate scenarios like the one

described above, this paper proposes a method of sigmoid node supplementation before every input to a weight node. These sigmoid nodes will allow the structure to learn the sensitivity of each input and adjust the parameters accordingly.

2. Objective

The objective of this paper is to propose a method to improve the ability of a decision making tool to predict the optimal decision when only some of the inputs are known. By complimenting the AHP with machine learning and sigmoid node supplementation, relationships and dependencies within the cues can be exploited to greatly improve the accuracy of a deficient decision making structure to predict the output of the complete decision structure.

3. Methodology

The implementation of the method follows four steps:

1 Definition of the information state space: a list of all combinations of possible inputs.
2 Equations of the sigmoid functions and supplementation into the decision structures
3 Mathematical description of the decision structure
4 Sigmoid parameter optimization through machine learning

These steps are described in more detail in the following subsections.

3.1 Information State Space

Given a complete input matrix $x \in \mathbb{R}^{KxM}$, where K is the number of criteria and M is the number of cues in the largest criteria, each of the inputs, $\{x_{ij} | \ i = 1, 2, \dots, K \ , j = 1, 2, \dots, M\}$ are determined to be either independent or dependent**.** [1]

The information state space is the space of all possible availability states of the cues. For N independent cues $x_{ij} \in x$ there are 2^N information states.

3.2 Sigmoid Functions

Each of the weight nodes are initialized to have a sigmoid function applied to their input. The equation of a general sigmoid function is:

$$S(a, b, x\ ) = \frac{1}{1 + e^{-a(x-b)}} \tag{1}$$

Where:

- x is the input to the sigmoid function
- a is the shape parameter
- b is the shift parameter
- S is the sigmoid output

Each of the sigmoid functions is initialized to be a standard sigmoid function (SSF). A SSF has parameter values $a = 4.5 \ and \ b = 0.5$. These parameter values are used for initialization since the corresponding sigmoid function closely mimics the identity function. As a consequence, the traditional decision structure produces very similar output to the SSF supplemented one. This is desirable because it implies that through SSF

supplementation the output of the new decision structure still represents the cue and criteria importance derived through the AHP. The following is an example sigmoid supplemented decision structure.

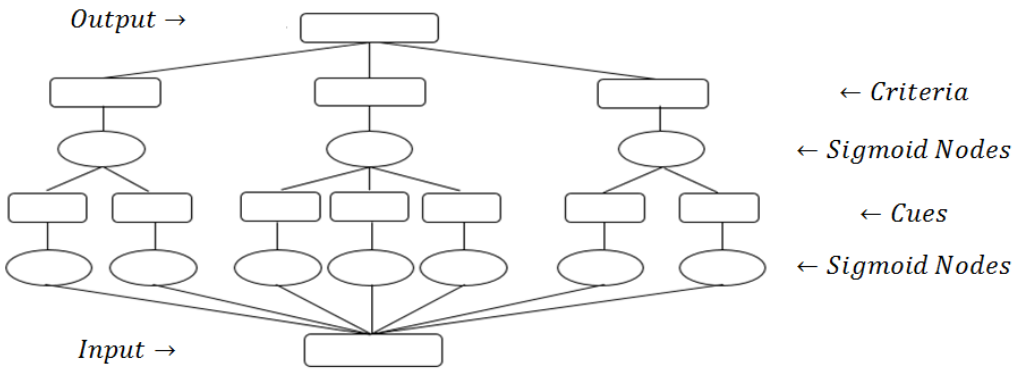

Figure 1: Sigmoid Supplemented Decision Structure

3.3 Mathematical Description of a Decision Structure

Based on the method of output calculation and the variable definitions described by the authors, [1] the output of a sigmoid decision structure can be calculated by the following formula.

$$T_y(\theta_y, \varphi_y, \delta_y, \tau_y, x) = \sum_{i \le K} \left[\theta_{y_i} S\left(\varphi_{y_i}, \delta_{y_i}, \tau_{y_i}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x)\right)\right] \quad (2)$$

Where:

- $y \in [0, 2^N - 1]$ is the information state
- θ_y and ω_y are the weight matrices
- S is a sigmoid function
- φ_y and ρ_y are sigmoid shape matrices
- δ_y and β_y are sigmoid shift matrices
- $\tau_{y_i}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x) = \sum_{j \le M}\left(\omega_{y_{ij}} S\left(\rho_{y_{ij}}, \beta_{y_{ij}}, x_{ij}\right)\right)$ is the input to the sigmoid node for criteria $i \in \{1, \ldots, K\}$

3.4 Learning Process

Learning is performed on the sigmoid shape and shift variables simultaneously. To perform gradient descent, explicit parameter partial derivatives need to be calculated with respect to some cost function. The partial derivative of the sigmoid function with respect to each of its parameters will be useful in the next calculations. These partials are shown here:

$$\frac{\partial S(a,b,c)}{\partial a} = [S(a,b,c)]^2 (c-b) e^{-a(c-b)} \quad (3)$$

$$\frac{\partial S(a,b,c)}{\partial b} = -\frac{\partial S(a,b,c)}{\partial c} = [S(a,b,c)]^2 a e^{-a(c-b)} \quad (4)$$

The partial derivatives with respect to the output T of each of the shape and shift parameters for the criteria layer are:

$$\frac{\partial T_y}{\partial \varphi_{y_i}}(\theta_y, \varphi_y, \delta_y, \tau_y, x) = \theta_{y_i} \frac{\partial S(\varphi_{y_i}, \delta_{y_i}, \tau_{y_i}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x))}{\partial \varphi_{y_i}} \tag{5}$$

$$\frac{\partial T_y}{\partial \delta_{y_i}}(\theta_y, \varphi_y, \delta_y, \tau_y, x) = \theta_{y_i} \frac{\partial S(\varphi_{y_i}, \delta_{y_i}, \tau_{y_i}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x))}{\partial \delta_{y_i}} \tag{6}$$

The partial derivatives of the output of the decision structure with respect to the shape and shift parameters of the cue layer are:

$$\frac{\partial T_y}{\partial \rho_{y_{ij}}}(\theta_y, \varphi_y, \delta_y, \tau_y) = \left(\frac{\partial \tau_{y_i}}{\partial \rho_{y_{ij}}}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x)\right) \frac{\partial T_y}{\partial \tau_{y_i}}(\theta_y, \varphi_y, \delta_y, \tau_y) \tag{7}$$

$$\frac{\partial T_y}{\partial \beta_{y_{ij}}}(\theta_y, \varphi_y, \delta_y, \tau_y) = \left(\frac{\partial \tau_{y_i}}{\partial \beta_{y_{ij}}}(\omega_{y_i}, \rho_{y_i}, \beta_{y_i}, x)\right) \frac{\partial T_y}{\partial \tau_{y_i}}(\theta_y, \varphi_y, \delta_y, \tau_y) \tag{8}$$

To find the partial derivatives with respect to a cost function, apply the chain rule of differentiation, first taking the partial of the cost with respect to T_y and then of T_y with respect to the desired parameter.

The parameters are then updated iteratively with some batch size to shift in the opposite direction of their cost partial derivative multiplied by some positive learning rate. This is represented though a gradient shift of the following form:

$$[\varphi_y, \delta_y, \beta_y, \rho_y] \xleftarrow[t+1 \leftarrow t]{} [\varphi_y, \delta_y, \beta_y, \rho_y] - [\alpha_\varphi, \alpha_\delta, \alpha_\beta, \alpha_\rho] \cdot \nabla C_{\varphi_y, \delta_y, \beta_y, \rho_y} \tag{9}$$

Where:

- C is a cost function dependent on input cases x
- t is the iteration step
- α_A is the learning rate for parameter A

4. Experiment

An experiment in which the values of each of the independent cues are generated by uniform probability distributions on realistic intervals is conducted. The values of the dependent cues are calculated as a product of the independent ones on which they rely. The details of this experiment are described in an earlier paper by the authors. [1] The complete information state decision structure and all of its truncations are initialized to include SSF nodes. A uniform training data set was created using the SSF supplemented complete decision structure as the reference output.

The process of learning the parameters of each of the sigmoid nodes is performed. This is applied to every information state $y \in \{0, 2^N - 2\}$. The following chart depicts the

improvement for each individual information state and is organized by the number of available cues. The red bars represent the untrained SSF supplemented decision structure prediction accuracies for each of the information states. The green bars are the improved accuracies of the trained decision structures on the same set of input data

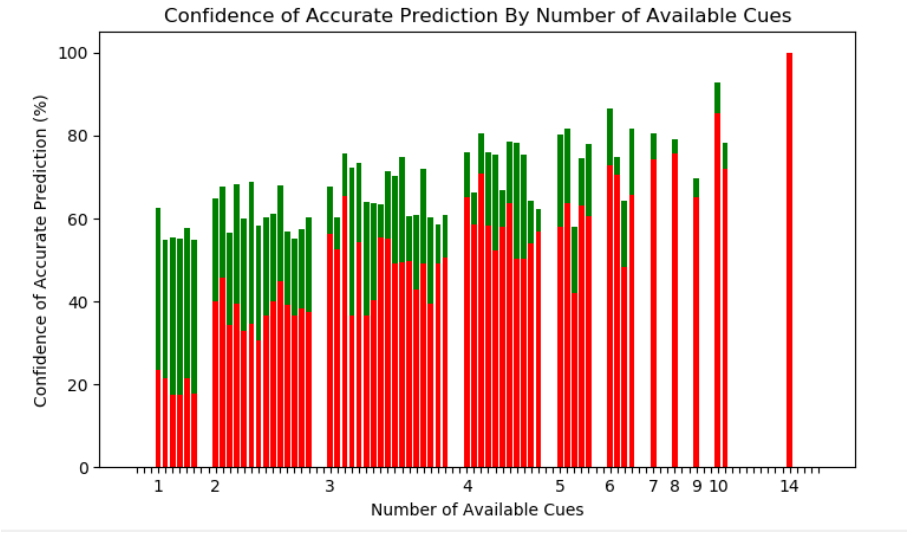

Figure 2: Confidence in Accurate Prediction

In each information state the prediction accuracy is increased and particularly in lower information states large improvement is seen. The improvement seen is a product of learning the probability of outcome based on the available input data. This includes the recognition of any probabilistic relationship between cues or to the output. The following table is a summary of the improvement that is shown in the above bar graph.

Table 1: Accurate Prediction Confidence Improvement:

	Average	**Best**	**Worst**	**STD**
Improvement	17.7 %	42.5 %	0.7 %	9.62 %

The table above shows that every information state can be at least slightly improved and that the information states that are the least confident can be vastly improved. The best case exhibits an improvement of as much as 42.5% confidence of accurate prediction.

5. Limitations

The greatest limitation of this model is the availability of training data. If no training data is available then it is hard to train the model. For this implementation, the training data set was 1500 inputs cases (500 batches of 3 cases). Another limitation is the manual selection of hyper parameter values (learning rates and batch size for gradient descent). To improve the method, it might be beneficial to use a version of gradient descent in which the learning rates and other hyper parameters are automatically updated to improve results.

6. Conclusions

The AHP is a useful tool for synthesizing a decision based on the opinions of a subject matter expert. Supplementation of SSF nodes into the standard decision structure preserves the decisions that would have been made before the supplementation. The goal is to be able to predict the decision of a complete SSF supplemented decision structure with weights generated through the AHP when not all of the information is available. This paper has shown that by applying the AHP to derive the weights and using machine learning to adjust the parameters of the sigmoid nodes in the deficient decision structures, the predictive capabilities can be vastly improved. This method is useful for all states of information availability and shows promise for use in practice.

7. References

[1] Kamps, Caelum, Jassemi-Zargani, Rahim, *Weight Adjustment Using Machine Learning Applied to the Analytical Hierarchy Process*, ISAHP 2018, July 2018.

[2] Saaty, T.L. (1987). *The Analytical Hierarchy Process – What It Is And How It Is Used.* Pittsburgh, PA: Pergamon Journals Ltd.