**Paper**: Constrained Interval Type-2 Fuzzy Classification Systems for XAI

**Published** in 2020 IEEE International Conference on Fuzzy Systems

**Scored** AIA: 10

Source: <https://www.easa.europa.eu/sites/default/files/dfu/EASA-DDLN-Concepts-of-Design-Assurance-for-Neural-Networks-CoDANN.pdfs>

Source: <https://www.easa.europa.eu/sites/default/files/dfu/ddln_easa_codann2_public.pdf>

**The review and scoring of assurance methods:**

Each elemental metric is allocated one point, and each method is either given that point or not (0 or 1), as follows:

* **I. Specificity to AI:** some assurance methods are generically tailored to many systems; others are deployable only to intelligent systems; one point was assigned to methods that focused (i.e. specifc) on the inner workings of AI systems.
* **II. The existence of a formal method:** this metric indicates whether the manuscript under review presented a formal (quantitative and qualitative) description of their method (1 point) or not (0 points).
* **III. Declared successful results:** in experimental work of a method under review, some authors declared success and presented success rates, if that is present, we gave that method a point.
* **IV. Datasets provided:** whether the method has a big dataset associated with it for testing (1) or not (0). This is an important factor for reproducibility and research evaluation purposes.
* **V. AI system size:** methods were applied to a small AI system, other were applied to bigger systems for instance, we gave a point to methods that could be applied to big real-world systems rather than ones with theoretical deployments.
* **VI. Declared success:** whether the authors declared success of their method in reaching an assured AI system (1) or not (0).
* **VII. Mentioned limitations:** whether there are obvious method limitations (0) or not (1).
* **VIII. Generalized to other AI deployments:** some methods are broad and are able to be generalized for multiple AI systems (1), others are “narrow” (0) and more specifc to one application or one system.
* **IX. A real-world application:** if the method presented is applied to a real-world application, it is granted one point.
* **X. Contrasted with other methods:** if the method reviewed is compared, contrasted, or measured against other methods, or if it proves its superiority over other methods, then it is granted a point.

|  |  |
| --- | --- |
| **AIA Scoring Category** | **Why?** |
| **Specificity to AI** |  |
| **The existence of a formal methods** |  |
| **Declared successful results** |  |
| **Datasets provided** |  |
| **AI systems size** |  |
| **Declared success** |  |
| **Mentioned limitation** |  |
| **Generalized to other AI deployments** |  |
| **A real-world application** |  |
| **Contrasted with other methods** |  |

**1/ AI Trustworthiness framework:**

* There are 4 building blocks that structure AI Trustworthiness framework that contribute into the gaining confidence in trustworthiness of DL application
  + AI Trustworthiness analysis = provide guidance to apply how to address keys/pillars in law
  + Learning Assurance = gain confidence at appropriate level that an ML application supports the intended functionality
  + XAI = human centric that deal with capability to explain how AI application comes to its results and output
  + AI safety risk mitigation = supervision of the function of AI application may be necessary.

**1 / How to assure high accuracy performance in neural network? – Learning Process/Learning Assurance**

* Classification? Regression? Optimization Techniques?
* We want fundamentals theoretical guarantees and challenges of the framework for Learning Assurance.

Diagram

Description automatically generated

* Inference = testing phase
* Dataset Splitting? randomly with the ratios 70%–15%–15% yields training, validation, and test datasets Dtrain, Dval, Dtest respectively

**a/ Generalizability** = It is important for NN to perfrm well on unseen data. The ability of a learning algorithm to produce models that perform well on unseen data is referred to as generalizability, or generalization capacity

* We want to secure a good generalizability – an estimation of out-of-sample errors is required. We want to min

Diagram

Description automatically generated

* The difficulty in guaranteeing any bounds on the latter raises from the fact that the true distribution is unknown and can only be estimated.

**a.1/ Bias vs Variance Trade off / Approximation generalization:**

* Bias = The quantity bias(F, n) named bias is the average over all points x ∈ X of the difference between the average model and the target function
* Intuitively, the bias of a learning algorithm can be interpreted as a measure of how well the average model deviates from the true one and thus is a measure of model quality. One wants to make the bias small to have the average model close to the true function f .
* Variance = Intuitively, variance of a learning algorithm can be interpreted as a measure of its fluctuations around the average model and thus reflects how stable it
* **Goal**: We want to obtain a model that can both capture the most important characteristic of the dataset and generalize well to unseen data => Bias and Variance Trade off

Histogram

Description automatically generated

* High bias = poor approximation of the target = large in-sample error
* High variance = poor generalization since small fluctuations in training data might lead to large variations in final model on testing
* Simple models usually have high bias and low variance (sometimes called underfitting), while more complicated ones have lower bias, but higher variance (sometimes called overfitting
* **Both** overfitting and underfitting come with risks: overfitting will lead to models that do not generalize well, while underfitted models will not achieve a satisfactory performance. A tradeoff between these two extremes must be reached, depending on the performance and safety requirements

Chart, line chart

Description automatically generated

- **Method:** Bootstrapping: a class of methods that was introduced by Efron. Bootstrap aggregating (bagging). . The goal is to reduce variance, as in the second example above.

- **Method**: Jackknife. Boosting algorithms work by successively training models, where previously misclassified example are given higher weights for the training of the subsequent models. The goal is to reduce bias and eventually variance.

**a.2/ SLT = statistical learning theory (Generalization):**

- 1. It provides ways to obtain well-generalizing machine learning model

- 2. it provides the means to guarantee bounds on the generalization gap (5.4) (see Figure 5.2). In other words, it makes the statement that predicting unseen data accurately from a finite training sample is pos

* Ensuring theoretical guarantees on the generalization gap (5.4), also called generalization bounds
* **Method**: Probably Approximately Correct (PAC)-learning
* **Method**: Data-independent, algorithm-independent bounds: e derived by Vapnik and Chervonenkis
* **Method**: Data-dependent, algorithm-independent bounds
* **Method:** Data-dependent, algorithm-dependent bounds

**b/ Learning Assurance:**

* Learning Assurance needs to impose strict requirements on the datasets used for development, the development process itself, and verification of the system behavior both during development and operation
* “verification is not simply testing. Testing, in general, cannot show the absence of errors”

Diagram

Description automatically generated

* **Data Management** = It covers the identification of the various datasets used for training and evaluation (typically the training, validation, and test datasets) and the dataset preparation (including collection, labeling and processing)
  + addresses the validation objective of completeness and correctness of the datasets with respect to the product/system requirements and to the ConOps
  + considerations on the quality of the datasets
  + quality of the datasets
  + cover objectives on the independence between datasets an evaluation of the bias and variance inherent to the data.
* **Learning process management** = selection and validation of key elements such as the training algorithm, the activation function, the loss function, the initialization strategy, and the training hyperparameters, which all have the potential to influence the result of the training in terms of performance
* **Model Training** = Once trained, the model performance, bias and variance are evaluated, using the validation dataset
* **Learning Process Verification =** evaluating the trained model performance on the test dataset
  + An evaluation of the bias and variance of the trained model should be performed, as well.
  + The training phase and its verification can be repeated iteratively until the trained model reaches the expected performance.
* Not do on hardware = no model implementation, inference model verification, data verification

**B1/ Dataset management** = these give performance guarantees of a model on unseen data during the operational phase

B1.1 Operational domain identification:

* A crucial step, before considering data quality, is to correctly identify the input space X and its distribution (i.e. the probability space X )
* Failing to do so would prevent establishing any learning guarantees, even though the data is “correct” according to the requirements below
* This is the common issue of “domain bias

Chart

Description automatically generated with medium confidence

**c/ NN Verification Methods:**

Graphical user interface, application, website

Description automatically generated

* Text

  Description automatically generated
* Text

  Description automatically generated
* Text

  Description automatically generated
* Chart

  Description automatically generated with low confidence

**2/ How to explain neural network decision?**

* SHAP?

**3/ How to secure neural network decision?**