2.3.3 Verification of the learned model

**Practical guidance – cross-domain**

Authors: Rob Ashmore (Dstl), and Dr Radu Calinescu and Dr Colin Paterson (Assuring Autonomy International Programme)

The Model Verification stage of the ML lifecycle is concerned with the provision of auditable evidence that a model will continue to satisfy its requirements when exposed to inputs which are not present in the training data.

**Stage Input and Output Artefacts**

The key input artefact to this stage is the trained model produced by the Model Learning stage. The key output artefacts are a verified model, and a verification result that provides sufficient information to allow potential users to determine if the model is suitable for the intended application(s).

**Activities**

1. **Requirement Encoding** - This activity involves transforming requirements into both tests and mathematical properties, where the latter can be verified using formal techniques. Requirements encoding requires a knowledge of the application domain, such that the intent which is implicit in the requirements may be encoded as explicit tests and properties. A knowledge of the technology which underpins the model is also required, such that technology-specific issues (such as overfitting or adversarial vulnerabilities) may be assessed through the creation of appropriate tests and properties.

2. **Test-Based Verification** - This activity involves providing test cases (i.e. specially-formed inputs or sequences of inputs) to the trained model and checking the outputs against predefined expected results. A large part of this activity involves an independent examination of properties considered during the Model Learning stage (guidance on model learning is provided in section 2.3.2), especially the Performant and Robust properties. In addition, this activity also considers test completeness, i.e. whether the set of tests exercised the model and covered its input domain sufficiently. The latter objective is directly related to the Complete property from the Data Management stage (guidance on data management is provided in section 2.3.1).

3. **Formal Verification** - This activity involves the use of mathematical techniques to provide irrefutable evidence that the model satisfies formally-specified properties derived from its requirements. Counterexamples are typically provided for properties that are violated, and can be used to inform further iterations of activities from the Data Management and Model Learning stages.

**Desired Assurance Properties**

In order to be compelling, the verification results (i.e. the evidence) generated by the Model Verification stage should exhibit the following key properties:
1. **Comprehensive** - This property is concerned with the ability of Model Verification to cover:
   a. all the requirements and operating conditions associated with the intended use of the model
   b. all the desired assurance properties from the previous stages of the ML lifecycle (e.g., the completeness of the training data, and the robustness of the model).

2. **Contextually relevant** - This property considers the extent to which test cases and formally verified properties can be mapped to contextually meaningful aspects of the system that will use the model. For example, for a model used in an autonomous car, robustness with respect to image contrast is less meaningful than robustness to variation in weather conditions.

3. **Comprehensible** - This property considers the extent to which verification results can be understood by those using them in activities ranging from data preparation and model development to system development and regulatory approval. A clear link should exist between the aim of the Model Verification and the guarantees it provides. Limitations and assumptions should be clearly identified, and results that show requirement violations should convey sufficient information to allow the underlying cause(s) for the violations to be fixed.

**Methods**

Table 1 provides a summary of the methods that can be applied during each Model Verification activity in order to achieve the desired assurance properties (desiderata). Further details on the methods listed in Table 1, along with the references to the documents cited in the table are available in [1].
Table 1 – Assurance methods for the Model Verification

**Summary of Approach**

1. Take the model to be verified produced by the Model Learning stage (guidance on model learning is provided in section 2.3.2).
2. Apply appropriate methods in order to undertake each model verification activity to demonstrate the machine-learnt model is suitable for its intended use.
   a. Apply appropriate methods for requirement encoding
   b. Apply appropriate methods for verification by adopting either a test-based or formal approach
3. Provide a verification result that provides sufficient information to allow potential users to determine if the model is suitable for its intended application.

**References**


