Performance and Interpretability in Fuzzy Logic Systems – can we have both?

Direnc Pekaslan, Chao Chen, Christian Wagner, and Jonathan M. Garibaldi

University of Nottingham, NG8 1BB, UK
{direnc.pekaslan,chao.chen,christian.wagner,jon.garibaldi}@nottingham.ac.uk

Abstract. Fuzzy Logic Systems can provide a good level of interpretability and may provide a key building block as part of a growing interest in explainable AI. In practice, the level of interpretability of a given fuzzy logic system is dependent on how well its key components, namely, its rule base and its antecedent and consequent fuzzy sets are understood. The latter poses an interesting problem from an optimisation point of view – if we apply optimisation techniques to optimise the parameters of the fuzzy logic system, we may achieve better performance (e.g. prediction), however at the cost of poorer interpretability. In this paper, we build on recent work in non-singleton fuzzification which is designed to model noise and uncertainty 'where it arises', limiting any optimisation impact to the fuzzification stage. We explore the potential of such systems to deliver good performance in varying-noise environments by contrasting one example framework - ADONiS, with ANFIS, a traditional optimisation approach designed to tune all fuzzy sets. Within the context of time series prediction, we contrast the behaviour and performance of both approaches with a view to inform future research aimed at developing fuzzy logic systems designed to deliver both – high performance and high interpretability.

Keywords: Non-Singleton Fuzzy System · Interpretability · ADONiS · ANFIS · Parameter Tuning

1 Introduction

A key aspect of the vision of interpretable artificial intelligence (AI) is to have decision-making models which can be understood by humans. Thus, while an AI may deliver good performance, providing an insight of the decision process is also an important asset for the given model. Even though the interpretability of AI is widely acknowledged to be a critical issue, it still remains as a challenging task [17].

Fuzzy set (FS) theory introduced by Zadeh [34], establishes the basis for Fuzzy Logic Systems (FLSs). Zadeh introduced them to capture aspects of human reasoning and in FLSs are frequently being referred to as 'interpretable'. The main rationale for the latter is that FSs are generally designed in respect to linguistic labels and are interconnected by linguistic rules, which can provide insight into 'why/how results are produced' [28]. This capacity for interpretability is one of the main assets of fuzzy logic and is often one of the key motivations to use FLSs in decision-making [4].

While FLSs are considered to possess mechanisms which can provide a good degree of interpretability, research establishing the latter has been comparatively limited. Only in recent years an increasing number of studies have started to focus on fundamental questions such as what *interpretability* is, in general, and in particular in respect to FLSs? From a complexity point of view, how many rules or how many variables per rule is interpretable? Or from a semantic point of view, to which degree are properties of the partitioning of the variables (e.g. *completeness, distinguishability or complementarity*) key for interpretable FLSs? [1, 12, 15, 19] These studies show that the interpretability of FLSs depends on their various components i.e. the number of rules, the structure of the rule set and the actual interpretability of each rule - which in turn depends on how meaningful the actual FSs are, i.e. how well they reflect the model which the interpreting stakeholder has in mind when considering the given linguistic label [12, 13, 28].

Traditionally, AI models use statistical optimisation techniques to tune parameters based on a data-driven approach. While these optimisation procedures provide performance benefits, they commonly do not consider whether the resulting model is interpretable or not. This poses an interesting question for the optimisation or tuning of FLS: can we use statistical optimisation to tune FLS parameters without negatively affecting the given FLSs interpretability? I.e., can we have both: interpretability and good performance?

There are several established approaches to tune FLSs using statistical optimisation. Here, ANFIS (adaptive-network-based fuzzy inference system), introduced by Jang [14], and later extended in [6] for interval type-2 fuzzy logic system has been one of the most popular. ANFIS uses statistical optimisation to update FLS parameters based on a given training dataset with the objective to deliver good performance, i.e. minimum error. However, during the optimisation, ANFIS does not consider aspects of interpretability [27], for example potentially changing antecedent and consequent sets drastically in ways which do not align with stakeholders' expectations.

This paper explore whether and how we can design FLSs which can preserve their interpretability while also providing the required degrees of freedom for statistical tuning to deliver good performance.

To achieve the latter, we focus on Non singleton FLSs (NSFLSs) [5, 22], which are designed to model disturbance affecting a system through its inputs within the (self-contained) fuzzification stage. Recently, NSFLS approaches have received increasing attention [10, 11, 21, 24–26, 29, 31, 32], with a particular focus on the development of FLSs which 'model uncertainty where it arises', i.e. FLSs which model input uncertainty directly and only within the input fuzzification stage. The latter provides an elegant modelling approach which avoids changing otherwise unrelated parameters (e.g. antecedent or consequent FSs) in respect to disturbance affecting a systems' inputs. Most recently, the ADONiS framework [23] was proposed, where input noise is estimated and the fuzzification stage is adapted at run-time, delivering good performance in the face of varying noise conditions. As noted, ADONiS limits tuning to the fuzzification stage, leaving rules (which can be generated based on experts insights or in a data-driven way) 'untouched', thus providing a fundamental requirement for good interpretability.

In this paper, we compare and contrast the effects of employing both the ANFIS optimisation and the ADONiS adaptation frameworks in response to varying noise levels in a time-series prediction context. We do not aim to explore which approach delivers the best time series prediction (for that, many other machine learning methods are available), but rather, how the resulting FLSs compare after tuning, when both approaches deliver good or at least reasonable results. Specifically, we focus on the degree to which the key parameters – antecedents and consequents are preserved (we maintain an identical rule set to enable systematic comparison), and thus to which degree the original interpretability of a FLS can be preserved post-tuning using such approaches.

The structure of this paper is as follows. Section II gives a brief overview of singleton, non-singleton FSs, as well as the ADONiS and ANFIS models. Section III introduces methodology including details of the rule generation, training and testing. Section IV provides detailed steps of the conducted experiments and a discussion of the findings. In Section V, the conclusions of experiments with possible future work directions are given.

2 Background

2.1 Singleton Non-Singleton Type-1 Fuzzy Sets

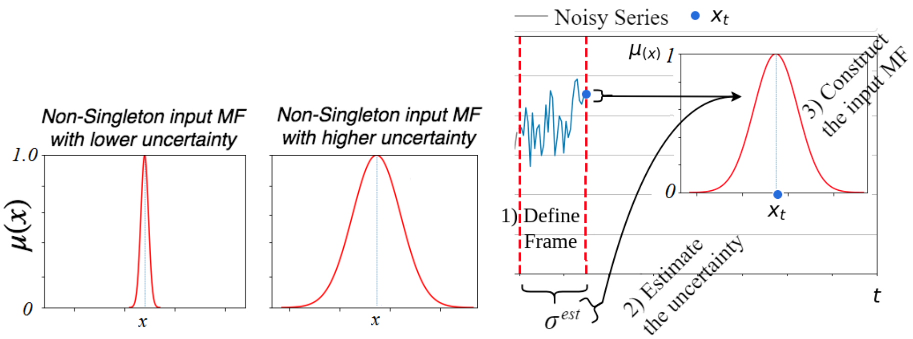
In the fuzzification step of fuzzy models, a given crisp input is characterised as membership function membership function (MF). Generally in singleton fuzzification, the given input x is represented by singleton MF.

When input data contain noise, it may not be appropriate to represent them as singleton MFs, as there is a possibility of the actual value being distorted by this noise. In this case, the given input x is mapped to non-singleton MFs with a support where membership degree achieves maximum value at x. Two samples of non-singleton MFs -under relatively low and high noise- can be seen in Fig. 1a.

Conceptually, the given input is assumed to be likely to be correct, but because of existing uncertainty, neighbouring values also have potential to be correct. As we go away from the input value, the possibility of being correct decreases. As shown in Fig. 1a the width of the non-singleton input is associated with the uncertainty levels of the given input.

2.2 ADONiS

The recently proposed ADONiS [23] framework provides two major advantages over non-singleton counterpart models: (i) in the fuzzification step, it captures input uncertainty through an online learning method–which utilises a sequence


Non-Singleton input MF with lower uncertainty
Non-Singleton input MF with higher uncertainty
1.0
μ(x)
0
x
x
Noisy Series
● x_t
μ(x) 1
0
x_t
1) Define Frame
2) Estimate the uncertainty
3) Construct the input MF
σ^{est}
t

(a) Non-Singleton Gaussian FSs.

(b) Illustration of constructing input FSs based noise estimation in a sliding window.

Fig. 1. Different Non-Singleton FSs and ADONiS Framework structure

of observations to continuously update the input Fuzzy Sets (ii) in the inference engine step, it handles the captured uncertainty through the $sub-$NS [24] method to produce more reasonable firing strengths.

Therefore, the ADONiS framework enables us to model noise and uncertainty 'where it arises' and also to limit any optimisation impact to the fuzzification and inference steps. In doing so, ADONiS limits tuning to the fuzzification stage and remain rules (which can be generated based on experts insights or in a data-driven way) 'untouched', thus providing a fundamental requirement for good interpretability. –if rules and sets were understood well initially.

The general framework structure of the ADONiS framework can be summarised in the following four steps:

1. Defining a frame size to collect a sequence of observations. For example, when using sensors, such as in a robotics context, the size of the frame may be selected in respect to the sampling rate of the sensors or based on a fixed time frame.
2. In the defined frame, the uncertainty estimation of the collected observation is implemented. Different uncertainty estimation techniques can be implemented in the defined frame.
3. Non-singleton FS is formed by utilising the estimated uncertainty around the collected input. For example, in this paper, Bell shaped FSs are used and the detected uncertainty is utilised to define the width of these FSs.
4. In the inference engine step of NSFLSs, interaction between the input and antecedent FSs results in the rule firing strengths which in turn determines the degree of truth of the consequents of individual rules. In this step, in this paper, the $sub-$NS technique [24] is utilised to determine the interaction and thus firing strength between input and antecedent FSs.

The overall illustration of ADONiS can be seen in Fig. 1b and for details, please refer the [23, 24].

2.3 ANFIS

Neuro-fuzzy models are designed to combine the concept of artificial neural networks with fuzzy inference systems. As one common model, ANFIS is widely used in many applications to improve the performance of fuzzy inference systems [2, 3, 9, 16]. With ANFIS, model parameters are 'fine-tuned' during optimisation procedures to obtain more accurate approximation than a predefined fuzzy system. An ANFIS illustration with seven antecedents can be seen in Fig. 2 [6].

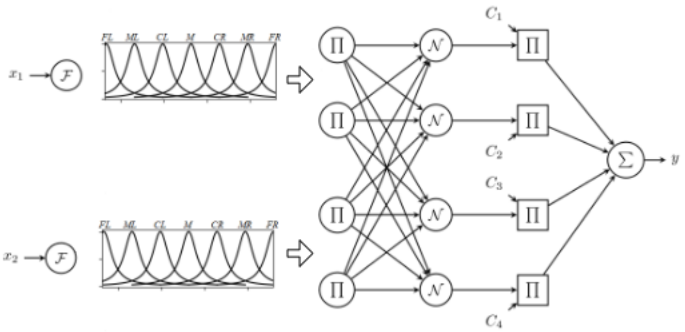


Fig. 2. ANFIS structure

3 Methodology

A Mackey-Glass (MG) time series is generated and 1009 noise-free values are obtained for t from 100 to 1108. One of the common models for noise is additive white Gaussian noise [20] . Three different signal-to-noise ratios (20 dB, 5 dB and 0 dB) are used to generate noisy time series with additive Gaussian white noise. These four (noise-free and noisy) datasets are split into 70% (training) and 30% (testing) samples to be used in different variants of the experiments. In the MG generation, τ value is set to be 17 to exhibit chaotic behaviour.

3.1 Rule Generation

In the literature there are many different rule generation techniques, either expert-driven or data-driven [8, 18]. In this paper, one of the most commonly used techniques for FLS rule generation – the *one-pass* Wang-Mendel method is utilised. Even though in the case of interpretability assessment, Wang-Mendel may not be the best approach to generate rules, in order to make a base rule set for both ADONiS and ANFIS and make a fair comparison, we choose to use *one-pass* Wang-Mendel method. In the future, different rule reduction algorithms or other rule generation techniques can be investigated. By following similar FLS architecture in [33], the rule generation is implemented as follows:

First, the domain of the training set $[x_{min}, x_{max}]$ is defined. In order to capture all inputs (including the ones which are outside of the input domain), the defined domain is expanded by 10% and the *cut-off* procedure is implemented for the inputs which are outside of this domain.

Then the input domain is evenly split into seven regions, and bell-shaped antecedents are generated. As shown in Fig. 3, these are named as *Further Left (FL), Medium Left (ML), Close Left (CL), Medium (M), Close Right (CR), Medium Right (MR), Further Right (FR).*

As in [33], nine past values are used as inputs and the following (10^{th}) value is predicted, i.e. the output.

After forming the input-output pairs as $((\mathbf{x}^1 : y^1), (\mathbf{x}^2 : y^2), ..., (\mathbf{x}^N : y^N),)$ each input value within the pair is assigned to the corresponding antecedent FS $(FL, .., M, .., FR)$. As practised in the Wang-Mendel *one-pass* method, the same seven FSs are used for the consequent FSs, and the outputs $(\mathbf{y}^i)$ are assigned to the corresponding FSs $(FL, .., M, .., FR)$as well. A sample of the generated rules can be seen in (1). For details, please refer [33].

$$R^1 = \textbf{IF}\, x_1\, is\, MR\, \textbf{AND}...\ x_9\, is\, M\ \textbf{THEN}\, y_1\, is\, CR \tag{1}$$

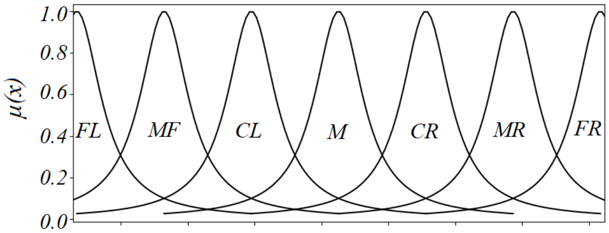


Fig. 3. An illustration of the seven antecedent MFs used.

3.2 Training

ADONiS When implementing ADONiS, no formal optimisation procedure is used. Therefore, previously established antecedents $(FL, .., M, ..FR)$ and model rules remain untouched.

ANFIS Optimisation In ANFIS implementation, each of the seven antecedent MFs are assigned an input neuron (See Fig. 2) [6]. Then, the gradient descent optimisation technique is implemented to update the antecedent MF parameters and the consequent linear functions. In the meantime, the least-squares estimation method [30] is used to update the parameters of consequent linear functions in each training epoch. During each epoch, the antecedent FS parameters are updated for each input. Therefore, while beginning with only seven antecedents, after optimisation, many different antecedent FSs may be generated–with associated increase in model complexity.

3.3 Performance Evaluation

In order to assess the noise handling capability of each model, we calculate the difference between model predictions and noise-free data values at each time-point. Both ADONiS and ANFIS performances are measured by using the common root-mean-squared-error (RMSE) and in addition, the recently proposed Unscaled Mean Bounded Relative Absolute Error (UMBRAE) [7]. UMBRAE combines the best features of various alternative measures without suffering their common issues.

To use UMBRAE, a benchmark method needs to be selected. In this paper, the benchmark method simply uses the average of input values as predictions. With UMBRAE, the performance of a proposed method can be easily interpreted: when $UMBRAE$ is equal to 1, the proposed method performs approximately the same as the benchmark method; when $UMBRAE < 1$, the proposed method performs better than the benchmark method; when $UMBRAE > 1$, the proposed method performs worse than the benchmark method.

4 Experiment and Results

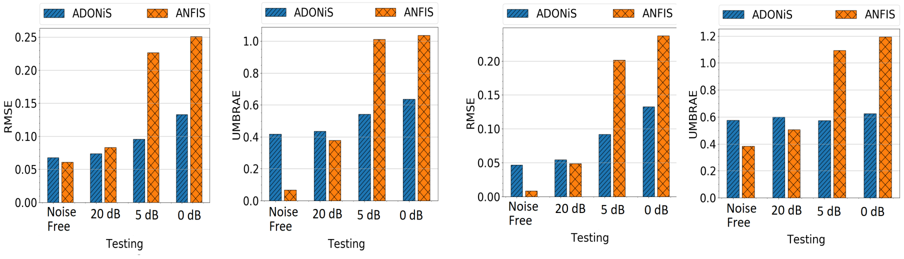
In total, 4x4=16 different experimental scenarios are implemented, using different noise levels in both rule generation/optimisation and testing phases. Specifically, four different training sets (noise-free, 20, 5 and 0 dB) and four different testing sets (noise-free, 20, 5 and 0 dB) are used–to represent a variety of potential real-world noise levels. In each experiment of ADONiS, the first 700 values are used to generate rules and the remaining 300 values are used for testing. Note that as ADONiS uses 9 inputs to construct input FSs, the first 9 values of the testing set are omitted, leaving only the final 291. In ANFIS, while using the exact same rules as ADONiS, the first 400 data pairs are used as the training set; the following 300 data pairs are used as a validation set; and the final 291 of the remaining 300 data pairs are used as testing set.

4.1 Experiment 1: Noise-free Rule Generation

In the first experiment, the rule set is generated using the noise-free time series dataset. Four different testing datasets (noise-free, 20, 5 and 0 dB) are used.

Results of the ADONiS prediction experiment, with noise free testing, can be seen on the left hand side of Fig. 4a. Note that since there is no noise in the testing dataset, the generated input FSs tend to be a singleton FS. Thus, the traditional singleton prediction is implemented in this particular experiment.

After completing noise-free testing, and using the same rule set (from the noise-free training dataset), the 20 dB testing dataset is used in the prediction experiment of ADONiS. The RMSE result of this experiment is shown in Fig. 4a. Thereafter, the remaining 5 dB and 0 dB testing datasets are used with the same rule set–RMSE results are shown in Fig. 4a.



(a) Experiment 1: Noise-Free rule generation

(b) Experiment 2: 20dB Noisy rule generation

Fig. 4. RMSE and UMBRAE results for both Experiment 1 and Experiment 2.

Following the ADONiS prediction (with noise-free rule set and four different testing datasets), ANFIS optimisation is carried out on the previously generated rule parameters and the antecedent parameters are updated in the *'black-box'* manner. Then, these updated antecedents are used in the prediction of noise-free testing dataset. The results of this experiment are shown in Fig. 4a as orange bars. Overall, as can be seen, ANFIS outperform ADONiS significantly in this particular experiment.

Thereafter, the same updated rules from the noise-free training dataset, are used with the 20 dB testing dataset. The performance of ANFIS is reported in Fig. 4a. As can be seen, ADONiS and ANFIS have similar performances under the noise-free and 20 dB noisy testing variant.

Following this, 5 dB and 0 dB noisy datasets were used in testing–RMSE results are illustrated in Fig. 4a. As shown, in both of these noisy conditions, ADONiS outperform ANFIS substantially.

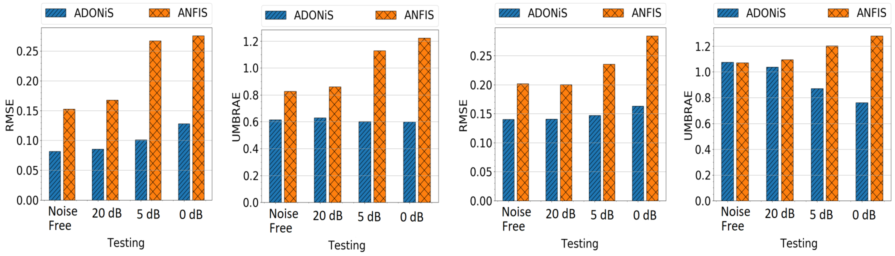
As the second error measure, UMBRAE is calculated between the prediction and noise-free input datasets. These sets of experiment results can be seen right hand side of Fig. 4a.

4.2 Experiment 2: 20 dB Noisy Rule Generation

In the four sets used in this experiment, rule generation is completed by using the 20 dB noisy time series dataset. The resulting rules are then used in ADONiS predictions on the noise-free, 20 dB, 5 dB and 0 dB noisy datasets. The RMSE experiment results are shown in Fig. 4b.

After ADONiS implementation, ANFIS optimisation is implemented on the antecedents' parameters, according to the 20 dB noisy training dataset. Then the ANFIS predictions are performed on the same four (noise-free, 20 dB, 5 dB and 0 dB) different datasets. These prediction results are illustrated in Fig. 4b. These findings show a clear trend that under noise-free or low-noise conditions, ADONiS and ANFIS provide similar performances. Under higher noise levels (5 and 0 dB), ADONiS has a clear performance advantage.

Equivalent results, as evaluated using the UMBRAE error measure, are illustrated on the right hand side of Fig. 4b.



(a) Experiment 3: 5 dB Noisy rule generation

(b) Experiment 4: 0 dB Noisy rule generation

Fig. 5. RMSE and UMBRAE results for both Experiment 3 and Experiment 4.

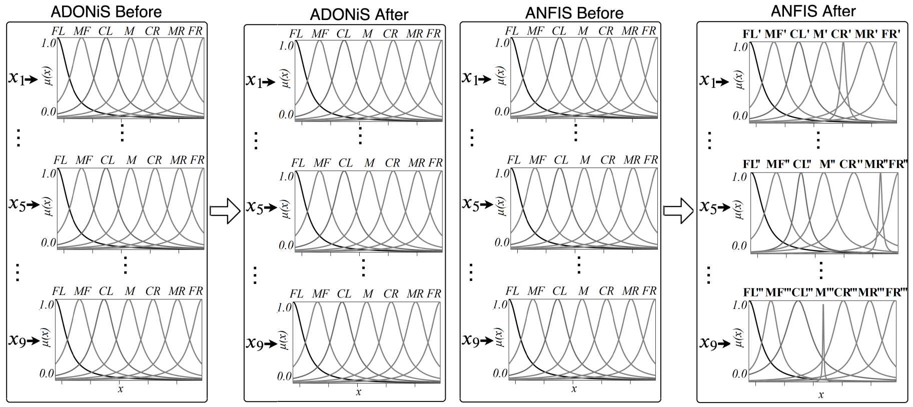
4.3 Experiment 3 and 4: 5 and 0 dB Noisy Rule Generation

The same procedures from the previous experiments are followed. First, rules are generated, based upon the 5dB noisy time series datasets. Next, ADONiS performance is tested with the four (noise-free, 20 dB, 5 dB and 0 dB) datasets. Afterwards, ANFIS optimisation is used to update the antecedent parameters and ANFIS predictions are completed on the same four (noise-free, 20 dB, 5 dB and 0 dB) testing datasets. 5dB rule generation results are shown in Fig. 5a for RMSE and UMBRAE.

Thereafter, 0 dB rule generation is completed and the four different testing results are illustrated in Figs. 5b.

5 Discussion

Overall, the interpretability of a fuzzy model builds upon several components i.e. rules, antecedents and/or consequent numbers, and the semantics at the fuzzy partitioning level. Traditionally, while optimisation techniques may provide a better performance, it leads to changing the parameters (i.e. antecedents FSs) based on a training dataset which results in a less interpretable model. However, since FLSs have mechanisms to provide interpretability, the changing of these parameters in a data-driven way can deteriorate the interpretability of models by causing for example a loss of complementarity, coverage or distinguishability of FSs across a universe of discourse and thus the meaningfulness of the used FSs. Conversely, tuning parameters in the fuzzification step can maintain the interpretability as well as provide a performance benefits.



(a) A comparison of the used antecedent in ADONiS.

(b) A comparison of used antecedents in ANFIS.

Fig. 6. The used antecedent FSs in both model ADONiS and ANFIS.

Regarding the rule generation in the experiments, while different approaches have been introduced [18], in this paper we follow the well established Wang-Mendel [33] rule generation technique. We acknowledge that other approaches may be equally or more viable for example in the given domain of time series prediction, nevertheless, for this paper, our key objective was to generate one basic rulebase which is maintained identical across all FLSs, thus providing a basis for systematic comparison. Further, we note that the specific antecedent and consequent FSs used here are selected arbitrarily (to evenly partition the domain of the variables), and thus are not meaningful in a traditional linguistic sense. However, in this paper, we consider the preservation of the original shape of the FSs (post-tuning) as important (as it is that shape which will be meaningful in applications of FLSs such as in medical decision making, common control applications, etc.).

In the experiment, we first explore the ADONiS model which targets the fuzzification step by limiting the optimisation effect but handling noise 'where it arises'. Second, traditional ANFIS optimisation is used. In this section, after a brief performance comparison, the interpretability is discussed for both models.

Overall, when all the results are scrutinised all together (Figs. 4a,4b,5a,5b), it can be seen that ADONiS and ANFIS provide comparable performance. While ANFIS shows better performance in the noise-free training and noise-free cases, especially under high levels of noise, ADONiS' performance is better than that of the ANFIS-tuned FLS.

In the experiments, by following the structure in [33], the input domain is divided into 7 antecedents, from *Further Left* to *Further Right* ($FL, .., M, .., FR$) (See Fig.3) and each input is assigned with these antecedents as shown on the left hand side of Fig. 6a. The same rule set is generated once. In the ADONiS approach, no optimisation procedure is performed offline (all tuning is done online through adaptation) and all the rules, antecedents, consequents remain intact. As can be seen on the right hand side of Fig. 6a, the same antecedents and consequents are used in the testing stage for ADONiS. Here, the input uncertainty is captured and handled throughout the fuzzification and inference engine process rather than optimising antecedent or consequent parameters. We note that this is intuitive as changes affecting the inputs should not affect the linguistic models of antecedents and consequents - preserving interpretability. For example, when a rule is examined in (2), all the *Medium* (M) MFs are the same as in Fig. 3 and it can be observed that the given sample inputs x_1 and x_9 are processed using the same MFs.

$$IF\, x_1\, is\, M ...\, x_5\, is\, MR ...\, x_9\, is\, M\ THEN\, y_1\, is\, CR \tag{2}$$

On the other hand, in the ANFIS implementation, although the same rules are used (see the left hand side of Fig. 6b) the optimisation procedure focuses on the antecedent parameters. Thus, the parameters are changed in respect to the training data, changing the antecedent and thus necessarily making it different to the original (considered interpretable) model (see the right hand side of Fig. 6b). This overall can affect both the semantics and the complexity at the fuzzy partitioning level. For example, the *Medium* MF is changed through the optimisation procedure. As can be seen in Fig. 6b and rule (3), the *Medium'* (M') and *Medium"'* (M''') are not the same for inputs x_1 and x_9 which inhibits the interpretability of the model.

$$IF\, x_1\, is\, \boldsymbol{M'} ...\, x_5\, is\, \boldsymbol{MR''} ...\, x_9\, is\, \boldsymbol{M'''}\, THEN\, y_1 \tag{3}$$

Therefore, overall, these initial results show that while both models can provide comparable prediction results under different levels of noise, tuning parameters in the fuzzification stage only can help to maintain the semantic meaningfulness (completeness, distinguishability and complementarity) of the used antecedent FSs which can overall provide a more 'interpretable' FLS model in contrast to a 'brute force' optimisation approach such as offered by traditional optimisation approaches for FLSs such as ANFIS.

6 Conclusions

One of the main motivations to use FLSs is their capacity for interpretability ability which is highly related to both complexity (number and structure of rules, variables) and semantic (completeness, distinguishability or complementarity at the level of the fuzzy set partitions) aspects. In regards to the performance of FLSs, while optimisation techniques can be applied to deliver improved performance, such optimisation has traditionally lead to changes of the same key parameters which are vital for interpretability, thus delivering improved performance at the cost of poorer interpretability. In this paper, we explore the possibility of automatically tuning an FLS to deliver good performance, while *also* preserving its valuable interpretable structure, namely the rules (kept constant), antecedents and consequents. Through a detailed set of time series prediction experiments, the potential of the ADONiS framework, which handles input noise where it arises, is explored in comparison to a traditional ANFIS optimisation approach. The behaviour and performance of both approaches is analysed with a view to inform future research aimed at developing FLSs with both high performance and high interpretability.

We believe that these initial results highlight a very interesting research direction for FLSs which can maintain interpretability by modelling complexity only in specific parts of their structure. Future work will concentrate on expanding the experimental evaluation with different rule generation techniques and datasets while broadening the capacity for optimisation beyond the specific design of ADONiS. Also, the use of interpretability indices will be explored to compare/contrast different model efficiently in regards to performance and interpretability ability.

References

1. Alonso, J.M., Magdalena, L., González-Rodríguez, G.: Looking for a good fuzzy system interpretability index: An experimental approach. International Journal of Approximate Reasoning **51**(1), 115 – 134 (2009). https://doi.org/https://doi.org/10.1016/j.ijar.2009.09.004, http://www.sciencedirect.com/science/article/pii/S0888613X09001418
2. Atsalakis, G.S., Valavanis, K.P.: Forecasting stock market short-term trends using a neuro-fuzzy based methodology. Expert Systems with Applications **36**(7), 10696–10707 (2009). https://doi.org/https://doi.org/10.1016/j.eswa.2009.02.043, http://www.sciencedirect.com/science/article/pii/S0957417409001948
3. Azadeh, A., Saberi, M., Gitiforouz, A., Saberi, Z.: A hybrid simulation-adaptive network based fuzzy inference system for improvement of electricity consumption estimation. Expert Systems with Applications **36**(8), 11108–11117 (2009). https://doi.org/https://doi.org/10.1016/j.eswa.2009.02.081, http://www.sciencedirect.com/science/article/pii/S0957417409002413
4. Bodenhofer, U., Bauer, P.: A formal model of interpretability of linguistic variables. In: Interpretability issues in fuzzy modeling, pp. 524–545. Springer (2003)

5. Cara, A.B., Wagner, C., Hagras, H., Pomares, H., Rojas, I.: Multiobjective optimization and comparison of nonsingleton type-1 and singleton interval type-2 fuzzy logic systems. IEEE Trans. Fuzzy Syst. **21**(3), 459–476 (June 2013). https://doi.org/10.1109/TFUZZ.2012.2236096
6. Chen, C., John, R., Twycross, J., Garibaldi, J.M.: An extended ANFIS architecture and its learning properties for type-1 and interval type-2 models. In: Proceedings IEEE International Conference on Fuzzy Systems. pp. 602–609 (2016)
7. Chen, C., Twycross, J., Garibaldi, J.M.: A new accuracy measure based on bounded relative error for time series forecasting. PLOS ONE **12**(3), 1–23 (2017)
8. Chi, Z., Wu, J., Yan, H.: Handwritten numeral recognition using self-organizing maps and fuzzy rules. Pattern Recognition **28**(1), 59 – 66 (1995). https://doi.org/https://doi.org/10.1016/0031-3203(94)00085-Z, http://www.sciencedirect.com/science/article/pii/003132039400085Z
9. Esfahanipour, A., Aghamiri, W.: Adapted Neuro-Fuzzy Inference System on indirect approach TSK fuzzy rule base for stock market analysis. Expert Systems with Applications **37**(7), 4742–4748 (2010). https://doi.org/https://doi.org/10.1016/j.eswa.2009.11.020, http://www.sciencedirect.com/science/article/pii/S0957417409009622
10. Fu, C., Sarabakha, A., Kayacan, E., Wagner, C., John, R., Garibaldi, J.M.: A comparative study on the control of quadcopter uavs by using singleton and non-singleton fuzzy logic controllers. In: 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). pp. 1023–1030 (July 2016). https://doi.org/10.1109/FUZZ-IEEE.2016.7737800
11. Fu, C., Sarabakha, A., Kayacan, E., Wagner, C., John, R., Garibaldi, J.M.: Input uncertainty sensitivity enhanced nonsingleton fuzzy logic controllers for long-term navigation of quadrotor uavs. IEEE/ASME Transactions on Mechatronics **23**(2), 725–734 (April 2018). https://doi.org/10.1109/TMECH.2018.2810947
12. Gacto, M.J., Alcalá, R., Herrera, F.: Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. Information Sciences **181**(20), 4340–4360 (2011). https://doi.org/https://doi.org/10.1016/j.ins.2011.02.021, http://www.sciencedirect.com/science/article/pii/S0020025511001034
13. Garibaldi, J.M., Ozen, T.: Uncertain Fuzzy Reasoning: A Case Study in Modelling Expert Decision Making. IEEE Transactions on Fuzzy Systems **15**(1), 16–30 (feb 2007)
14. Jang, J.S.: ANFIS: adaptive-network-based fuzzy inference system. IEEE Transactions on Systems, Man, and Cybernetics **23**(3), 665–685 (1993)
15. Jin, Y.: Fuzzy modeling of high-dimensional systems: Complexity reduction and interpretability improvement. IEEE Transactions on Fuzzy Systems **8**(2), 212 – 221 (2000). https://doi.org/10.1109/91.842154, http://epubs.surrey.ac.uk/7644/
16. Khanesar, M.A., Kayacan, E., Teshnehlab, M., Kaynak, O.: Extended Kalman Filter Based Learning Algorithm for Type-2 Fuzzy Logic Systems and Its Experimental Evaluation. IEEE Transactions on Industrial Electronics **59**(11), 4443–4455 (2012)
17. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., Sayres, R.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). arXiv preprint arXiv:1711.11279 (2017)
18. Li-Xin Wang: The wm method completed: a flexible fuzzy system approach to data mining. IEEE Transactions on Fuzzy Systems **11**(6), 768–782 (Dec 2003). https://doi.org/10.1109/TFUZZ.2003.819839

19. Liu, H., Gegov, A., Cocea, M.: Complexity Control in Rule Based Models for Classification in Machine Learning Context. In: Angelov, P., Gegov, A., Jayne, C., Shen, Q. (eds.) Advances in Computational Intelligence Systems. pp. 125–143. Springer International Publishing, Cham (2017)
20. Liu, X., Tanaka, M., Okutomi, M.: Single-image noise level estimation for blind denoising. IEEE Transactions on Image Processing **22**(12), 5226–5237 (2013)
21. Mohammadzadeh, A., Kayacan, E.: A non-singleton type-2 fuzzy neural network with adaptive secondary membership for high dimensional applications. Neurocomputing **338**, 63–71 (2019). https://doi.org/https://doi.org/10.1016/j.neucom.2019.01.095, http://www.sciencedirect.com/science/article/pii/S0925231219301882
22. Mouzouris, G.C., Mendel, J.M.: Nonlinear time-series analysis with non-singleton fuzzy logic systems. In: Computational Intelligence for Financial Engineering, 1995., Proceedings of the IEEE/IAFE 1995. pp. 47–56. IEEE (1995)
23. Pekaslan, D., Wagner, C., Garibaldi, J.M.: Adonis - adaptive online non-singleton fuzzy logic systems. IEEE Transactions on Fuzzy Systems pp. 1–1 (2019). https://doi.org/10.1109/TFUZZ.2019.2933787
24. Pekaslan, D., Garibaldi, J.M., Wagner, C.: Exploring Subsethood to Determine Firing Strength in Non-Singleton Fuzzy Logic Systems. IEEE World Congress On Computational Intelligence (2018)
25. Pourabdollah, A., Wagner, C., Aladi, J.H., Garibaldi, J.M.: Improved uncertainty capture for nonsingleton fuzzy systems. IEEE Trans. Fuzzy Syst. **24**(6), 1513–1524 (Dec 2016). https://doi.org/10.1109/TFUZZ.2016.2540065
26. Pourabdollah, A., John, R., Garibaldi, J.M.: A new dynamic approach for non-singleton fuzzification in noisy time-series prediction. In: Fuzzy Systems (FUZZ-IEEE), 2017 IEEE International Conference on. pp. 1–6. IEEE (2017)
27. Rajab, S.: Handling interpretability issues in anfis using rule base simplification and constrained learning. Fuzzy Sets and Systems **368**, 36 – 58 (2019). https://doi.org/https://doi.org/10.1016/j.fss.2018.11.010, theme: Fuzzy Systems and Learning
28. Razak, T.R., Garibaldi, J.M., Wagner, C., Pourabdollah, A., Soria, D.: Interpretability and complexity of design in the creation of fuzzy logic systems — a user study. In: 2018 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 420–426 (Nov 2018). https://doi.org/10.1109/SSCI.2018.8628924
29. Ruiz-García, G., Hagras, H., Pomares, H., Ruiz, I.R.: Toward a fuzzy logic system based on general forms of interval type-2 fuzzy sets. IEEE Transactions on Fuzzy Systems **27**(12), 2381–2395 (2019)
30. Sorenson, H.W.: Least-squares estimation: from Gauss to Kalman. IEEE Spectrum **7**(7), 63–68 (jul 1970)
31. Tellez-Velazquez, A., Cruz-Barbosa, R.: A cuda-streams inference machine for non-singleton fuzzy systems. Concurrency and Computation: Practice and Experience **30** (10 2017). https://doi.org/10.1002/cpe.4382
32. Wagner, C., Pourabdollah, A., McCulloch, J., John, R., Garibaldi, J.M.: A similarity-based inference engine for non-singleton fuzzy logic systems. In: Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on. pp. 316–323. IEEE (2016)
33. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. IEEE Trans. Syst., Man, Cybern. **22**(6), 1414–1427 (1992)
34. Zadeh, L.: Fuzzy sets. Information and Control **8**(3), 338 – 353 (1965). https://doi.org/https://doi.org/10.1016/S0019-9958(65)90241-X