Selecting decision trees for power system security assessment

Bugaje, Al-Amin B. ; Cremer, Jochen L.; Sun, Mingyang; Strbac, Goran

DOI
10.1016/j.egyai.2021.100110

Publication date
2021

Document Version
Final published version

Published in
Energy and AI

Citation (APA)
Bugaje, A.-A. B., Cremer, J. L., Sun, M., & Strbac, G. (2021). Selecting decision trees for power system security assessment. *Energy and AI*, *6*, 1-10. Article 100110. https://doi.org/10.1016/j.egyai.2021.100110

Energy and AI 6 (2021) 100110

Contents lists available at ScienceDirect

Energy and AI

journal homepage: www.elsevier.com/locate/egyai

Selecting decision trees for power system security assessment

Al-Amin B. Bugaje [a], Jochen L. Cremer [b,*], Mingyang Sun [c], Goran Strbac [a]

[a] Department of Electrical & Electronic Engineering, Imperial College London, London, SW7 2AZ, UK
[b] Department of Electrical Sustainable Energy, TU Delft, Mekelweg 5, 2628 CD Delft, Netherlands
[c] Department of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Keywords:
Dynamic security assessment
Machine learning
Decision trees
ROC curve
Cost curves
Cost sensitivity

ABSTRACT

Power systems transport an increasing amount of electricity, and in the future, involve more distributed renewables and dynamic interactions of the equipment. The system response to disturbances must be secure and predictable to avoid power blackouts. The system response can be simulated in the time domain. However, this dynamic security assessment (DSA) is not computationally tractable in real-time. Particularly promising is to train decision trees (DTs) from machine learning as interpretable classifiers to predict whether the system-wide responses to disturbances are secure. In most research, selecting the best DT model focuses on predictive accuracy. However, it is insufficient to focus solely on predictive accuracy. Missed alarms and false alarms have drastically different costs, and as security assessment is a critical task, interpretability is crucial for operators. In this work, the multiple objectives of interpretability, varying costs, and accuracies are considered for DT model selection. We propose a rigorous workflow to select the best classifier. In addition, we present two graphical approaches for visual inspection to illustrate the selection sensitivity to probability and impacts of disturbances. We propose cost curves to inspect selection combining all three objectives for the first time. Case studies on the IEEE 68 bus system and the French system show that the proposed approach allows for better DT-selections, with an 80% increase in interpretability, 5% reduction in expected operating cost, while making almost zero accuracy compromises. The proposed approach scales well with larger systems and can be used for models beyond DTs. Hence, this work provides insights into criteria for model selection in a promising application for methods from artificial intelligence (AI).

1. Introduction

A power system is a network of transmission equipment that facilitates the transportation of electrical energy, typically from mega sources (generators) to large sinks (loads). To maintain the equilibrium of the system, the generator output must always equal load demand. The system operator is responsible to prevent blackouts and ensure the security of the power system by conducting security assessments. Security assessments measure a power system's vulnerabilities to faults and equipment failure. To maintain equilibrium, typically, these generators are fossil fuel-based and consist of prime-movers that automatically and in real-time adjust the generator output to match the demand. Consequently, the power system can, on most occasions, accommodate the uncertainty in energy demand via increasing or reducing the generator output, as the case may be. However, the introduction of large amounts of renewable energy sources in the generation mix is accompanied by power electronics devices that make real-time controlling the generator output difficult.

In the future, power systems must integrate large-scale renewable energy. A critical future change for power system operation is the uncertainty in energy generation, and this is specifically challenging for the management of the reliability of the system [1]. The reliability of the system is the ability to supply electricity to the end-users with sufficient enough probability (adequacy) and to withstand imminent disturbances/contingencies without interruption of service (security) [2]. The adequacy measures the reliability of the system over a long period (around months). In real-time operation, the (static-) security is typically assessed and controlled.

1.1. Security assessment

In the analysis of the security, typically, two parts are separated: the static and the dynamic security analysis. In static analysis, the focus is on whether the system operating condition (OC) fulfils all physical limits in the post-fault state (OC c and d in Fig. 1). Dynamic security

[*] Corresponding author.
E-mail addresses: abb18@imperial.ac.uk (A.-A.B. Bugaje), j.l.cremer@tudelft.nl (J.L. Cremer), mingyangsun@zju.edu.cn (M. Sun), g.strbac@imperial.ac.uk (G. Strbac).

https://doi.org/10.1016/j.egyai.2021.100110
Received 3 June 2021; Received in revised form 26 July 2021; Accepted 31 July 2021
Available online 8 August 2021

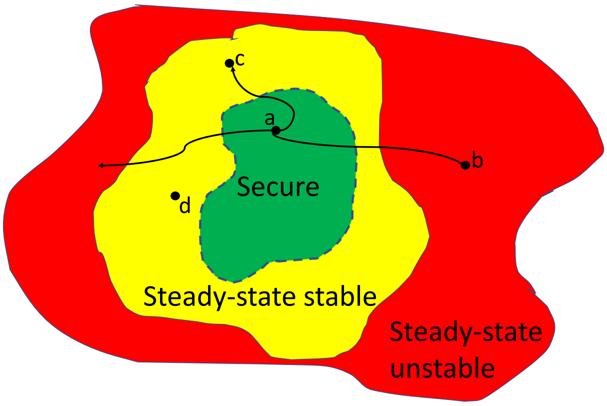


Fig. 1. Different post fault trajectories from an operating condition (OC) (a), to an unstable OC (b), a static and dynamically secure OC (c), a static secure but dynamically insecure OC (d) [3].

analysis focuses on whether the system survives the transition from the pre-disturbance to the post-disturbance condition (OC c in Fig. 1). This dynamic security analysis is more difficult than the static analysis, and hence, stronger static security limits are often selected in place of dynamic security [4]. However, operations can be more efficient if DSA is considered in real-time.

The analysis of dynamic security involves the study of various types of system-wide stability phenomena, such as rotor angle, frequency, or voltage stability [5]. The analysis of stability can be challenging as the power system is a nonlinear system, and numeric analysis including time-domain simulations would be required [5,6], for instance for voltage stability or rotor stability. This analysis via time-domain simulations is challenging in real-time operations as the simulations require significant computational capacity [7]. This is computationally challenging as each possible disturbance needs consideration as event-type perturbations, and many possible operating conditions need assessing, requiring a separate time-domain simulation.

1.2. The machine learning approach to security assessment

The machine learning approach to security assessment is to predict the outcome of the stability analysis [8]. This prediction can replace the analysis itself, and the key benefit is that the prediction is instantaneously available. As this benefit is promising, many variations of the machine learning approaches were proposed [9]. The most common approach is to use a binary classifier as a model, which subsequently predicts whether an operating condition is stable or unstable (in situations where the entire DSA outcome is used, then secure/insecure).

The machine learning approach (Fig. 2) requires training the model offline before real-time operation and involves these four steps: creating a training database, pre-processing the data, training the machine learning model, and evaluating the trained model. In the first step, the training database is created in a balancing way as done in [10]) or extrapolated from historical observations that may be biased toward secure operating conditions [11]. Subsequently, the pre-fault operating conditions (for instance load scenarios) are assessed with the stability analysis for a single or set of disturbances, and a post-fault metric for stability/security is adopted (e.g., as in [12]). The combination of pre-fault operating conditions and the post-fault metric makes up the training database. In the second step, the data is pre-processed for the concurrent training of the model. The database is analysed to select or extract relevant features for reducing the number of dimensions [13] and representing the information in a lower dimension [14]. This pre-processing, for example, the Synthetic Minority Oversampling Technique (SMOTE) [15], can also balance the database as often more secure than insecure data is available. In the third step, the machine learning model is trained. DTs are often used because of their high interpretability, which is crucial for such a critical task as security assessment, where operators require a manual inspection to understand and rely on these machine learning models. Typically one DT model is trained per disturbance as the stability/security is different for each disturbance [8,16]. In the fourth step, the models are evaluated, selected, and eventually updated. This is discussed in the following section.

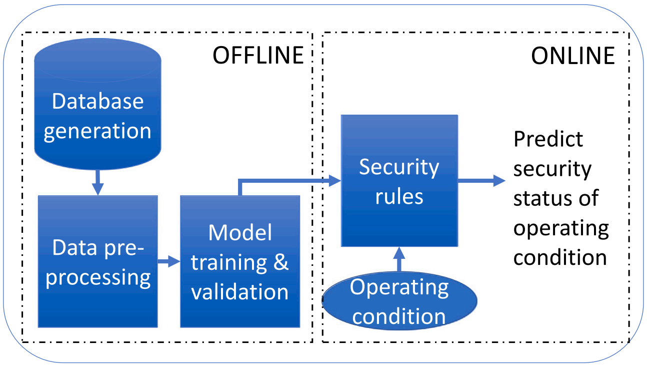


Fig. 2. Overall workflow for data-driven security assessment.

1.3. Selection of the machine learning model

The selection and evaluation of the DT model involve finding the model with the highest performance out of a large set of trained models. Typically, the performance is measured by testing how the model performs on data that is not part of the training. For instance, a testing set is used to compute the testing error (the ratio of inaccurate predictions). Other performance metrics as the F_1 score [17] allow for a harmonic balance of the precision and recall for different errors [18] as used in [19], or the G-mean score that computes the geometric mean as used in [20]. Also, graphical approaches can be used to select models, such as the precision–recall (PR) curve [21] or receiver operating characteristic (ROC) curve [22] as applied to DSA in [23,24]. However, selecting a model based on a single criterion may be sub-optimal. In DSA, the following three need to be considered:

Firstly, predicting errors for the different classes can have various impacts. A missed alarm is much more severe than a false alarm. Missing an alarm can result in power blackouts and load shedding that have high expected costs, however, a false alarm may require only preventive and corrective control measures (e.g., generation re-dispatch) to be taken that are significantly cheaper. Considering the different impacts of errors is important when training the model specifically if the training database is imbalanced in the classes [9]. This difference in the impact of errors renders several performance criteria unsuitable, such as the test error. The ROC curve or F_1 score may be more suitable. However, the expected outage cost to the end-customer should be considered as the performance metric in security assessments [25] and non of these scores allows for directly measuring the performance in terms of expected costs. Computing the outage cost is difficult as it depends on the disturbance and the load condition [25,26]. However, estimates of the costs are considered in cost-sensitive learning by adjusting the decision threshold when predicting with the model [27,28]. Cost-sensitive learning was applied to DSA in combination with ensemble DTs [29], with deep learning [30] and with SMOTE as the imbalance challenge addressed is similar [20].

Secondly, it is crucial to consider the interpretability (and complexity) of the selected model. Models that are high in their complexity are not interpretable for operators [9] and this renders them unsuitable for the application to DSA. Operators responsible for the critical task of security assessments may prefer interpretable DT models in their decision-support tools such that manual inspection remains possible and errors can be identified [16,31]. Therefore, the interpretability of the models needs consideration when selecting models for DSA.

Thirdly, it is also crucial to consider frequent changes in the system. Frequent changes in system parameters may require to change the selection of models. For instance, the weather changes frequently, and with that, the likelihood of contingencies [32,33]. If the probability is high, an operator may select more conservative DT models than at times with low likelihood. In practice, 1000s of models may be used in real-time [16] and a fast, adaptable selection process is needed.

1.4. Contributions

The contribution of this work is to propose a rigorous workflow that considers all three aforementioned specific needs (accuracy, interpretability, and cost sensitivity) to select the best classifier for the application of security assessment, and in addition two approaches for graphical inspection to demonstrate the selection sensitivity to key parameters such as probability and impact of disturbances. The cost-curves [34], that are based on ideas from cost sensitive-learning [27] are introduced. This workflow allows for the first time, selecting the best machine-learned DT model for DSA in response to frequent changes of expected costs or likelihood of contingencies, as well as their uncertainty. The proposed workflow is fast, simple, and effective in selecting the best models.

The proposed workflow is studied on the IEEE 68-bus system and further extended to the French transmission system. The challenge of selecting models/classifiers based on predictive accuracy is presented, resulting in sub-optimal selections. Subsequently, the benefits of the proposed multi-objective selection procedure are demonstrated. Finally, the cost-curve approach is showcased.

The remainder of the paper is structured as follows. Section 2 presents the three different objectives when selecting a model for DSA. Section 3 presents the proposed workflow and the two graphical approaches that allow considering all three objectives together. Finally, Section 4 presents the case study and Section 5 concludes this work.

2. Objectives in selecting DT classifiers for DSA

In supervised machine learning, the procedure of learning and selecting a classifier starts with defining the models and hyperparameters to study. After these are defined, the range of hyperparameters is typically explored with a cross-validation search procedure to address under- and overfitting of the data. In the cross-validation search, typically many different combinations of values for the hyperparameters are explored, and for each, a classifier is trained using the gini index or entropy in DTs. The result of this exploration of combinations of hyperparameter values is a set of classifiers (one per combination). Subsequently, one classifier of this set must be selected. For this selection, various metrics can be used. Typically, the validation accuracy (an approximation of the testing accuracy) is chosen. As pointed out earlier, choosing the performance of the testing accuracy may not represent the needs for DSA to balance the different costs of inaccuracies, interpretability, and robustness.

2.1. Minimising the effects of inaccuracies

One of the objectives of selecting classifiers is to minimise the effect of inaccurate predictions. Two types of inaccurate predictions exist: missed alarms (false positives) and false alarms (false negatives), and these have different effects/costs. The symbols FP and FN correspond to the absolute numbers of inaccurately predicted operating conditions, of the types of false positives and false negatives, respectively. The specific effects (or impacts, costs) of these inaccurate predictions are C_{FN} for false-negative predictions and C_{FP} for false-positive predictions, where $C_{FP} \gg C_{FN}$ as missed alarms have higher impacts than false alarms. For missed alarms, these involve the expected outage cost to the end-consumer, and for false alarms, these are the expected costs for unnecessary preventive/corrective control actions. Accurate predictions have zero costs, hence $C_{TP} = C_{TN} = 0$ for true negatives (TN) and true positives (TP), respectively. The cost ratio is

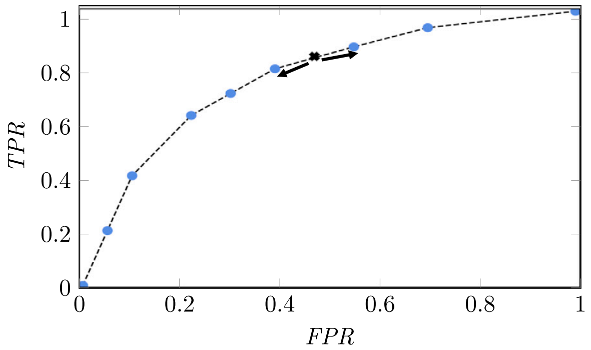


Fig. 3. The ROC curve for evaluating a classifier. Different combinations of TPR and FPR are obtained by varying the decision threshold Z (shown with arrows). The perfect classifier would be in the top left corner having $TPR = 1$ and $FPR = 0$.

$$\gamma = \frac{C_{FP}}{C_{FP} + C_{FN}}. \tag{1}$$

Typically, the objective in binary classification is to minimise the test error $\frac{FP+FN}{FP+FN+TP+TN}$, however, firstly, the test error cannot directly be modelled; hence in the training, an approximation for the test error is typically used (e.g., the training error, entropy). Secondly, by minimising this objective, this imbalance in the cost/impact of the two different types of inaccuracies is not considered. Another way to evaluate and train a classifier in binary classification is to maximise the F_1 score.

The F_1 score equally weights precision and recall, however, it does not consider the different costs of the two classes [17], similarly as the training error and entropy. One approach to consider both inaccuracies is to use the ROC curve. The ROC curve is a graphical approach that shows the true positive rate $TPR = \frac{TP}{TP+FN}$ and the false positive rate $FPR = \frac{FP}{TN+FP}$ [22]. Various combinations of TPR and FPR are computed by varying the threshold Z that the classifier uses for prediction. This threshold Z is used to obtain the predicted class. Initially, the classifier outputs a score $S \in [0,1]$. Subsequently, this score is compared against the threshold Z (the default value is $Z = 0.5$) to determine the predicted class. If the score $S \geq Z$, then the prediction is the positive class $Y = 1$, otherwise negative $Y = 0$. Hence, for a testing set, the combinations of (TPR, FPR) values are computed for varying Zs. Then, these points build the ROC curve.

The ROC curve is used to evaluate classifiers with cost-sensitivity as shown in Fig. 3. Each point corresponds to a single classifier where the decision threshold was varied in $Z = [0, 1]$. It is possible to estimate the optimal $\tilde{Z}^*$ from an effect/cost minimising viewpoint. The objective of minimising costs/impacts of inaccurate predictions is

$$\Sigma = FN * \Pi_+ * C_{FN} + FP * \Pi_- * C_{FP}, \tag{2}$$

where

$$\Pi_- = \frac{N_-}{N_- + N_+} \quad \text{and} \quad \Pi_+ = \frac{N_+}{N_- + N_+} \tag{3}$$

are the two class distributions of positive N_+ and negative N_- points in the testing set [28]. Typically N_+ and N_- are not exactly known, however, Π_+ and Π_- can be assumed to be similar to the distributions in the training set. Note that in DSA, often $\Pi_+ \gg \Pi_-$ is an additional (class) imbalance to $C_{FP} \gg C_{FN}$. Although, C_{FN} and C_{FP} are unknown, estimates could be assumed $\tilde{C}_{FN}$ and $\tilde{C}_{FP}$ and the expected costs are

$$\tilde{\Sigma} = FN * \Pi_+ * \tilde{C}_{FN} + FP * \Pi_- * \tilde{C}_{FP}. \tag{4}$$

Then, the estimated cost ratio $\tilde{\gamma}$ is computed by using Eq. (1) and the estimates $\tilde{C}_{FN}$ and $\tilde{C}_{FP}$. Subsequently, the expected costs from Eq. (4)

$$N\tilde{\Sigma} = \frac{(1-TP) * \Pi_+ * (\frac{1-\tilde{\gamma}}{\tilde{\gamma}}) + FP * (1-\Pi_+)}{\Pi_+ * (\frac{1-\tilde{\gamma}}{\tilde{\gamma}}) + (1-\Pi_+)}. \tag{5}$$

This (normalised) expected cost is minimised when selecting the decision threshold at the estimated cost-optimal point as

$$\tilde{Z}^* = \frac{\Pi_- * \tilde{C}_{FP}}{\Pi_- * \tilde{C}_{FP} + \Pi_+ * \tilde{C}_{FN}}. \tag{6}$$

2.2. Maximising the interpretability

The second objective of a classifier for DSA is to maximise interpretability. If the learning approach is interpretable, then human experts, here the system operator, can build trust in using these approaches. The classifier is interpretable when the learned model can be understood and offers insights into the process of how a prediction is being made. This requires models and data that are non-complex. For instance, the model complexity can be described by the type of parametrisation, the number of hyperparameters, and the number of features. In DTs, the type of parametrisation is a linear splitting scheme. As this linear splitting scheme is not complex, DTs are known for their interpretability. In DTs, the number of hyperparameters can be measured as the number of nodes/splits as each node involves a single hyperparameter. Hence, the number of nodes is a measure for the model complexity (and inversely interpretability). For trading-off the model complexity with accuracy, regularisation is typically used in training for the purpose of avoiding overfitting. Hence, in a similar way accuracy and interpretability can be traded-off as in [31].

2.3. Maximising the robustness of classifiers

The third objective is to maximise the robustness of the classification decisions under uncertainties in the input parameters, as costs/impacts and probabilities of contingencies. The cost/impact of contingencies on the system $\tilde{C}_{FN}$, $\tilde{C}_{FP}$ and $\tilde{\gamma}$ may be wrongly estimated and these uncertainties result in sub-optimal prediction decisions. Some classifiers may be more prone to these uncertainties than others, hence, it is proposed to consider this uncertainty to select classifiers that are less prone.

The objective is to minimise Eq. (2). However, actually, Eq. (4) is minimised as the cost ratio γ is unknown and the estimate $\tilde{\gamma}$ is used. The impact of the uncertainty in these estimates can be studied in a sensitivity analysis involving comparing the expected normalised costs $N\tilde{\Sigma}$ from Eq. (5) with the normalised actual costs

$$N\Sigma = \frac{(1-TP) * \Pi_+ * (\frac{1-\gamma}{\gamma}) + FP * (1-\Pi_+)}{\Pi_+ * (\frac{1-\gamma}{\gamma}) + (1-\Pi_+)} \tag{7}$$

that is similarly derived as Eq. (5). These two costs $N\tilde{\Sigma}$ and $N\Sigma$ are compared for various errors $\Delta\gamma$ in the cost-ratios, where $\tilde{\gamma} = \gamma \pm \Delta\gamma$. This sensitivity analysis involves computing the false negatives and false positives $(FN, FP) = f(Z)$ for a given test set, where f is the prediction function of the classifier and the decision threshold Z is varied. Then, the normalised actual costs $N\Sigma$ are computed from Eq. (7) using γ and the normalised expected costs $N\tilde{\Sigma}$ from Eq. (5). Subsequently, the differences in these costs are compared, and the various estimation errors $\Delta\gamma$ are studied to understand the sensitivity of this cost difference. These studies can be repeated for multiple classifiers and various decision thresholds to find the best combination of classifiers and threshold Z most insensitive (robust) against uncertainties in the costs.

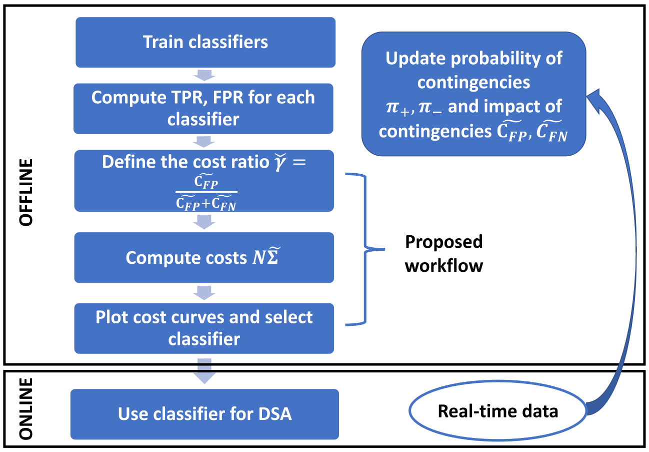

OFFLINE
Train classifiers
Compute TPR, FPR for each classifier
Define the cost ratio $\breve{\gamma} = \frac{\widetilde{C_{FP}}}{\widetilde{C_{FP}}+\widetilde{C_{FN}}}$
Compute costs $N\widetilde{\Sigma}$
Plot cost curves and select classifier
Update probability of contingencies π_+, π_- and impact of contingencies $\widetilde{C_{FP}}, \widetilde{C_{FN}}$
Proposed workflow
ONLINE
Use classifier for DSA
Real-time data

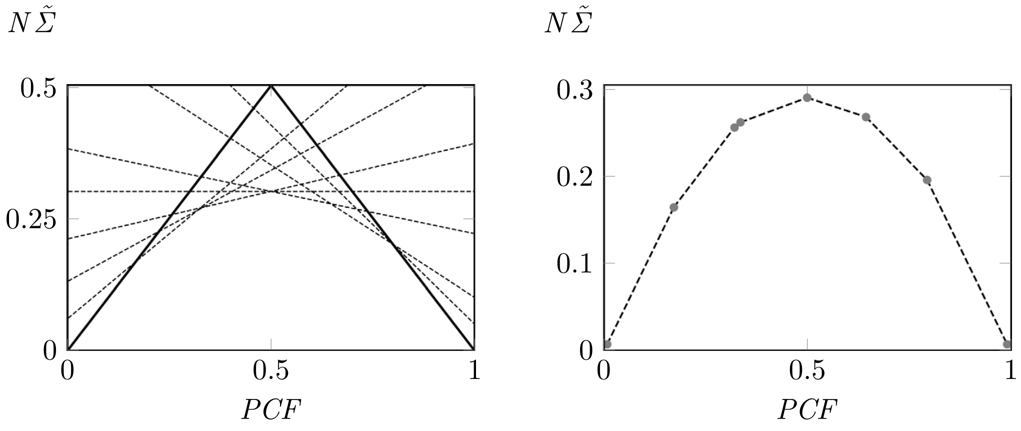
Fig. 4. Proposed cost-curve workflow to select the best DT classifier offline. The selection can be updated with real-time data.

3. Multi-criteria selection of classifiers

This section presents the proposed rigorous workflow to select the best DT classifier, and subsequently, introduces two approaches for graphical inspection to illustrate and show the selection sensitivity to estimated parameters. The first graphical approach modifies the ROC curve and allows considering the first two objectives: to minimise inaccurate predictions and maximise the interpretability as shown in Fig. 11. The second approach modifies the cost-curves, as shown in Fig. 6. The proposed modification to the second approach allows trading-off on all three objectives.

The proposed workflow has two parts: (i) offline training and (ii) online selection. The first part is the proposed offline workflow to train and prepare classifiers for application in the online selection workflow. The offline workflow of using the proposed cost-curve approach is illustrated in Fig. 4. Initially, many classifiers are trained by varying some (hyper-)parameters, such as the DT depth, and each classifier training follows cross-validation resulting in $\mathcal{N}$ candidate DTs, $\Omega = \{C_i^{(p)}, \ \forall \ i = 1, 2, \ldots, \mathcal{N}\}$, and $p \in \mathfrak{R}$ is the number of DT nodes. Subsequently, the task is to select the best DT classifier from this set Ω according to the introduced criteria of Section 2.

The second part is the proposed online selection workflow to consider realtime information when selecting a classifier. The probabilities of contingencies Π_+, Π_-, the expected cost of contingencies $\tilde{C}_{FP}, \tilde{C}_{FN}$, and the probability cost function's range of interest $[PCF^L, PCF^U] \in [0, 1]$ are updated with real time data. The range $[PCF^L, PCF^U]$ represents various expected combinations of contingency probabilities and expected contingency costs. Then, for each DT $C_i^{(p)}$, the normalised expected costs $N\tilde{\Sigma}$ are computed with Eq. (5) within the range of $[PCF^L, PCF^U]$ at K user-defined, equidistant steps. The average normalised expected cost is then computed as $\overline{N\tilde{\Sigma}} = \frac{\sum_{i=1}^{K} N\tilde{\Sigma}}{K}$, where $\overline{N\tilde{\Sigma}}$ measures on average the expected impact (e.g. cost of loss of load) of wrongly classifying the security status of an operating condition. Subsequently, the average normalised actual cost $\overline{N\Sigma}$ ($N\Sigma$ computed with Eq. (7)) is calculated assuming $\tilde{\gamma} = \gamma \pm \Delta\gamma$ within the same $[PCF^L, PCF^U]$ range and K equidistant steps, where $\overline{N\Sigma}$ measures on average the actual impact of wrongly classifying the security status of an operating condition. This step is done to compare the sensitivity of a classifier $C_i^{(p)}$ to estimation errors $\Delta\gamma$ of the cost ratio $\tilde{\gamma}$ that is assumed. The number of steps K is selected by the user, and the more steps are selected, the better the resolution of both actual $\overline{N\Sigma}$ and expected $\overline{N\tilde{\Sigma}}$ costs. Finally, the proposed, optimal DT $C_i^{(p)}$ is the one with the minimum average relative costs across the range of $[PCF^L, PCF^U]$ as $\mathcal{G}(\Sigma) = \frac{\overline{N\Sigma}-\overline{N\tilde{\Sigma}}}{\overline{N\Sigma}}$, and DT nodes $p \leq a$, where a is a user-defined criterion,



(a) Construction of cost-curve

(b) Final cost-curve

Fig. 5. Construction of cost-curves: In (a), each dashed linear curve corresponds to one blue point in the ROC curve Fig. 3. The combined minimal expected cost $N\tilde{\Sigma}$ of (a) is where the selected Z equalled the cost-optimal decision threshold $\tilde{Z}^* = PCF$ and is shown in (b).

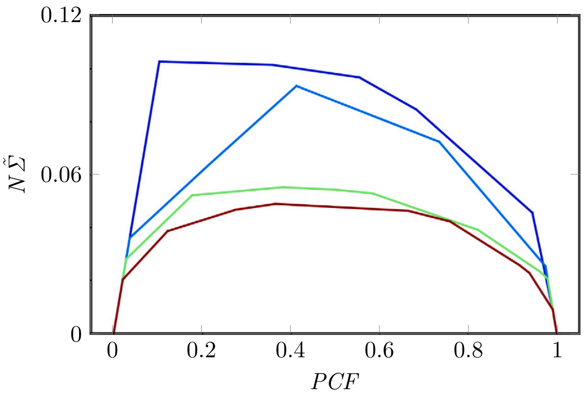


Fig. 6. Proposed cost-curve based selection considering multiple criteria. The lime classifier has the best trade-off in terms of interpretability and expected cost in the relevant interval $0.8 \leq PCF \leq 1$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

and keeping a small aims for high interpretability. This is the optimal DT as the contingency probabilities and contingency expected costs are uncertain.

The final cost-curve provides a graphical illustration of the sensitivity of selecting the best classifier to minimise costs of missed and false alarms considering inaccurate estimations of the costs $\tilde{C}_{FP}, \tilde{C}_{FN}$ and probabilities of contingencies Π_+, Π_-. The DT $C_i^{(p)}$ whose cost-curve has the least variation in the range $[PCF^L, PCF^U]$ is the most suitable DT and would be the ideal DT from the perspective of cost uncertainty. The two proposed graphical approaches are introduced as follows.

3.1. Visual inspection using the ROC curve

A colour bar is used for the interpretability on the ROC curve, and interpretability of a DT model (here, in terms of number of nodes) correlates inversely with the DT model complexity. The ROC curve allows to study the performance of a binary classifier when adjusting the decision threshold Z as illustrated in Fig. 3. The advantage of the ROC curve is the visual ability to compare classifiers across ranges of various Zs instead of a single point comparison that does not allow for variability, such as in selecting based on computing the cost-optimal $\tilde{Z}^*$ using Eq. (6). Classifier comparison using the ROC curve starts by drawing each classifier's ROC curve. Subsequently, the cost-optimal decision thresholds $\tilde{Z}^*$ are marked. An example is presented in Fig. 11.

In this example, the DT classifier corresponding to the blue curve is the best in terms of interpretability (fewer nodes), and the brown is the best in terms of accuracy (closest to the top left corner). However, when considering these two objectives in the ROC curve together, the best trade-off is the classifier corresponding to the cyan curve. Although the interpretability is slightly worse than the blue classifier, the TPR is significantly better (almost 0.05 higher). Also, the classifiers represented by the brown and green curves only offer marginal improvements in terms of TPR. However, they are worse in terms of interpretability. Therefore, cyan curve is the best classifier in this example. This selection procedure can be quickly and visually performed by an operator to trade-off the cost-optimality, accuracy, and interpretability using a single graphical approach.

3.2. Proposed cost-curve approach for graphical inspection

The proposed approach modifies cost-curves to include information on the sensitivity of inaccurate estimations of the costs/impacts. The cost-curve shows the normalised expected cost $N\tilde{\Sigma}$ from Eq. (5) with varying probability cost function PCF. This is the main difference to the ROC curve where the expected costs of inaccurate predictions are not directly presented such as in the cost-curve. The proposed modification of the cost-curve allows selecting the best classifier by considering all three aforementioned objectives.

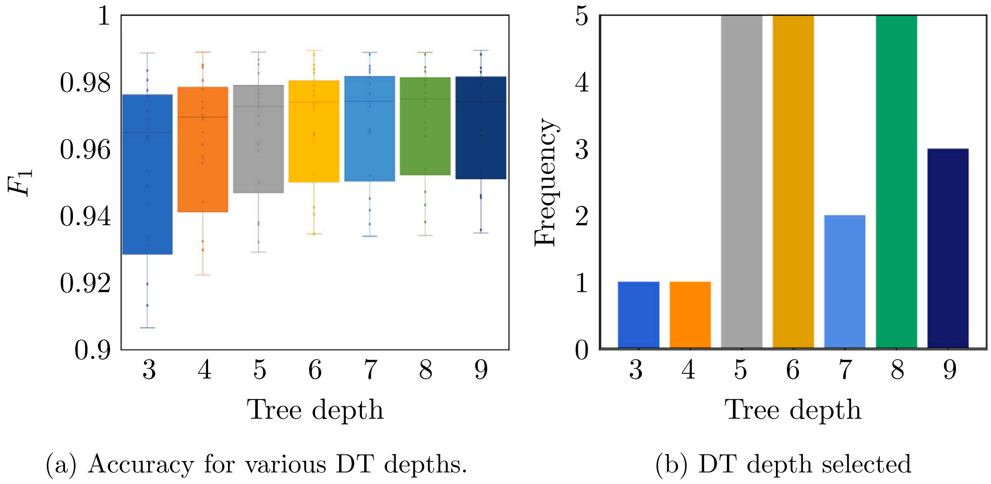
The construction of the cost-curve starts with the ROC curve of the classifier. The ROC curve is constructed from a set of (TPR, FPR) points corresponding to applying different decision thresholds Z to the score-output S of the classifier as illustrated in Fig. 3. Subsequently, the cost-curve aims to investigate the normalised expected costs for each of these points dependent on changes in $\tilde{C}_{FN}$, $\tilde{C}_{FP}$, Π_- and Π_+. Changes in these are considered by using a single parameter, the probability cost function, defined as:

$$PCF = \frac{\Pi_- * \tilde{C}_{FP}}{\Pi_- * \tilde{C}_{FP} + \Pi_+ * \tilde{C}_{FN}}. \tag{8}$$

Subsequently, the normalised expected cost from Eq. (5) are

$$N\tilde{\Sigma} = (FP - FN) * PCF + FN, \tag{9}$$

where FP and FN can be directly computed from the ROC values TPR and FPR. Then, Eq. (9) is the linear equation connecting the points $(PCF, N\tilde{\Sigma}) = (0, FN)$ and $(PCF, N\tilde{\Sigma}) = (1, FP)$. The constructions of the lines corresponding to the blue points in Fig. 3 are presented in Fig. 5(a). The lower envelope of the cost-curve is the minimum costs that can be obtained. This lower envelope represents selecting the cost-optimal $\tilde{Z}^*$ where the Z applied to the classifier equalled



(a) Accuracy for various DT depths. (b) DT depth selected

Fig. 7. The predicting performance of classifiers with 7 different DT depths. The box plots in (a) show min, max, interquartiles, and median for $22 \times 7 = 154$ trees. Using (a), the 22 classifiers with the highest F_1 score are selected and the selection-frequencies are in (b).

$Z = \tilde{Z}^* = PCF$. This final minimal cost-curve is presented in Fig. 5(b). The cost-curve shows the classifier performance across all cost-class distributions whereas the ROC curve allows presenting only a single cost-optimal point. Subsequently, the proposed approach is to use cost-curves for selecting the best classifier along with the relevant range of cost-class distributions as shown in Fig. 6 to ensure the best selection according to the discussed multiple objectives.

The sensitivity of errors in estimating the cost ratio $\tilde{\gamma}$, on the selection based on cost-curves can be studied by computing the corresponding actual cost-curve for $N\Sigma$ based on the actual cost γ assuming an error $\Delta\gamma$ as pointed out in Section 2.3. Firstly, recall that the estimated cost ratio $\tilde{\gamma}$ influences the choice of $\tilde{Z}^*$, and that the cost-curve is defined by $Z = \tilde{Z}^*$. If the choice of estimated cost ratio $\tilde{\gamma}$ differs from the actual cost ratio γ, then the normalised actual cost can be computed based on the actual value of γ and can be presented similarly as the estimated cost-curve. The PCF values do not differ for the normalised expected and actual costs as the threshold $\tilde{Z}^*$ is based on the expected cost ratio of $\tilde{\gamma}$. This modification of the cost-curve allows for considering the cost/impact of estimation errors in $\tilde{\gamma}$ and represents another advantage over the ROC curve. Subsequently, the proposed cost-curve approach can consider interpretability using colour schemes in the same way as in the ROC curve approach. Consequently, this approach considers all 3 objectives as criteria for visual inspection.

4. Case study

In this case study, firstly, the challenge of sub-optimal selections when using a single objective in selecting classifiers is studied. Secondly, the effectiveness of the proposed workflow that considers multiple objectives concurrently to select the best DT classifier is studied, with visualisation using the two proposed graphical approaches. Thirdly, the proposed workflow is studied when considering the uncertainty of the cost-estimations. Finally, the computational times of the proposed workflow is studied and the limitations discussed. Finally, the limitations of the proposed approach are discussed.

4.1. Test system and assumption

The case study is mainly carried out on the IEEE 68-bus system and the scalability is demonstrated in a study of the French transmission system.

The first data set for this case study was generated using the network data from the IEEE 68-bus system [35]. $N_D = 12000$ operating conditions were sampled as follows. The active loads were sampled from a multivariate Gaussian distribution (via Monte Carlo sampling), and the correlation was assumed to follow Pearson's correlation with a correlation coefficient of 0 75. The distribution was converted to a marginal Kumaraswamy(1.6, 2.8) distribution using the method of inverse transformation. The AC-model of the network was used to compute the active and reactive powers of generators to ensure feasible operating conditions. This sampling process of the IEEE 68-bus system results in 12000 samples, where each describes the operating condition of the power system in a steady-state. Then, all phase angles and voltages, reactive and active power flows, and the reactive and active power injections were used to construct the feature vector $X \in \mathcal{R}^{N_D \times N_F}$, where $N_F = 438$ was the number of features. For each of these 12000 operating conditions, the transient stability response to faults was simulated using MATLAB Simulink. $N_C = 22$ different three-phase line outages were simulated as event-type events on the pre-fault steady-state condition and the faults are cleared after 0.1 s. The simulation time was 10 s on a standard desktop computer. If at any point in time the difference of any phase angles of the generators was larger than 180°, then the operating condition for that particular contingency was considered as unstable, and the corresponding element of the label matrix $Y \in \{0, 1\}^{N_D \times N_C}$ was set at 0 for unstable, otherwise stable 1. In total $N_D \times N_C = 12000 \times 22 = 264000$ simulations were performed as each of the 22 contingencies need to be simulated independent.

The second data set of the French transmission system was used. The French transmission system had 1955 transmission lines, 798 transformers, 1886 buses, 411 generators and 127 shunt elements. This data-set consisted of $N_D = 16722$ operating conditions in a feature vector $X \in \mathcal{R}^{N_D \times N_F}$, where $N_F = 35873$ is the number of features. To generate a single data point required 56 s time on a computer cluster [16] and 1980 different contingencies were analysed where each required a single time-domain simulation. Subsequently, the time-domain simulations were assessed and 9 reliability metrics were computed, including overload, loss of synchronisation, over/under-voltages, small-signal stability, transient stability, et cetera. More details can be found in [16, 36]. This second data set was used to demonstrate the scaling of the proposed approach.

The subsequent processes of the training workflows (feature selection, DT training, et cetera.) were carried out on a Dell XPS 13 9360 running an Intel(R) Core(TM) i5-8250U processor with 8 GB installed RAM. DTs were trained with the CART algorithm [37] from the package *scikit-learn* 0.18.1 [38] in Python 3.5.2. The default training settings were selected except using gini impurity instead of entropy to measure the quality of the splits. The data-set was split into training/testing sets in ratio of 75 %/25 %. The feature variable X was used and the labels Y were used as the input for the training of the classifier, however, for each contingency, a single DT was trained (in total 22

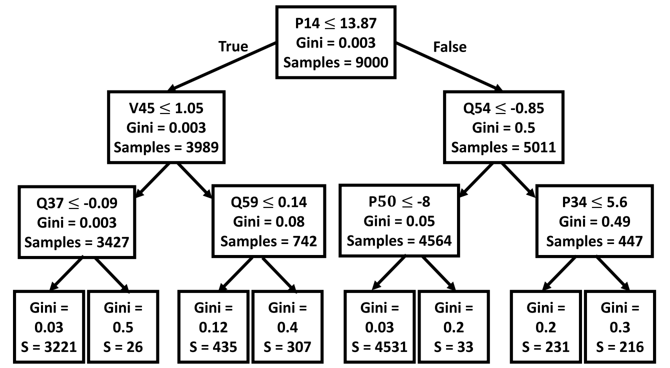


Fig. 8. Tree structure of a DT with DT depth of 3. P and Q are the active and reactive power-flow between buses, and V is the voltage at the bus.

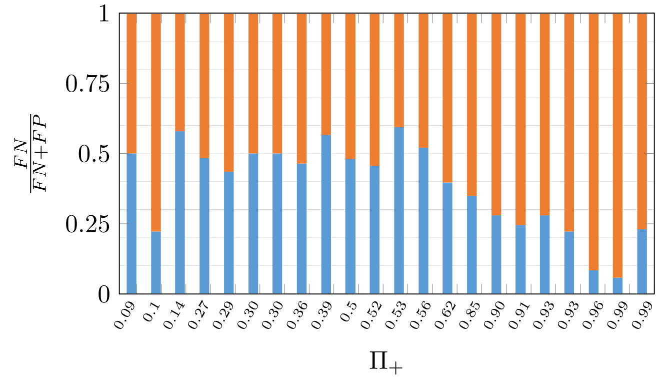


Fig. 9. Ratio between missed (🟧) and false(🟦) alarms for the 22 DTs without considering the impact of different costs. Π_+ is the class ratio of positive conditions.

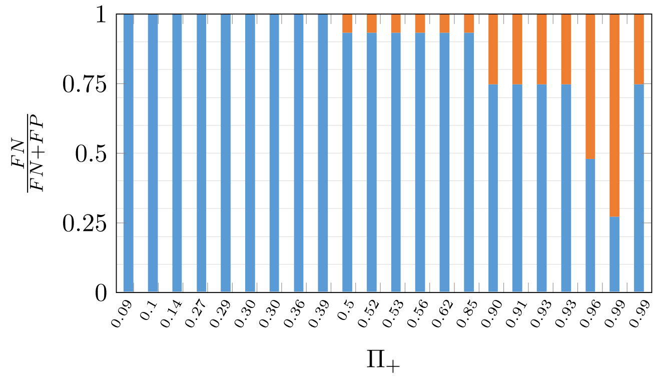


Fig. 10. Ratio between missed (🟧) and false (🟦) alarms for the 22 DTs by considering the impact of different costs.

DTs). 5-fold cross-validation was applied to address under-/overfitting. Subsequently, the Platt method was used to calibrate the score-output S of the classifier [39].

4.2. Selections based on single criteria

The first study illustrates the effect on the model interpretability when predictive accuracy is used for selecting the model. Typically, the F_1 score or the test accuracy is used for selecting the model. In this study, firstly, DTs were trained for the depths $\{1-20\}$ for each of the 22 contingencies. Subsequently, 5-fold cross-validation was used to select the best DT depth based on the highest F_1 score. Fig. 7(a) presents the F_1 accuracy values for all different tree depths involved in this study showing that an increase in tree depth on average results in higher F_1 scores, however, over-fitting occurred for larger depths than 9 and no tree was selected with a depth larger than 9. Fig. 7(b) shows the exact breakdown of the final selected classifiers and most of the selected 22 DT depths were around larger depths of $[5, 9]$. However, the DT structures of larger trees are not easily readable (interpretable). For example, a DT with DT depth $= 3$ (Fig. 8) has 15 nodes. Conversely, a DT with DT depth $= 9$ has 100 nodes. When focusing only on the predicting accuracy, the user may select the DT with depth $= 9$ as the F_1 score of 0.975 outperforms the tree with depth $= 3$ having an F_1 score of 0.96. However, when focusing only on the interpretability, the operator may select the tree with DT depth $= 3$ with a lower F_1 score of 0.96. These two criteria are contrasting and require a suitable trade-off.

The second study investigates the impact when considering neither the difference in expected costs nor the class imbalances. Typically classifiers are trained to minimise the test inaccuracy, and this is insensitive to the differences in costs and the imbalance and is therefore prone to more missed alarms than false alarms. The 22 DTs from the first study were used, and their relationship between missed and false alarms was investigated in Fig. 9. In this training procedure, neither the impact of costs nor class imbalances were considered. It shows: when the imbalance is large, for instance when many more positive than negative operating conditions are in the database (toward high Π_+), the share of missed alarms increases significantly. This is not in favour of operators, as typically the expected cost for missed alarms is large than that of false alarms $C_{FP} \gg C_{FN}$, hence the operators aim to avoid missed alarms. Here, it is assumed that $C_{FP} : C_{FN} = 2 : 1$. However, when the 22 DTs and a shifted decision threshold $\tilde{Z}^*$ from Eq. (6) was used as described in Section 2.1, then the ratio of missed alarms decreased significantly as demonstrated in Fig. 10. Ideally, when considering the costs and imbalances, the ratio of missed and false alarms would have been constant for all different contingencies with different imbalances. However, Fig. 10 shows an increase of missed alarms toward high class imbalances of high Π_+. The reason is that these trained classifiers for high Π_+ have more knowledge available on positive than on negative operating conditions and are therefore more accurate on positive conditions. Cost-sensitive learning aims to address that imbalance, however, can never fully address it. This highlights another trade-off that needs to be made between minimising test inaccuracy and the impact of different costs. These two studies showed that considering a single criterion is insufficient when selecting a classifier.

4.3. Multi-criteria selection with modified ROC approach

In this study, the proposed selection workflow is investigated using the modified ROC approach. This workflow allows for a visual inspection of the accuracy performance and interpretability at the same time when selecting the classifier. Firstly, a variety of classifiers were trained with tree depths of $\{2, 3, 4, 5\}$. To select the best classifier, the ROC approach was applied as follows: the TPR and FPR values for each tree were obtained by varying the decision threshold $\tilde{Z}^*$ within $[0, 1]$. The TPR and FPR values are obtained from the test set and each combination TPR and FPR values represent the classifier being used with a different decision threshold. Subsequently, their values were plotted for each of the 4 trees in Fig. 11. The colour spectrum shows the different levels of interpretability (or DT depths). The blue tree is most interpretable while the brown tree is the least interpretable. Subsequently, the cost-optimal points for each tree were marked with the X symbol for an assumed cost ratio $\tilde{C}_{FP} : \tilde{C}_{FN} = 2 : 1$. These points represent the cost-optimal use of the given tree with the specified cost ratio. The ROC approach can be used to select the best among these 4 cost-optimal DTs. For instance, the lime and cyan curves (and points) are noteworthy. Both classifiers have similar values for TPR and FPR. However, the classifier represented in the cyan colour offers

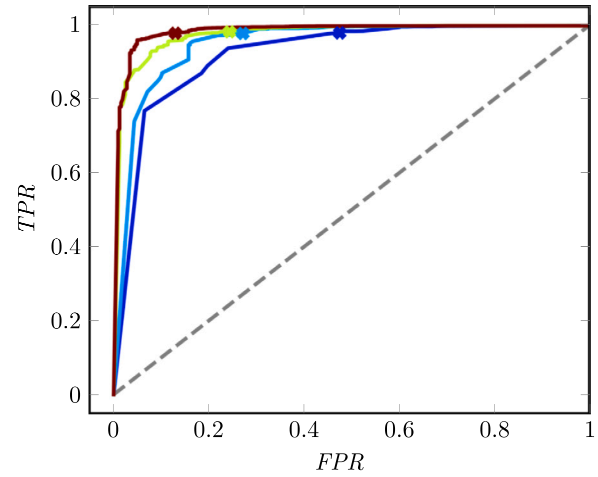


Fig. 11. Selection based on modified ROC curves for multiple criteria. The DT classifier in cyan has the best trade-off in terms of cost-optimality (x on the curve) and interpretability (the number of DT-nodes from 0 to 50, where 0 is on the left side of the colour scheme ▬). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

higher interpretability and is the final classifier. It would be difficult to see the similarity of these two curves by analysing the predicting performances based on data and hence the proposed selection is more robust as the entire curve can be assessed while keeping the focus on the comparison of the cost-optimal points. The proposed approach increased interpretability by 82 % (on average 30 DT nodes in the proposed approach and 54 nodes when focusing only on accuracy) while the accuracy slightly decreased (0.960 in the proposed approach and 0.964 otherwise).

4.4. Multi-criteria selection with cost-curves

In this study, the proposed selection workflow with graphical inspection using the cost-curve approach is in focus. The cost-curve approach allows for visual inspection of the accuracy performance, the interpretability, and quantifies the expected costs of misclassifications. The same DT classifiers with tree depths of $\{2, 3, 4, 5\}$ were used as in the previous study. The construction of the cost-curves followed the steps described in Section 3.2. The four resulting cost-curves are presented in Fig. 6. Showing these curves allows comparing classifiers across different intervals of costs instead of a single point-wise comparison as in the ROC curve approach. This is useful as the class-cost distribution PCF can change frequently (PCF is a function of the likelihood of contingency and the outage costs). In this example, the relevant region is $0.8 \leq PCF \leq 1$ as this is where high class imbalances are (toward large Π_+). The brown classifier has the lowest expected costs Σ and the blue classifier has the highest interpretability. The expected cost of the brown and lime classifiers are similar within the relevant region, however, the lime classifier has a steeper curve for PCF values lower than around 0.9. Also here, an operator may select the lime classifier as it represents a good trade-off between interpretability and normalised expected cost in the relevant interval. This proposed novel cost-curve approach can be used by an operator to study cost-class distribution intervals, and goes beyond the point-wise comparison that would be possible with ROC-curves or other data-based comparisons. These are additional insights to the model-selection.

4.5. Reduction of loss of load costs under uncertainties

In this study, the proposed selection workflow is investigated when considering uncertainties in estimates of the cost parameter $\tilde{\gamma}$. An error of 20 % was assumed, where $\Delta\gamma = \pm 0.2\gamma$ for the two classifiers from the previous study (lime and brown of Fig. 6). The actual normalised cost for loss of load $N\Sigma$ using γ for the various PCF values based on $\tilde{\gamma}$ were computed for three cases as described in Section 3.2, where the baseline is $\Delta\gamma = 0$ representing no error and the two error cases $\Delta\gamma = \pm 0.2\gamma$. Subsequently, the three cases are presented in Fig. 12. In the baseline case, the normalised expected cost $N\tilde{\Sigma}$ for loss of loads equals the normalised actual costs $N\Sigma$ of loss of loads. In cases with errors, the actual costs $N\Sigma$ deviate from the expected costs $N\tilde{\Sigma}$ in both classifiers. In the relevant region, $0.8 \leq PCF \leq 1$, the brown classifier shows a higher variability in the impact of parameter estimations. This is also demonstrated by analysing the average costs $\overline{N\Sigma}$, $\overline{N\tilde{\Sigma}}$, and standard deviations $\sigma_{N\Sigma}$, $\sigma_{N\tilde{\Sigma}}$. Table 1 shows the brown classifier has a higher standard deviation than the lime classifier. In addition, the relative change in operating costs for loss of load $\varsigma(\Sigma)$ is lower in the lime classifier 13 % versus 18 % for the brown classifier (in the case $\Delta\gamma = +0.2\gamma$). Thus the lime classifier is more robust against uncertainties, estimation errors and is, therefore, the final selected classifier under this viewpoint.

Table 1
Actual normalised costs $N\Sigma$ (to the basis 10^{-2}) considering errors of 20 % in $\tilde{\gamma}$. The presented values are averages and standard deviations in the interval $0.8 \leq PCF \leq 1$ of Fig. 12.

$\Delta\gamma$	-0.2γ	0	$+0.2\gamma$
Lime classifier (depth 4)	2.7(.7)	3.1(.7)	3.5(.7)
Brown classifier (depth 5)	2.4(.7)	2.8(.8)	3.3(.9)

Table 2
Computation time for the offline and online workflows showing the proposed approach to compute cost curves for classifier selection scales well to large systems.

State	Process	IEEE 68 bus	French System (1886 bus)
Offline	Data generation	40 min	260 h
	Feature selection	(4.0 ± 0.8) s	(8.7 ± 3.8) min
	DT training	(0.40 ± 0.01) s	(2.4 ± 3.5) s
	DT testing	(0.01 ± 0.01) s	(0.03 ± 0.02) s
	Cost curve plotting	(0.06 ± 0.01) s	(0.07 ± 0.03) s
	Selection by operator	5 s	5 s
Online	Prediction	< 0.01 s	< 0.01 s

4.6. Computation time and scalability

The scalability of the proposed selection workflow is analysed both for larger systems and for comparing many DT classifiers.

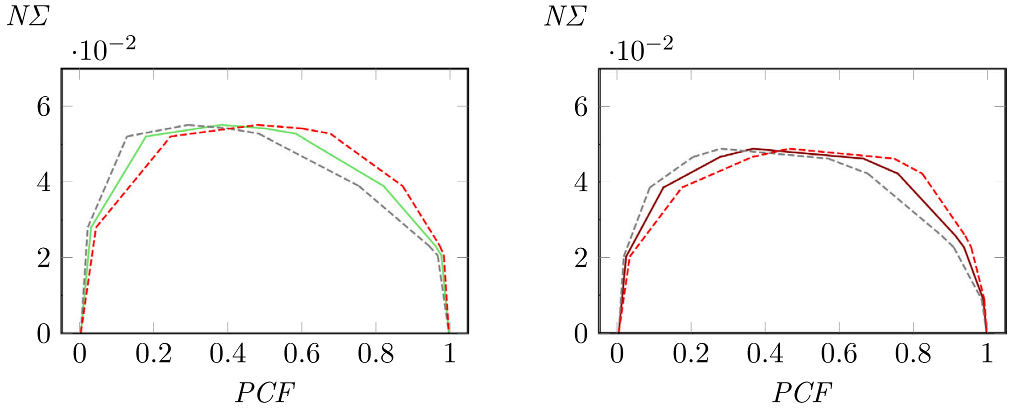
Table 2 shows the computational times for the offline workflows and testing the classifiers online on both the IEEE 68-bus and French systems. The table shows the average time for ten contingencies selected at random that have a balanced class distribution ($\Pi_- \geq 35$ %).

4.6.1. Scalability of proposed selection workflow to larger systems

The proposed selection workflow scales well with the size of the system. Table 2 shows the average time for plotting cost-curves is similar for both the IEEE 68 bus and French systems, 0.06 s and 0.07 s respectively. The data generation and feature selection are, however, dependent on the size of the network.

4.6.2. Comparing many classifiers with cost curves

Table 2 shows the data generation is the most computationally intensive step in the ML workflow with more than 99.9 % of the time. The DT training time, on average 0.4 s for the IEEE 68 bus system, and 2.4 s for the French system, is negligible in comparison. Thus, many DTs can be trained and afterward compared. However, as the proposed selection workflow allows for the comparison of a small number of classifiers at a time (in the order of 5), a pre-selection step can be added. For instance, a metric-based selection approach can be used (example F1-score), to reduce the number of candidate classifiers.



(a) Lime classifier

(b) Brown classifier

Fig. 12. Sensitivity study on errors in estimations of parameter $\tilde{\gamma}$. The actual costs $N\Sigma$ under the error of +20 % (---) and −20 % (---) are presented against the estimated costs $N\tilde{\Sigma}$ that is shown as the baseline. In (a) and (b) are the lime and cyan DTs from Fig. 6. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.7. Discussion

The proposed selection workflow selects security rules with 5 % lower relative operational costs under uncertainties in the potential impacts of contingencies (e.g., loss of load). Additionally, the proposed workflow increases model interpretability by up to 80 %. This proposed workflow is a pivotal step toward manual inspection of security rules and supports operators building up the trust for using these rules in the critical task of DSA. The proposed workflow is fast and adaptive. It takes less than 1 s to plot the cost-curves and around 5 s to select the best security rule. In addition, the workflow allows studying sensitivities on the input parameters when the basis for choosing decisions changes, and in response, adjusts quickly to these changes. Such an adaptive approach for model selection is needed as the future power system is ever-shifting. For instance, the probabilities of contingencies and the impact of faults can change within hours.

A first limitation of the proposed workflow is to conceptualise an intuition for the parameter sensitivities as they do not directly have a physical meaning. The sensitivity serves as a relative comparison between different security rules as a tool to compare them and decide on the best security rule. A second limitation is that the proposed workflow does not support different types of models (e.g., neural networks and DTs), as in this work, interpretability is defined for a single type of model, DTs. In the future, a general definition for interpretability can be developed to select among different types of models.

5. Conclusion

This proposed work showcased a promising application for methods from the field of AI which is DSA for power systems. This work also provides insights into the importance of metrics and criteria to learn models from AI for DSA and beyond. Those insights are transferable from DTs exercised in this work to other AI models. This work focuses on selecting a DT model for power system security assessment. Typically, a single selection criterion, the predictive accuracy is used, resulting in sub-optimal data-driven security rules. As a result, security rules are often not interpretable and can result in many missed alarms. These missed alarms have very high risks and economic costs for system operations. In response, we propose a rigorous selection workflow to consider multiple objectives in the model selection: accuracy, interpretability, and cost-robustness. The workflow increases interpretability by more than 80 % while making minimal compromises in the predictive accuracy. Likewise, the proposed workflow reduces expected relative operating costs by around 5 % with little compromise in the predictive accuracy. Other single-objective-based selection approaches miss such trade-offs, and finding these trade-offs is the key advantage of the proposed workflow. Also, the proposed workflow is fast and adaptive to new situations of system operation. The proposed workflow computes cost-curves within less than 0.1 s, and operators can select the best security rules based on analysing the sensitivity to new situations. This adaptation is crucial as it increases interpretability through visual inspection and offers a high degree of situational awareness to the operators. In the future, this work shall consider selections across different types of models and include a general definition of interpretability.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors were supported by a scholarship funded by the Nigerian National Petroleum Corporation, NNPC, the TU Delft AI Labs Programme, NL, and the research project IDLES, UK (EP/R045518/1).

References

[1] Panciatici P, Bareux G, Wehenkel L. Operating in the fog: Security management under uncertainty. IEEE Power Energy Mag 2012;10(5):40–9.
[2] Kundur P. Power system stability. Power Syst Stab Control 2007. 8–1.
[3] Cremer JL. Probabilistic dynamic security assessment for power system control (Ph.D. thesis), London: Imperial College; 2020.
[4] Capitanescu F. Critical review of recent advances and further developments needed in AC optimal power flow. Electr Power Syst Res 2016;136:57–68.
[5] Kundur P, Paserba J, Ajjarapu V, Andersson G, Bose A, Canizares C, et al. Definition and classification of power system stability $\{IEEE/CIGRE\}$ joint task force on stability terms and definitions. IEEE Trans Power Syst 2004;19(3):1387–401.
[6] Blondel VD, Tsitsiklis JN. A survey of computational complexity results in systems and control. Automatica 2000;36(9):1249–74.
[7] Chiang H-D. Power system stability. Wiley Encyclopedia Electr Electron Eng 2001.
[8] Wehenkel LA. Automatic learning techniques in power systems. Kluwer Academic Publishers; 1998.
[9] Duchesne L, Karangelos E, Wehenkel L. Recent developments in machine learning for energy systems reliability management. Proc IEEE 2020;108(9):1656–76.
[10] Thams F, Venzke A, Eriksson R, Chatzivasileiadis S. Efficient database generation for data-driven security assessment of power systems. IEEE Trans Power Syst 2019;35(1):30–41.
[11] Konstantelos I, Sun M, Tindemans SH, Issad S, Panciatici P, Strbac G. Using vine copulas to generate representative system states for machine learning. IEEE Trans Power Syst 2018;34(1):225–35.

[12] Vanfretti L, Sevilla FRS. A three-layer severity index for power system voltage stability assessment using time-series from dynamic simulations. In: IEEE PES innovative smart grid technologies, Europe. 2014, p. 1–5.

[13] He M, Zhang J, Vittal V. A data mining framework for online dynamic security assessment: Decision trees, boosting, and complexity analysis. In: IEEE PES innovative smart grid technologies. 2012, p. 1–8.

[14] Sun M, Konstantelos I, Strbac G. A deep learning-based feature extraction framework for system security assessment. IEEE Trans Smart Grid 2018;10(5):5007–20.

[15] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. J Artificial Intelligence Res 2002.

[16] Konstantelos I, Jamgotchian G, Tindemans SH, Duchesne P, Cole S, Merckx C, et al. Implementation of a massively parallel dynamic security assessment platform for large-scale grids. IEEE Trans Smart Grid 2016;8(3):1417–26.

[17] Hand D, Christen P. A note on using the F-measure for evaluating record linkage algorithms. Stat Comput 2018;28(3):539–47.

[18] Powers DM. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. Int J Mach Learn Technol 2:1 2011;37–63.

[19] Saner CB, Kesici M, Yaslan Y, Genc VI. Improving the performance of transient stability prediction using resampling methods. In: IEEE international conference on electrical and electronics engineering, 2019, p. 146–50.

[20] Zhu L, Lu C, Dong ZY, Hong C. Imbalance learning machine-based power system short-term voltage stability assessment. IEEE Trans Ind Inf 2017;13(5):2533–43.

[21] Davis J, Goadrich M. The relationship between precision-recall and ROC curves. In: Proceedings of the 23rd international conference on machine learning, 2006, p. 233–40.

[22] Fawcett T. An introduction to $\{ROC\}$ analysis. Pattern Recognit Lett 2006;27(8):861–74.

[23] Moulin L, Da Silva AA, El-Sharkawi M, Marks RJ. Support vector machines for transient stability analysis of large-scale power systems. IEEE Trans Power Syst 2004;19(2):818–25.

[24] Sadeghi M, Sadeghi MA, Nourizadeh S, Ranjbar AM, Azizi S. Power system security assessment using adaboost algorithm. In: Proceedings north american power symposium, 2009.

[25] Kirschen D, Jayaweera D. Comparison of risk-based and deterministic security assessments. IET Gener Transm Distrib 2007;1(4):527–33.

[26] McCalley J, Asgarpoor S, Bertling L, Billinion R, Chao H, Chen J, et al. Probabilistic security assessment for power system operations. In: IEEE PES general meeting. 2004, p. 212–20.

[27] Elkan C. The foundations of cost-sensitive learning. In: International joint conference on artificial intelligence, Vol. 17, 2001, p. 973–8.

[28] Nikolaou N, Edakunni N, Kull M, Flach P, Brown G. Cost-sensitive boosting algorithms: Do we really need them? Mach Learn 2016;104(2):359–84.

[29] Hang F, Huang S, Chen Y, Mei S. Power system transient stability assessment based on dimension reduction and cost-sensitive ensemble learning. In: IEEE conference on energy internet and energy system integration. IEEE; 2017, p. 1–6.

[30] Zhou Y, Zhao W, Guo Q, Sun H, Hao L. Transient stability assessment of power systems using cost-sensitive deep learning approach. In: IEEE conference on energy internet and energy system integration, 2018, p. 1–6.

[31] Cremer JL, Konstantelos I, Strbac G. From optimization-based machine learning to interpretable security rules for operation. IEEE Trans Power Syst 2019;34(5):3826–36.

[32] Xiao F, McCalley JD, Ou Y, Adams J, Myers S. Contingency probability estimation using weather and geographical data for on-line security assessment. In: 2006 International conference on probabilistic methods applied to power systems. IEEE; 2006, p. 1–7.

[33] Fanucchi RZ, Bessani M, Camillo MHM, London JaBA, Maciel CD. Failure rate prediction under adverse weather conditions in an electric distribution system using negative binomial regression. In: International conference on harmonics and quality of power, 2016, p. 478–83.

[34] Drummond C, Holte RC. Cost curves: An improved method for visualizing classifier performance. Mach Learn 2006;65(1):95–130.

[35] Pal B, Chaudhuri B. Robust control in power systems. Springer Science & Business Media; 2006.

[36] Cremer JL, Strbac G. A machine-learning based probabilistic perspective on dynamic security assessment. Int J Electr Power Energy Syst 2021;128:106571.

[37] Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and regression trees. Int Group 1984;432:151–66.

[38] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in python. J Mach Learn Res 2011;12:2825–30.

[39] Platt J. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Adv Large Margin Classifiers 1999;10:61–74.