Training Data Generation Strategies for Data-driven Security Assessment of Low Voltage Smart Grids

Juan J Cuenca, Emanuel Aldea, Eloann Le Guern-Dall'o, Raphaël Féraud, Guy Camilleri, Anne Blavette

Training Data Generation Strategies for Data-driven Security Assessment of Low Voltage Smart Grids

Juan J. Cuenca
IRIT / SATIE Lab / IETR
Université de Toulouse /
ENS de Rennes, Université de Rennes
Toulouse, France
juan.cuenca-silva@irit.fr

Emanuel Aldea
SATIE CNRS UMR 8029
Université Paris Saclay
Gif-sur-Yvette, France
emanuel.aldea@universite-paris-saclay.fr

Eloann Le Guern-Dall'o
CNRS, SATIE Lab / IETR
ENS de Rennes, Université de Rennes
Rennes, France
eloann.le-guern-dallo@ens-rennes.fr

Raphaël Féraud
Orange
Lannion, France
raphael.feraud@orange.com

Guy Camilleri
IRIT-SMAC
Univ. de Toulouse, CNRS, Toulouse INP, UT3
Toulouse, France
guy.camilleri@irit.fr

Anne Blavette
CNRS, SATIE Lab / IETR
ENS de Rennes, Université de Rennes
Rennes, France
anne.blavette@ens-rennes.fr

***Abstract*—Control of small-scale resources in low and medium voltage electricity networks is being decentralised, which increases the need and frequency of use of smart grid security assessment tools. This paper compares three data-driven approaches to classify if a smart grid is "safe" or "unsafe" (i.e., if grid constraints are respected) given an operational point as input: decision trees, gradient tree boosting and deep neural networks. Five novel training data generation strategies are proposed as alternatives to the standard random generation approach, aiming for data-driven models that generalise realistic scenarios better. Simulations are conducted using the IEEE European low voltage test network. Trained models are tested following trends from the literature and using realistic scenarios from the test network documentation, and electric vehicle charging patterns. Our results highlight the inadequacy of the current training data generation strategy, and offer better-performing alternatives. At last, we report on computational times dedicated to training our models, and discuss potential implications for future data-driven smart grid applications.**

***Index Terms*—smart grids, security assessment, data-driven methods, training data, computational time**

I. Introduction

With the evolution of the electricity sector, the control and automation of small-scale energy resources is being decentralised, increasing the importance of grid supervision [1]. Regardless of how and who deploys the active management of such resources, it is paramount for distribution system operators (DSO) to guarantee that no operational state results in congestion events (e.g., under/overvoltage, and/or line and transformer current/power ratings being exceeded), as this could trigger protections or damage equipment in smart grids (SG), with service interruptions as a consequence.

This project was conducted as part of the EDEN4SG project funded by the French National Research Agency (ANR). It also received financial support from the CNRS through the MITI interdisciplinary programs through its exploratory research program. Source code at: https://gitlab.com/satie.sete

The supervision of distribution grids is currently done with a solution inherited from transmission systems, power flow (PF) simulations: numerous computational tools have been proposed to solve a set of equations to verify the physical quantities of the SG. In brief terms, a software takes the topology of the grid (i.e., buses, lines, loads, generators, etc.), and evaluates an operational state to approximate the voltage in each bus and the power flow in each line using an iterative approach (e.g., Newton-Raphson or fixed-point iterative methods [2]). One important concern is balancing this need of DSOs to verify operational states, with the non-negligible computational effort associated.

The evolution in computational power over time has seen the development of data-driven techniques in many fields. Artificial Intelligence (AI) for example has been applied to different power systems problems [3]. Given enough experience (data) it is possible to train a tool to replicate results from measurements or simply replace the PF simulator for a fraction of the computational time [4]. For the purpose of this study we will focus on a classification problem applied to grid supervision: given an operational point (OP) as input (e.g., each customer's demand), a data-driven model must classify it in a binary output: as "safe" or "unsafe"; which can then facilitate decision making by DSOs.

An effective data-driven model must be trained to capture the complex input-output relationships in this multi-input-single-output type of problem. Literature on classification algorithms for grid supervision (also defined as "security assessment") in [3], [5], showcase different models that can reach accuracies of up to 99 % with significant reductions in computational time compared to iterative PF methods. These good performances suggest that data-driven methods are indeed a valid solution to tackle security assessment; however,

two important limitations are noted from these surveys.

1) Training and testing datasets generation: in real life, it is difficult to obtain real measurements for low voltage SGs due to privacy concerns [6], this means that training datasets must be generated synthetically. The current trend is generating operational scenarios randomly within an upper and lower margin from the base case of the SG (e.g., between 50 % and 120 % of the base load of a network), but there are no formal guarantees of this strategy providing enough generality for data-driven models to perform in scenarios outside of the training distribution [7]. To the best of the authors knowledge, no alternative data generation strategy has been applied to SG supervision in the literature.

2) Negligence of the computational effort required to train the models: a major gap found in the surveys referenced earlier is the lack of reporting on training times. Half of the surveyed security assessment studies in [3], [5] fail to report the time required for training. This is important because it has been reported that data-driven models may not be resilient to changes [5]: if every time the topology changes or a new user is connected to the SG we must train a new model or retrain an existing one, training time is an important metric.

Accordingly, this paper will perform a comparison of different data-driven methods, including training effort, performance and operational benchmarks. The objective of this study is to shed light on the challenges which are currently overlooked, and which must be addressed before data-driven approaches are deployed in real life SG supervision applications. In line with this, the contributions of this article are as follows.

- Presenting and comparing data generation strategies alternative to random generation of OPs.
- Benchmarking state of the art data-driven methods for the security assessment classification problem applied to low voltage SGs.

The remainder of this paper is structured as follows. Section II presents the methodology used for this benchmark, including details of the data-driven methods compared, data generation strategies and relevant metrics. Section III presents details of the electricity grid used for study and associated dataset considerations. This is followed by Section IV where the simulation results are presented and discussed. The conclusions and future research opportunities are presented at the end, in Section V.

II. Proposed methodology

For the purpose of this work, three state-of-the-art data-driven methods will be trained using one traditional and five novel data generation strategies. This is to study model performances, as well as the computational times dedicated to training them and how fast they classify compared to existing iterative PF solutions.

A. Data-driven methods

1) Decision trees (DT): require the development of a hierarchical tree of rules. Training is done through the iterative adaptation to partitions of the training dataset [8]. Each node of the DT represents a splitting criterion that defines which branch (i.e., possible occurence) is followed in the decision path. The lowest endpoint nodes are called leafs and are associated to the classification output. DTs were selected for this study for their easiness of interpretation, as it is possible to visualise the hierarchical decision-making process [8]

2) Gradient tree boosting (GTB): also known as gradient boosted regression tree or gradient boosting machine, is a state-of-the-art machine learning technique [9]. GTB's most important reported quality is scalability: running ten times faster than existing hierarchical classification algorithms. In essence the innovation behind GTB is its ability to handle sparse data though a weighted quantile sketch procedure to approximate tree learning [9].

3) Deep neural networks: Artificial neural networks are reportedly able to capture appropriately non-linear relationships in complex problems [5]. A neuron receives an input, performs a non-linear but differentiable function, and gives an output, which is then passed on to some or all neurons in the next network layer as their input. The weights are iteratively tuned in the training of the algorithm using an optimizer [10], [11] based on the backpropagation of the error gradient to adapt to specific problems. A "deep" network has multiple hidden layers of neurons, which can achieve higher accuracy in non-linear problems, provided that weights are tuned with a sufficiently large amount of data [3]. Various heuristics have been proposed [12] to aid the DNN designer with their choice for number of layers, neurons and other hyper-parameters that better adapts to their problem, and this exploration remains mostly experimental.

B. Data generation strategies

The performance of data-driven methods is largely influenced by an appropriate training dataset. In the absence of real-life information, it is necessary to generate synthetic datasets and to label them as "safe" or "unsafe" before training can start. We put forward different alternatives for the problem of security assessment applied to low and medium voltage SGs. Note that datasets generated this way are in no way realistic, but may allow data-driven models to better infer their classification.

a) Random data generation: all articles surveyed in [3], [5] use randomly generated operational scenarios between two margins associated to the base loading of the SG [5], this may not work for SGs in the medium and low voltage. On one hand, medium to low-voltage SGs have fewer users connected at each node, which means consumption is not as aggregated. This demand presents a wider normalised interval for the upper and lower bounds to be considered, increasing the likelihood of outlier injections. On the other hand, the selection of the upper bound greatly limits the proportion of safe and unsafe scenarios (i.e., if the upper bound is too small, all scenarios will be safe; if it is too large, all scenarios will be unsafe), thus creating a highly imbalanced training dataset whose influence on the algorithm performance can be extremely negative. Therefore, we argue that training using

randomly generated data is not able to generalise for real life scenarios for medium to low voltage networks.

b) Guided single random step: we propose to explore iteratively a frontier of high-information content as defined in [7]. This data generation strategy involves generating a first OP and labelling it with the PF simulator. If the OP is safe, one of the nodes is randomly selected to increase its load by a random quantity, making the next OP "less safe". If the OP is unsafe, its load is decreased by a random quantity, making the next OP "safer". By repeating this label-oriented process we aim at "guiding" the dataset towards having a balanced pool of safe and unsafe scenarios.

c) Guided single fixed step: similar to that above, but instead of increasing/decreasing the load by a random step, a fixed step is predefined to discretise the search space.

d) Guided global random step: In this strategy, instead of selecting one random node at each iteration, the power consumption of all nodes is increased/decreased simultaneously by randomly generated quantities.

e) Guided single random step with multiple explorations: This strategy is equivalent to that in paragraph ***(b)*** but after a number of scenarios has been generated, a completely random OP restarts the exploration. This way several high information content frontiers can be explored.

f) Guided single fixed step with multiple explorations: Following the logic from the previous strategy, we also propose to generate a dataset using a fixed step like in paragraph ***(c)***, but with multiple explorations in different frontiers.

C. Data augmentation

To avoid biases towards a dominant category [4], it is proposed to use a state-of-the-art data augmentation strategy called Synthetic Minority Oversampling Technique (SMOTE), which increases the number of samples of the minority classification without adding new information to the model [13]. All generated datasets will be augmented using SMOTE to also test the effects of balancing in model performance.

D. Relevant metrics

After defining SG supervision as a binary classification problem (i.e., under an OP or input, the SG is either "safe" and "unsafe"), it is important to define which of the labels is more informative or important. The SG is a critical infrastructure, and DSOs are interested in knowing which OPs make it "unsafe": metrics will focus on this classification label. Moreover, out of numerous classification metrics registered in [3] this study will focus simultaneously on precision and recall.

$$Precis. = TP/(TP + FP) \tag{1}$$

$$Recall = TP/(TP + FN) \tag{2}$$

Precision is the proportion of correct predictions for "unsafe" (i.e., true positives (TP)) in all predicted scenarios, including false positives (FP). This metric evaluates out of all the scenarios classified by the models as "unsafe", how many are actually not safe (i.e., how precise is the "unsafe" model classification). Recall, also called sensitivity or true positive rate, represents the proportion of correct predictions (i.e., TP) out of all real occurrences (i.e., including the false negatives (FN)). For our problem, this metric evaluates out of all real "unsafe" situations, which proportion is the model able to correctly classify. These metrics are very important: DSOs want a model that is able to classify and not miss "unsafe" scenarios, as these would be the origin of service interruptions.

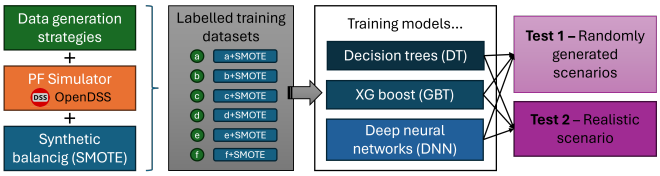

Fig. 1: Structure of the simulations performed.

III. Simulation details

Data-driven methods defined in Subsection II-A were implemented using Python libraries Sklearn.tree, XGBoost and Keras for DT, GTB and DNN respectively, the details of the models (e.g., hyperparameters, optimisers, etc.) can be found in the code repository provided with the article. Simulation were performed using a PC with a 32-core AMD 3970X (3.69 GHz) processor and 128 GB of RAM running Windows 10. Ten independent runs following structure in Fig. 1 were performed for different train dataset sizes, and data generation strategies to obtain several models for comparison in two tests.

A. Case study

The studied SG is the standard IEEE European low voltage test network (ELVTN) [14]. The ELVTN is a radial distribution system representative of urban networks. It is operated at 400 V, 50 Hz and has a total of 55 injection points (i.e., customers). 100 minute-long daily load profiles are part of the documentation for time-series simulations. An electrically equivalent model in the PF simulator OpenDSS [15] was used to label the OPs for the training and test datasets below.

B. Datasets

Several OPs were generated (i.e., each OP has 55 injections, one per customer) and labelled as "safe" if voltage at each node is not outside of a $\pm$ 0.05 p.u. band, and if the line Ampere limits of each line are respected, "unsafe" otherwise. This was done separately for training and testing as follows.

1) For training: each of the six data generation strategies in Subsection II-B and the corresponding six augmented datasets (i.e., using SMOTE) give the total twelve training dataset with 1 million OPs, each used for training. Fig. 2 shows the distribution of injections (i.e., consumption) in ten of the 55 nodes for the selected case study. Generating data randomly within two boundaries (i.e., strategy *(a)* used in the literature) explores a limited area of potential OPs. Note that for this strategy it is not relevant to explore values higher than 6 kW, as OPs generated this way would mostly be labelled "unsafe" (i.e., there would be high consumption at several places simultaneously), which ultimately makes the learning

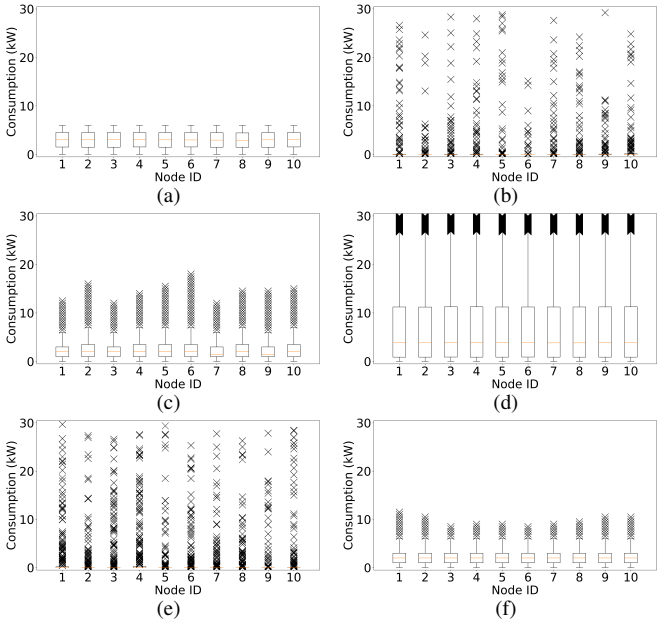

Fig. 2: Box plots for ten nodes showing generated OPs using the corresponding strategies a) to f) in Subsection II-B.

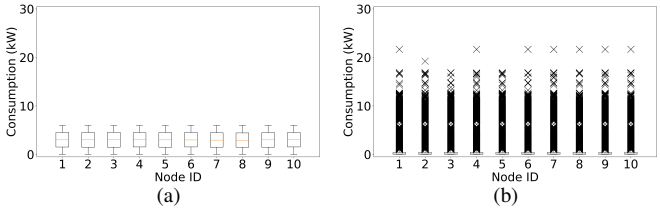

Fig. 3: Box plots for ten nodes with OPs dedicated for testing: (a) test 1 - random generation, and (b) test 2 - realistic scenario.

task far more difficult due to the large imbalance in the training dataset between the two classifications. In contrast, the other data generation strategies appear to cover a larger portion of the search space. To verify the effect of dataset size in training, four subsets of the generated datasets were used for training with random permutations at each run: 10^3 , 10^4, 10^5, and 10^6 OPs.

2) For testing: two datasets are proposed. First, a randomly generated test dataset following current trends from the literature (i.e., training and testing with data generated with the same strategy) was generated with 1 million labelled points for *Test 1*. Last, the daily profiles included in the documentation of the ELVTN were used (i.e., privacy sensitive information that is usually not available for training). This was coupled with electric vehicle (EV) charging patterns as described in [16]. One million OPs were generated randomly superposing permutations of profiles and EV charging patterns in this realistic scenario called *Test 2*. Fig. 3 shows box plots with these two test datasets.

IV. Results

Computational efforts for DT, GTB and DNN models trained using different data generation strategies and training dataset sizes are presented in Fig. 4a. Note that the computational cost of evaluating one OP is on average the lowest for

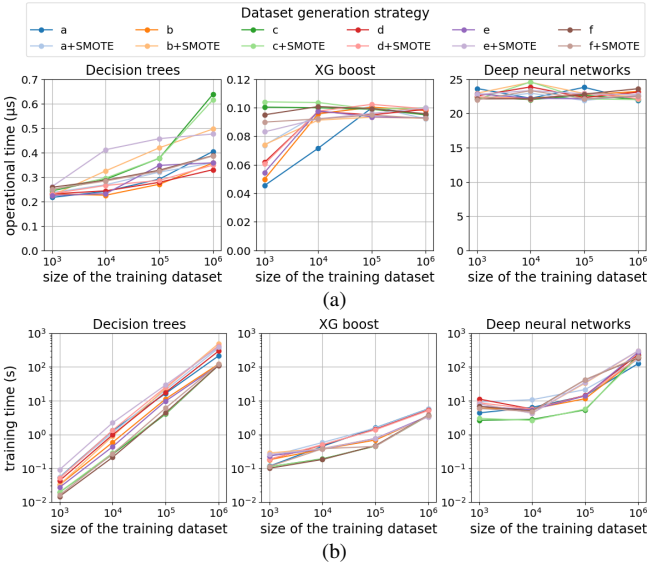

Fig. 4: Average computational efforts of the models: (a) operational time (i.e., how long it takes to evaluate one OP), and (b) training times (not including data generation and augmentation).

GTB, followed by DTs both in the order of a fraction of μs, DNN are slowest with tens of μs. For completeness, we report OpenDSS taking on average 395.2 μs to run a PF, plus 7.08 ms interfacing with Python, and defining if one OP is safe or not. This is translated to computational times gains oscillating in the ranges 2×10^3 to 10^4 times, 10^4 to 6×10^4 times, and 10 to 3×10^2 times for DT, GTB and DNN models respectively.

Results in Fig. 4b show that training effort largely depends on the training dataset size: DT and GTB show linear behaviours, while DNN show an exponential increase in training time with larger datasets. Moreover, note that for a training dataset size of 1 million OPs the training effort is on average up to 10^9, 10^8 and 10^7 times larger than the operational effort for DT, GTB and DNN respectively.

Another concern that is often passed over in the literature is the size of trained models. The largest model trained in this study was a DT that takes up 136.2 MB of disk space. Both DT and GTB create linearly heavier models with larger training datasets (e.g., a DT model trained with 10^3, 10^4, 10^5, and 10^6 OPs, weighted approximately 100 kB, 1 MB, 10 MB, and 100 MB respectively). In contrast the size of DNN models is decided with the hyper-parameter selection (i.e., independent of training dataset size): the size of the model is known *a priori* (e.g., all DNN models trained weighted approximately 104 kB). This information is relevant for some SG applications (e.g., when the security assessment classifier is part of embedded systems).

Model metrics are presented in Fig. 5, a scatter plot with average model precision and recall for the two different tests shows if the model is able to classify correctly. For the purpose of this study, a model is considered to have a good performance if both the precision and recall for "unsafe" label classification are higher than 0.98 (i.e., if they are within the insets of Fig. 5).

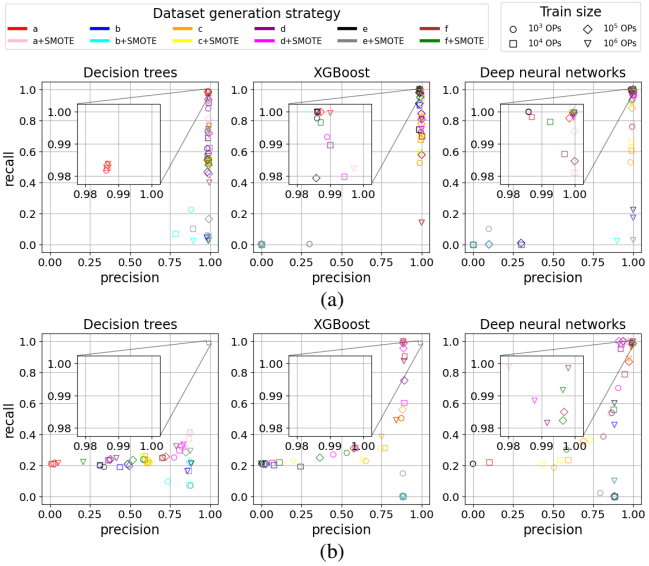

Fig. 5: Scatter plot with average precision and recall for models with different training datasets and sizes for: (a) test 1 - random, and (b) test 2 - realistic.

Test 1 replicates results from the literature. Results in Fig. 5a suggest that it is possible to have a good-performance model of any type with at least one data generation strategy. Training DTs with a dataset similar to that used for testing yields the best performance with all training dataset sizes. For GTB and DNN, multiple data generation strategies and data sizes achieve good performance models, suggesting better overall results. Note that data-driven methods trained with randomly generated data appear to work on randomly generated data. This is put in perspective when the same models are now tested with realistic scenarios. As shown in Fig. 5b, none of the DT or GTB models shows good performance in *Test 2*: models that appeared to work in *Test 1* are now unable to classify correctly "unsafe" OPs. Moreover, models trained with random generation (i.e., data strategy ***(a)***, current trend in the literature), are the worse performing on average. These results highlight the importance of data-generation strategies, alternatives to that of random generation.

While DT and GTB models fail the second test regardless of data generation strategy (i.e., none achieve simultaneous precision and recall above 0.98), DNN models trained using strategies f, f+SMOTE, d and d+SMOTE are promising alternatives even when using a smaller training dataset (e.g., subset with size 10^5 OPs). In addition to this, these models also passed *Test 1*, suggesting good ability to generalise. No formal guarantees are provided at this stage, but the authors expect this exploratory study to trigger further research on data generation strategies.

V. Conclusions

This article benchmarks three data-driven methods for security assessment classification applied to low voltage SGs. Five novel training dataset generation strategies, the use of data augmentation, and different dataset sizes were used to train numerous DT, GTB and DNN models. We provide evidence on the inadequacy of the standard training data generation strategy found in the literature, and we offer alternatives to train models that show better performance in realistic scenarios. Future work on this includes first, the inclusion of other data-driven methods in our benchmark, second, the inclusion of hyper-parameter fine-tuning when measuring training computational efforts, and third the use of other open-source synthetically-generated datasets.

A contrast between heavy efforts in training and computational gains in operation is put forward. This trade-off is often not reported in the literature and may be relevant depending on the application. If training and operation is performed with similar computational resources, our results suggest that designers must consider the life cycle of their data-driven model, and not only its performance: is the training of the algorithm justified by the number of OPs that will be evaluated? Alternatively, we note that if training is performed with significantly more computational resources, the training effort would be close to zero, eliminating this trade-off. Ultimately, this potential trade-off in computational efforts opens the door for the application of deep active learning to increase model resiliency and to reduced training times, which will be explored by the authors in future work.

References

[1] K. Ma and J. Mutale, "Incorporating control system reliability in active management of distribution systems with distributed generation," in *2010 IEEE PES ISGT Europe*, 2010, pp. 1–7.

[2] A. Bernstein, C. Wang *et al.*, "Load flow in multiphase distribution networks: Existence, uniqueness, non-singularity and linear models," *IEEE Trans. on Power Systems*, vol. 33, no. 6, pp. 5832–5843, 2018.

[3] F. De Caro, A. J. Collin *et al.*, "Review of data-driven techniques for on-line static and dynamic security assessment of modern power systems," *IEEE Access*, vol. 11, pp. 130 644–130 673, 2023.

[4] J.-M. H. Arteaga, F. Hancharou *et al.*, "Deep learning for power system security assessment," in *2019 IEEE Milan PowerTech*, 2019, pp. 1–6.

[5] A. Mehrzad, M. Darmiani *et al.*, "A review on data-driven security assessment of power systems: Trends and applications of artificial intelligence," *IEEE Access*, vol. 11, pp. 78 671–78 685, 2023.

[6] D. Lee and D. J. Hess, "Data privacy and residential smart meters: Comparative analysis and harmonization potential," *Utilities Policy*, vol. 70, p. 101188, 2021.

[7] F. Thams, A. Venzke *et al.*, "Efficient database generation for data-driven security assessment of power systems," *IEEE Trans. on Power Systems*, vol. 35, no. 1, pp. 30–41, 2020.

[8] B. de Ville, "Decision trees," *WIREs Computational Statistics*, vol. 5, no. 6, pp. 448–455, 2013.

[9] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD Conference*, 2016, pp. 785–794.

[10] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR 2015*, 2015.

[12] R. Reed and R. J. MarksII, *Neural smithing: supervised learning in feedforward artificial neural networks*. MIT Press, 1999.

[13] N. Chawla, K. Bowyer *et al.*, "Smote: Synthetic minority over-sampling technique," *J. Artif. Intell. Res. (JAIR)*, vol. 16, pp. 321–357, 06 2002.

[14] K. P. Schneider, B. A. Mather *et al.*, "Analytic considerations and design basis for the ieee distribution test feeders," *IEEE Trans. on Power Systems*, vol. 33, no. 3, pp. 3181–3188, 2018.

[15] D. Montenegro and R. Dugan, "Simplified a-diakoptics for accelerating qsts simulations," *Energies*, vol. 15, no. 6, 2022.

[16] S. Zafar, R. Féraud *et al.*, "Multi-armed bandits learning for optimal decentralized control of electric vehicle charging," in *2023 IEEE Belgrade PowerTech*, 2023, pp. 1–6.