Génération de jeux de données pour entraîner un classificateur de sécurité de réseau électrique intelligent de basse tension

Juan Cuenca, Emanuel Aldea, Eloann Le Guern-Dall'o, Raphaël Féraud, Riadh Zorgati, Fabien Petit, Guy Camilleri, Anne Blavette

Génération de Jeux de Données pour Entraîner un Classificateur de Sécurité de Réseau Électrique Intelligent de Basse Tension

Juan CUENCA[1], Emanuel ALDEA[2], Eloann LE GUERN-DALL'O[3], Raphaël FÉRAUD[4], Riadh ZORGATI[5], Fabien PETIT[6], Guy CAMILLERI[7], Anne BLAVETTE[3]

[1]IETR - UMR CNRS 6164, CentraleSupélec Rennes Campus, 35576 Cesson Sevigné, France;

[2]SATIE CNRS UMR 8029, Université Paris-Saclay, 91190 Gif-sur-Yvette, France;

[3]Univ Rennes, ENS Rennes, CNRS, IETR lab – UMR 6164, F-35000 Rennes, France;

[4]Orange Innovation, 22300 Lannion, France;

[5]EDF Lab, 91120 Palaiseau, France;

[6]SRD Energies, 86000 Poitiers, France;

[7]IRIT-SMAC - UMR 5505, Université de Toulouse, CNRS, Toulouse INP, UT3, 31400 Toulouse, France.

La simulation du pilotage de ressources flexibles distribuées dans les réseaux électriques de basse tension nécessite des outils rapides et fiables pour l'étude de la sécurité de ces réseaux - vérifier si les limites (ex : en tension, en courant) du réseau sont respectées. Cette communication décrit les performances de trois types de modèles de classification basés sur les données pour vérifier si l'état opérationnel d'un réseau peut être considéré comme "sûr" ou "non sûr". Le but de ces méthodes est de réduire l'effort computationnel par rapport aux outils classiques d'écoulement de puissance (*power flow*). Cinq stratégies de génération de données sont proposées pour l'entraînement de ces modèles, ensuite testés avec des scénarios réalistes. Nos résultats montrent que les modèles de type réseaux de neurones ont des performances acceptables à coût de calcul réduit. Notre étude souligne l'importance de produire des jeux de données synthétiques qui visent à une meilleure généralisation en inférence, avec des tailles de jeux de données réduites pour l'entraînement.

Mots-clés : réseau intelligent, étude de sécurité, méthodes basées sur les données, jeux de données, effort computationnel.

1. INTRODUCTION

Avec l'évolution du secteur de l'électricité et l'automatisation des ressources énergétiques distribuées, l'importance de la supervision du réseau augmente. Les gestionnaires du réseau de distribution (GRD) cherchent à prévenir les problèmes de congestion, de sous-tension ou dépassement de limites thermiques dans les états opérationnels possibles - ce qui pourrait déclencher les protections ou endommager les équipements connectés au réseau intelligent (RI). Différents outils de gestion décentralisée de l'énergie sont envisagés pour la prise de décision individuelle de nombreuses entités (par exemple, développer des politiques de recharge de voitures électriques avec l'intelligence artificielle [1]). Pour entraîner ces outils, il est nécessaire d'évaluer plusieurs situations opérationnelles potentiellement problématiques - vérifier l'absence de congestion, de sous-tension ou de dépassement de limites thermiques dans des RIs. Cette "étude de sécurité" peut être réalisée à partir de simulations d'écoulement de puissance *power flow* (PF), avec un coût computationnel (de calcul) non-négligeable. Un outil informatique utilise les données physiques du réseau (cet à dire les lignes, charges, générateurs, nœuds, etc.) et évalue l'état opérationnel pour approximer la tension sur chaque nœud et l'écoulement de puissance sur chaque ligne avec un processus itératif (ex : algorithmes de Newton-Raphson ou de point fixe). Cet article vise à comparer différents modèles d'étude de sécurité des RIs alternatifs, basés sur les données, en termes de performance et de coût de calcul pour explorer les enjeux pratiques associés.

Un modèle basé sur les données efficace doit être entraîné avec le but de s'adapter aux relations complexes de cette problématique multi-entrées-sortie-simple. La littérature sur les algorithmes de classification pour la supervision de réseaux (définie comme "security assessment" en anglais) montrent différents modèles qui peuvent achever des performances jusqu'à 99 % avec réductions importantes de temps de calcul comparés aux méthodes itératifs [2, 3]. Ces performances suggèrent que les modèles basés sur les données sont effectivement des solutions adaptées pour la supervision des réseaux dans le contexte d'automatisation de ressources énergétiques.

Cependant, une limitations importante a été notée lors de la révision de la littérature : la génération des données d'entraînement. En pratique il est difficile d'obtenir des données réelles de consommation sur les RIs de basse tension à cause de contraintes de confidentialité [4], cela veut dire que ces données doivent être générées synthétiquement. Les tendances actuelles sur la littérature sont sur la génération aléatoire de points opérationnels (PO) entre une limite supérieure et inférieure de puissance (ex : entre 50 % et 120 % de la charge nominale de chaque nœud du réseau), mais cette stratégie ne donne pas des garanties de généralité ou de performance pour les modèles basés sur les données en dehors de la plage des scenarios d'entraînement [5].

Par conséquent, cet article fera une comparaison des différents méthodes basés sur les donnés. Cette comparaison inclue l'effort d'entraînement, la performance, et l'effort computationnel d'inférence. L'objectif de cette étude est d'explorer les enjeux pratiques actuellement négligés pour le déploiement de ces outils de supervision des RIs. Les contributions de cet article sont :

- Présenter et comparer des stratégies de génération de données d'entraînement alternatives à celle dans la littérature.
- Comparer différents méthodes basés sur les données existantes pour la classification de sécurité appliquée aux RI basse tension.

L'article est structurée comme suit : la section 2 présente la méthodologie pour cette comparaison, les types de modèles, stratégies de génération de données, et métriques d'évaluation. Le cas d'étude est presenté lors de la section 3. Ensuite, la sec-

tion 4 présente les résultats en simulation. La conclusion et les perspectives de recherche se trouvent à la fin dans la section 5.

2. CLASSIFICATION DE SÉCURITÉ

Les méthodes basées sur les données sont proposées comme des alternatives à l'utilisation de simulateurs PF : avec suffisamment de données étiquetées, il est possible d'entraîner un outil qui réplique ou remplace ces outils pour une fraction de son coût de calcul [6]. Pour cette étude nous allons nous concentrer sur un probleme de classification pour la supervision du réseau : étant donné un PO en entrée (ex : la consommation en puissance active et réactive à chaque nœud), un modèle doit le classer comme "sûr" ou "non sûr".

2.1. *Méthodes basées sur les données*

Trois modèles basés sur les données ont été entraînés avec six jeux de données : un jeu de données généré à partir d'une stratégie classique, ainsi que cinq stratégies développées dans ces travaux et présentées ici. Ensuite, une comparaison est effectuée en termes de performance et de coût de calcul d'inférence, par rapport à un outil PF classique.

*1) Arbres Décisionnaires - Decision Trees (DT) :*sa base repose sur le développement d'un arbre hiérarchique de règles. L'entraînement est réalisé à partir de la partition itérative du jeu de données d'entraînement [7]. Chaque nœud de l'arbre représente un critère de division suivi dans le chemin de décision. Les DT ont été choisis pour cette étude en raison de leur facilité d'interprétation.

2) Gradient tree boosting (GTB) : XGBoost construit une foret d'arbres en ajoutant séquentiellement les arbres de manière à optimiser ses performances. La caractéristique clé de GTB est l'utilisation d'un processus de *weighted quantil sketch*, qui permet de diminuer les temps de traitements et de traiter les données éparses [8].

3) Réseaux de Neurones Profonds - Deep Neural Networks (DNN) : ils sont capables d'approximer des relations non linéaires dans des problèmes complexes. Un neurone reçoit une entrée, applique une fonction non-linéaire différentiable, puis génère une sortie qui est ensuite envoyée comme entrée à d'autres neurones dans la couche suivante du réseau. Un réseau profond possède plusieurs couches cachées de neurones, ce qui permet d'obtenir des précisions élevées dans la résolution de problèmes non linéaires [2].

2.2. *Génération de données d'entraînement*

En pratique, il est difficile d'obtenir des mesures réelles de consommation dans des réseaux électriques de basse tension en raison de problèmes de confidentialité [4]. Par conséquent, les données d'entraînement doivent être générées synthétiquement. Un jeu de données doit garantir une certaine généralité, ainsi qu'un nombre suffisant de scenarios "sûrs" et "non sûrs". Néanmoins, il n'existe aucune garantie formelle concernant la généralisation des méthodes basées sur des données en-dehors des scenarios d'entraînement [5]. Cela signifie que les stratégies suivantes doivent être évaluées expérimentalement pour prouver son efficacité dans scenarios opérationnels réalistes.

a) Génération aléatoire : dans la littérature, les PO pour l'entraînement de modèles sont générés aléatoirement entre une limite inférieure et une limite supérieure, les deux associées à la valeur nominale de consommation en puissance active à chaque nœud du RI [2]. Les RI de basse tension ont moins de consommateurs connectés à chaque nœud, ce qui signifie que la consommation en noeud présente usuellement un effet de foisonnement moindre qu'un nœud d'un réseau moyenne tension. Cette demande présente un intervalle normalisé étendu, incompatible avec des limites supérieures et inférieures fixes, ce qui augmente la probabilité d'injections de puissance aberrantes - *outliers*. De plus, la sélection de la limite supérieure définit le déséquilibre du jeu de données d'entraînement (ex : si la limite supérieure est trop petite, tous les scénarios seront étiquetés comme "sûrs", si elle est trop grande, tous les scenarios seront étiquetés "non sûrs").

b) Pas aléatoires individuels guidés : nous proposons l'exploration itérative d'une frontière à haut contenu d'information [5]. Cette stratégie de génération implique l'évaluation d'un PO de départ avec le simulateur PF. Si le PO est "sûr", un nœud est sélectionné pour augmenter sa charge d'une quantité aléatoire - ce qui rendra le prochain PO "moins sûr". Si le PO de départ était "non sûr", la charge serait diminuée au lieu d'augmentée afin de rendre le PO "plus sûr". Grâce à la répétition de ce processus guidé par l'étiquetage, nous visons à obtenir un jeu de données avec suffisamment de PO des deux catégories.

c) Pas fixes individuels guidés : l'augmentation ou la diminution de la charge se fait par pas fixe, au lieu d'être aléatoire. L'objectif est de discrétiser l'espace de recherche.

d) Pas aléatoires globaux guidés : au lieu de sélectionner un nœud aléatoire, la consommation de tous les nœuds est augmentée ou diminuée simultanément par des quantités aléatoires.

e) Pas aléatoires individuels guidés - explorations multiples : cette stratégie est équivalente à ***(b)***, mais après la génération d'une quantité pré-définie de scénarios, un PO totalement aléatoire redémarre l'exploration.

f) Pas fixes individuels guidés - explorations multiples : nous proposons également une stratégie de génération à pas fixes comme ***(c)***, avec les redémarrages qui permettent l'exploration de plusieurs frontières différentes.

2.3. *Data augmentation*

Pour éviter le biais vers une catégorie dominante (une proportion trop importante de scenarios "sûr" ou "non-sûr" dans le jeu de données d'entraînement) [9], nous avons utilisé un outil d'équilibrage forcé pour chaque stratégie : Synthetic Minority Oversampling Technique (SMOTE), pour forcer l'équilibrage de ces jeux de données sans additionner nouvels informations aux modèles. [10]. Chaque jeu de données d'entraînement sera donc augmenté avec l'utilisation de SMOTE pour tester l'effet de l'équilibrage sur la performance de classification.

2.4. *Métriques*

Pour comparer modèles de classification binaire comme dans cette étude, il est nécessaire de définir quel label est plus informatif ou important. Les RIs sont de l'infrastructure critique, et les GRD sont interessés plutot par les PO qui vont faire le réseau "non-sûr", par conséquence les métriques de l'étude vont se concentrer sur ce label. Par ailleurs, les métriques de classification sélectionnées pour cette étude seront en simultanée la précision et le *recall*. Les deux metriques sont très importantes pour le GRD, qui veut un modèle capable d'identifier correctement les situations "non-sûres", potentiellement l'origine des problèmes dans le service.

La précision est la proportion de prédictions correctes pour le label "non-sûr", cet-à-dire la fraction des vrai positifs (true positive TP en anglais), sur tous les prédictions du label ; faux positifis (FP) inclus. Cette métrique évalue combien des scénarios on été correctement labellisés comme "non-sûrs", donc la précision du modèle sur cette classification.

$$Precis. = \frac{TP}{TP + FP} \tag{1}$$

Le *recall* est aussi appelé sensitivité ou *true positive rate* en anglais, il représente la proportion de prédictions correctes (TPs) sur toutes les occurrences réelles ; faux négatifs (FN) inclus. Pour le problème de étudié dans cet article, la métrique évalue combien des scénarios "non-sûrs" ont été correctement identifiés.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

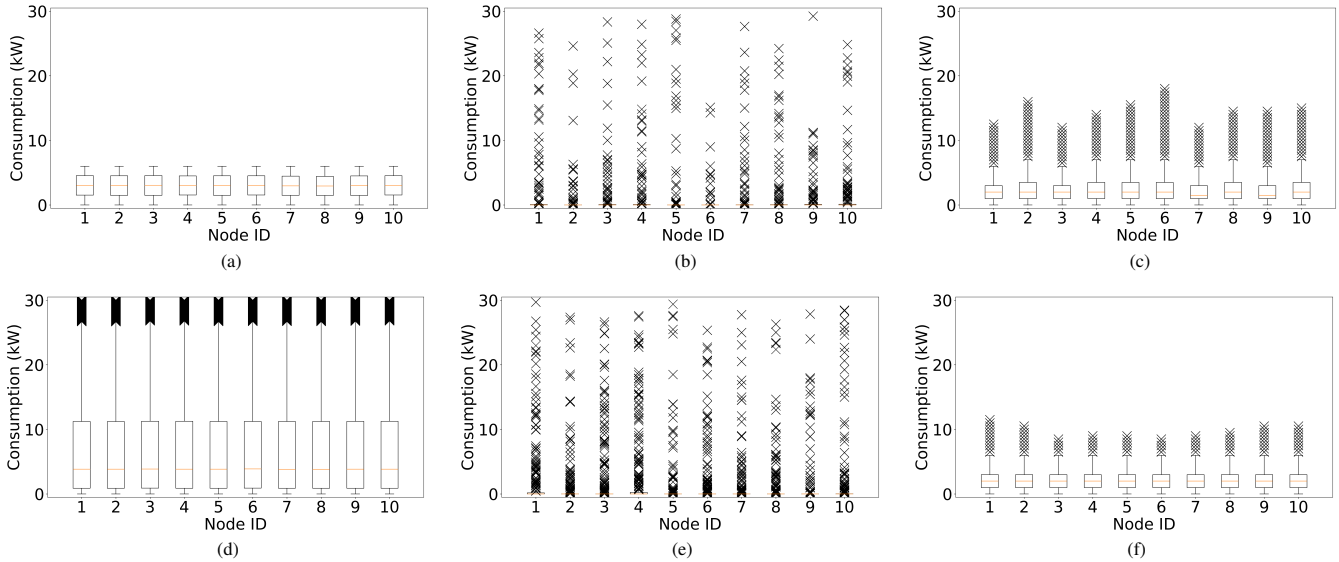


FIG. 1 – Boîte à moustaches pour dix nœuds de l'ELVTN. PO générés avec les stratégies a) à f) dans la section 2.2.

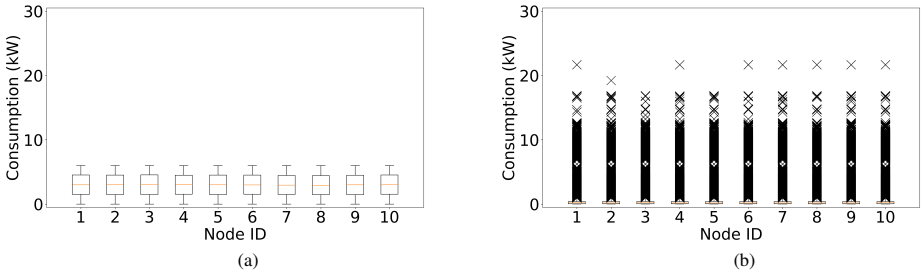


FIG. 2 – Boîte à moustaches pour dix nœuds de l'ELVTN pour les tests : (a) *Test 1* - génération aléatoire, et (b) *Test 2* - scénario réaliste.

3. DÉTAILS DE LA SIMULATION

Les méthodes basés sur les données définis dans la section 2.1 ont été implémentés à l'aide des librairies Sklearn.trr, XGBoost et Keras sur Python pour DT, GTB et DNN respectivement. Les simulations on été faites avec un portable 32-core AMD 3970X (3.69 GHz) processeur et 128 GB de RAM. Dix simulations indépendantes ont été faites avec des différents tailles de jeu de données d'entraînement pour obtenir plusieurs modèles, et les comparer à travers deux tests.

3.1. Cas d'étude

Le RI étudié est le réseau standard IEEE European low voltage test network (ELVTN) [11]. L'ELVTN est un réseau de distribution radial représentatif des réseaux urbains. Il fonctionne avec une tension nominale de 400 V, 50 Hz, et possède 55 points d'injection (consommation). Cent profils journaliers de consommation avec une résolution temporelle d'un minute font partie de la documentation pour les simulation de séries temporelles. Un modèle équivalent du réseau ELVTN dans le simulateur PF OpenDSS [12] a été utilisé comme "oracle" pour labelliser les PO d'entraînement et de test.

3.2. Jeux de données

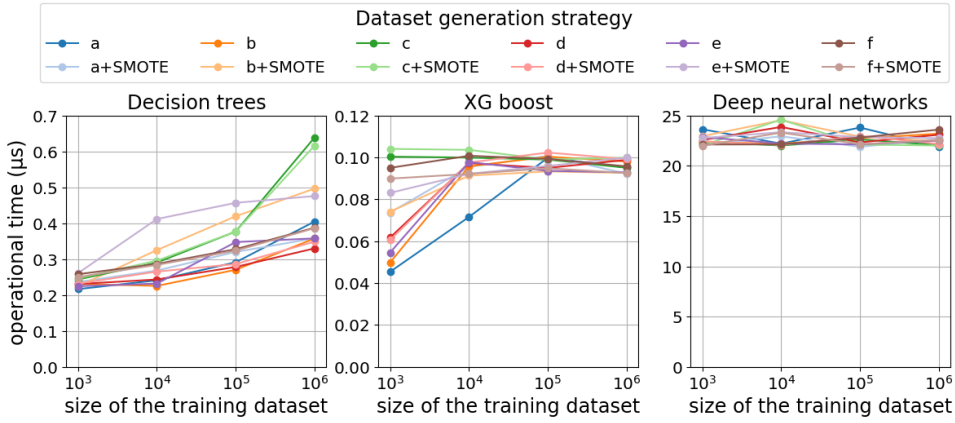
Plusieurs POs ont été générés (chaque PO a 55 injections, une par consommateur) et labellisés comme "sûr" si la tension sur chaque noeud n'est pas au-delà de la plage $\pm$ 0.05 p.u, et si les limites d'intensité de chaque ligne (en Ampères) sont respectés ; "non-sûr" sinon. Ce processus a été fait séparément pour générer les jeux de données d'entraînement et de test comme suit :

3.2.1. Pour l'entraînement

Chaque stratégie de génération dans la section 2.2 et les versions augmentées avec SMOTE donnent un total de douze jeux de données pour entraînement, chaque jeu avec un million de PO. La Fig. 1 montre un échantillon pour 10 sur 55 points d'injection pour le cas d'étude. Générer les données aléatoirement entre deux limites (la stratégie *(a)* commune dans la littérature) explore une zone limitée de POs potentiels. Notez qu'avec cette stratégie il n'est pas intéressant d'explorer valeurs supérieurs à 6 kW car tous les PO générés en dehors de cette limite sont labellisés "non-sûrs" (il y a trop de consommation simultanée dans plusieurs endroits). En contraste, les autres stratégies de génération semblent couvrir une portion plus large de l'espace de possibilités pour les PO. Pour vérifier l'effet de la taille des jeux de données sur la performance du modèle, quatre sous-ensembles de ces jeux de données ont été utilisés avec des permutations aléatoires sur chaque simulation : 10^3 , 10^4, 10^5, et 10^6 POs.

3.2.2. Pour les tests

Deux jeux de données indépendants sont proposés pour tester les modèles. En premier, un jeu de données d'un million de POs a été généré pour suivre la tendance de la littérature sur le *Test 1 :* entraîner et tester avec les mêmes données générées aléatoirement. Ensuite, les profils dans la documentation de l'ELVTN ont été utilisés pour représenter l'information indisponible à cause de contraintes de confidentialité. Pour avoir une possibilité plus importante de situations "non-sûres" les profils de charge des voiture électriques dans [13] ont été superposés pour générer un million de POs de *Test 2 :* scénario réaliste. La Fig. 2 montre les boîtes à moustaches des deux jeux de données de test.



(a)

(b)

FIG. 3 – Efforts computationnels moyens des modèles : (a) temps d'inférence (temps d'évaluation d'un OP), et (b) temps d'entraînement.

4. RÉSULTATS

Le temps de calcul moyen d'inférence (*operational time* en anglais) de chaque type de modèle entraîné, cet-à-dire DT, GTB et DNN avec chaque stratégie proposée et taille de jeu de données est sur la Fig. 3a. Notez que le coût d'inférence pour évaluer un PO est en moyenne le plus bas pour GTB, suivi par les DT autours d'une fraction de μs, DNN sont les plus lents en inférence avec des dizaines de μs. Pour comparer, nous reportons le temps d'inférence de l'oracle (OpenDSS) pour obtenir un label de 395.2 μs à partir d'un PF, en plus de 7.08 ms de communication avec Python. Cela se traduit dans un gain sur le temps d'inférence entre 2×10^3 et 10^4 fois, entre 10^4 et 6×10^4 fois ; et entre 10 et 3×10^2 fois pour les modèles DT, GTB et DNN respectivement. Les résultats sur la Fig. 3b suggèrent que l'effort pour entraîner les modèles est lié à la taille de jeu de données utilisée, les DT et GTB ont des incréments linéaires avec un jeu de données plus large, alors que le DNN augmente exponentiellement. De plus, notez que en utilisant le jeu de données plus large (de 10^6 POs) l'effort d'entraînement est en moyenne jusqu'à 10^9, 10^8 et 10^7 plus large que le temps d'infèrence pour DT, GTB et DNN respectivement.

Les résultats de performance sont présentés sur la Fig. 4, un nuage de points avec la précision et le *recall* montrent si le modèle est en moyenne capable de classifier correctement les POs de chaque test. Pour cette étude, un modèle est considéré comme performant si la précision et *recall* pour le label "non-sûr" est simultanément supérieur à 0.98 (cet-à-dire s'ils sont dans les cadres élargis de la Fig 4).

Le *Test 1* reproduit les résultats obtenus dans la littérature. La Fig. 4a suggère qu'il est possible d'avoir un modèle performant de n'importe quel type avec au moins une stratégie de génération. Notez comment faire l'entraînement de modèles DT avec un jeu de données similaire à celui de test est la seule façon de trouver des performances acceptables. Pour les modèles GTB et DNN, plusieurs stratégies de génération et tailles de jeu de données produisent des bons résultats. Cette tendance n'est conservé lors du test avec des scénarios réalistes. La Fig. 4b montre comment les mêmes modèles DT et GTB ont des très mauvaises performances sur le *Test 2* : un modèle qui semblait marcher lors du *Test 1* est incapable de classifier correctement un PO "non-sûr". Par ailleurs, les modèles entraînés avec la stratégie aléatoire ***(a)*** ont les pires performances en moyenne sur le deuxième test. Ces résultats soulignent l'importance du bon choix de stratégie de génération de données, et mettent en question la stratégie utilisée systématiquement dans la littérature.

Il faut noter que tous les modèles DT et GTB ont des performances insuffisantes sur le deuxième test (aucun modèle de ces types obtient une précision et un *recall* supérieurs à 0.98 en simultanée sur le *Test 2*). En revanche, les modèles DNN entraînés avec les stratégies ***(f)***, ***(f+SMOTE)***, ***(d)*** et ***(d+SMOTE)*** montrent des performances adéquates, même avec des différentes tailles de jeu de données (ex : sous-ensemble avec une taille de 10^5 POs. Notez ultérierement que les modèles performants sous le Test 2 ont aussi des bonnes performances sous le Test 1 ; cela suggère que ces modèles ont une bonne capacité de généralité.

5. CONCLUSIONS

Cet article vise à présenter une analyse comparative de trois types de modèles basés sur les données pour l'étude de sécurité appliquée aux RI de basse tension. Cinq stratégies novatrices de génération de données sont proposées et complétées par SMOTE pour produire des jeux de données de types et tailles variés, qui sont ensuite utilisés pour entraîner de nombreux modèles DT, GTB et DNN. Les perspectives de recherche incluent l'utilisation d'autres types de modèles pour la comparaison, le réglage automatique des hyper-paramètres, et l'utilisation d'outils de génération stochastique de jeux de données.

Un contraste entre les efforts d'entraînement et inférence et mise en avant. Cet compromis est souvent négligé lors des études précédents dans la littérature et pourrait être très important dans certaines applications. Si l'entraînement et l'inférence est faite avec des ressources computationnels similaires, les ré-

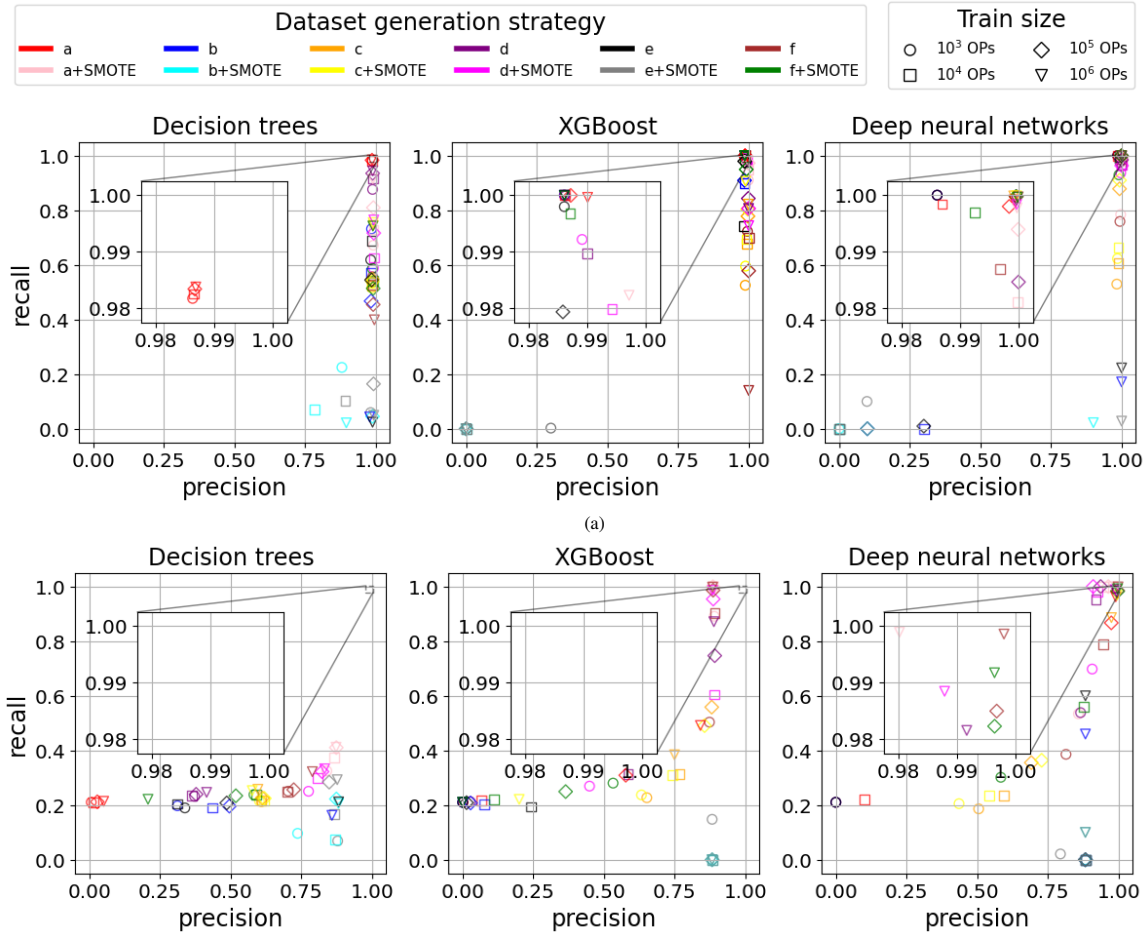


FIG. 4 – Nuages à points avec la précision et le *recall* moyens des modèles entraînés avec différents stratégies de génération et tailles de jeux de données d'entraînement pour : (a) *Test 1* - POs aléatoires, et (b) *Test 2* - scénario réaliste.

sultats de cette étude suggèrent que dans certains cas, il sera pertinent d'étudier le cycle de vie des modèles basés sur les données et non seulement ses performances. L'entraînement de ce modèle est-il justifié par le nombre d'inférences à faire ? Cette dernière question se pose comme perspective de recherche.

6. REMERCIEMENTS

Cette étude est financée par l'Agence National de la Recherche (ANR), projet EDEN4SG (ANR-22-CE05-0023). Ce projet a aussi obtenu le soutien financier du CNRS à travers les programmes interdisciplinaires de la MITI à travers son programme de recherche exploratoire.

7. RÉFÉRENCES

[1] E. L. Guern–Dall'o, R. Féraud, G. Camilleri, P. Maillé, H. B. Ahmed, J. Cuenca, R. Zorgati, F. Petit, and A. Blavette, "Multi-agent contextual combinatorial multi-armed bandits with linear structured super arm : application to energy management optimization in smart grids," in *PGMO-DAYS 2024*, 2024.

[2] F. De Caro, A. J. Collin, G. M. Giannuzzi *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.

[3] A. Mehrzad, M. Darmiani, Y. Mousavi *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.

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

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

[6] J. J. Cuenca, E. Aldea, E. Le Guern-Dall'o, R. Féraud, G. Camilleri, and A. Blavette, "Training data generation strategies for data-driven security assessment of low voltage smart grids," in *2024 IEEE Innovative Smart Grid Technologies EUROPE (ISGT-EU)*, 2024.

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

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

[9] J.-M. H. Arteaga, F. Hancharou, F. Thams, and S. Chatzivasileiadis, "Deep learning for power system security assessment," in *2019 IEEE Milan PowerTech*, 2019, pp. 1–6.

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

[11] K. P. Schneider, B. A. Mather, B. C. Pal *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.

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

[13] S. Zafar, R. Féraud, A. Blavette, G. Camilleri, and H. B. Ahmed, "Multi-armed bandits learning for optimal decentralized control of electric vehicle charging," in *2023 IEEE Belgrade PowerTech*, 2023, pp. 1–6.