Hybrid Learning Aided Inactive Constraints Filtering Algorithm to Enhance AC OPF Solution Time

Fouad Hasan, *Student Member, IEEE*, Amin Kargarian, *Senior Member, IEEE*,
and Javad Mohammadi, *Senior Member, IEEE*

***Abstract*—The optimal power flow (OPF) problem contains many constraints. However, equality constraints and a limited set of inequality constraints encompass sufficient information to determine the problem feasible space. This article presents a hybrid supervised regression-classification learning-based algorithm to predict active and inactive inequality constraints before solving AC OPF solely based on nodal power demand information. The proposed algorithm is structured using a mixture of classifiers and regression learners. Instead of directly mapping OPF results from demand, the proposed algorithm removes inactive constraints to construct a truncated AC OPF. This truncated optimization problem can be solved faster than the original problem with less computational resources. Numerical results on several test systems show the proposed algorithm's effectiveness for predicting active and inactive constraints and constructing a truncated AC OPF.**

***Index Terms*—Active constraint identification, machine learning, optimal power flow (OPF).**

NOMENCLATURE

g	Index for generators.
$i,\ j$	Index for buses.
l	Index for branches.
k	Index for demand samples.
n	Number of buses.
D	Nodal power demand vector.
$F_{\max}$	Maximum branch flow.
G	Actual generation vector.
P_d, Q_d	Real and reactive power demand.
p_{di}^L	Minimum value of load at bus i.
p_{di}^U	Maximum value of load at bus i.
p_g, q_g	Actual real and reactive power generation.
S	Complex power.
V_m	Voltage magnitude.
θ_i	Voltage angle of bus i.
$NI_P,\ NI_Q$	Actual net real and reactive power injection.
$h_v(x)$	Set of voltage constraints.
$h_l(x)$	Set of branch flow constraints.
$A(\cdot)$	Set of true active constraints.
$\tilde{A}(\cdot)$	Set of predicted active constraints by classifiers.
$\tilde{x}$	Predicted x values by learners.
Δ_{di}	Maximum perturbation range for demand at bus i.

I. INTRODUCTION

OPTIMAL power flow (OPF) is one of the main energy management functions that is solved every 5–15 min for power system scheduling and analysis [1], [2]. The size of the OPF problem depends on multiple factors, such as the number of buses and branches. Equality and inequality constraints represent power system characteristics and equipment. These constraints form the OPF feasible space (also known as feasible design space, feasible region, or design space).

Because of the nonconvex and complex nature of AC OPF, solving this problem for large systems is computationally expensive and time-consuming. Various approaches have been proposed in the literature to reduce the computational cost of OPF. Since most OPF inequality constraints are inactive in most cases, one potential approach for relieving computational costs is to identify inactive constraints and omit them from the optimization. There are a few papers for the identification of active and inactive constraints for OPF applications. Most of these papers rely on mathematical and optimization approaches to identify OPF inactive constraints.

As reported in [3], over 85% of branch constraints are inactive in security-constrained unit commitment (SCUC) problems. An analytical condition is developed to identify the set of inactive branch constraints for dc OPF formulation. The concept of umbrella constraints is presented in [4] to describe the feasible set of dc OPF with necessary and sufficient constraints aiming at reducing the size of the problem. This reference presents a mathematical optimization method that finds the umbrella constraints. A method is proposed in [5] to reduce the number of security constraints in SCUC. An optimization-based bound tightening scheme is presented that solves multiple linear

Manuscript received May 30, 2020; revised November 15, 2020 and January 7, 2021; accepted January 15, 2021. Date of publication January 21, 2021; date of current version March 17, 2021. Paper 2020-PSEC-0888.R2, presented at the IEEE Texas Power and Energy Conference, College Station, TX, USA, Feb 6–7, and approved for publication in the IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS by the Power Systems Engineering Committee of the IEEE Industry Applications Society. This work was supported by the National Science Foundation under Grant ECCS-1944752. *(Corresponding author: Amin Kargarian.)*

Fouad Hasan and Amin Kargarian are with the Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA 70803 USA (e-mail: fhasan1@lsu.edu; kargarian@lsu.edu).

Javad Mohammadi is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: jmohammadi@cmu.edu).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIA.2021.3053516

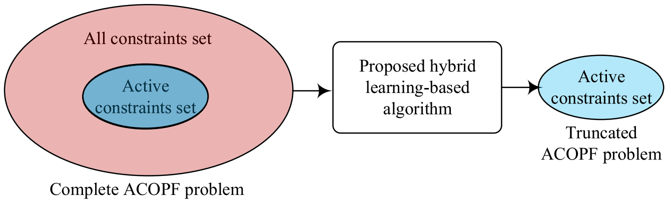


Fig. 1. Proposed active constrains filtering strategy.

programs in parallel to identify redundant linear security constraints. Each linear program contains fewer constraints than the original SCUC. It is observed that roughly 99% of constraints are redundant in real-world scenarios. The proposed algorithm requires network topology information and upper and lower bounds of nodal injection and branch flow limits. The algorithm is independent of unit commitment parameters and uncertain load values. Moreover, Xavier *et al.* [6] propose an iterative contingency search algorithm that can remove most inactive transmission constraints from the SCUC problem. Linear sensitivity factors are used to find violated constraints.

These approaches either solve suboptimization problems or implement iterative search techniques to find active constraints. Some of these approaches, however, might be more computationally expensive than the original optimization problem. Also, they are mainly developed based on convex dc OPF, not nonconvex ac OPF. Solving ac OPF is becoming of more interest in the power system community. Hence, innovative approaches are required for active/inactive constraints identification for the ac OPF problem.

This article presents a combined learning and model-based algorithm to predict inequality constraints' status before solving ac OPF and drop them from the optimization model to speed up the solution time. A hybrid supervised regression-classification-based approach is proposed to identify active and inactive bus voltage and branch flow constraints of ac OPF in the learning phase. The proposed algorithm reads nodal real and reactive power demand as inputs and predicts a subset of inequality constraints with the aim of reducing the size of OPF. One regression learner is trained to project generating units' production by reading demand information. The outputs of these learners are used along with demand information to train two classifiers, one for voltage constraints and another for branch flow constraints. As shown in Fig. 1, the proposed algorithm constructs a truncated ac OPF with a subset of inequality constraints predicted to be active at the optimal point. This makes the proposed algorithm different than several existing methods that directly predict OPF results from demand. The truncated and original ac OPF problems' solutions are almost the same while solving the truncated optimization is much faster and needs less computational resources. The simulation results show the proposed algorithm's effectiveness for identifying active constraints and constructing a truncated ac OPF.

The remainder of the article is organized as follows. Section II provides an overview of machine learning applications in solving various OPF-related problems. The proposed algorithm is presented in Section III. The used learners are explained in Section IV. Section V demonstrates the numerical simulation results. Section VI provides concluding remarks, and future work is discussed in Section VII.

II. Related Work

Machine learning algorithms learn from observation and analysis without any external influence and map a function between input and target data [7]. Machine learning applications to solve power systems problems [8], [9], particularly OPF, has gained a growing interest in recent years [10]. Most published papers focus on the direct projection of OPF solution by machine learning tools using demand information as inputs to learners while ignoring the knowledge of the known mathematical structure of the OPF problem [11]–[15]. This approach works like a black-box that read demand and estimates OPF results. A supervised machine learning-based security-constrained OPF framework is developed in [11] that uses multi-target regression to map the local information and generation dispatches. This framework uses local features as inputs to machine learning models. Navidi *et al.* [12] provide a direct mapping of OPF results using gradient boosting regression. Demand and production cost information is used as inputs to learners that predict the power and voltage of each generator. The nearest neighbor classification is used in [13] to provide an approximate unit commitment solution for market-clearing without the need for computationally expensive unit commitment solvers. Demand and wind generation are inputs to learners, and unit commitment decisions are outputs. In [14], machine learning is applied to predict OPF results to regulate voltage and power flow in distribution grids. In [14] and [15], the proposed method implements a decentralized OPF-based reactive power controller using multiple linear regression learners. This method is implemented on a system with multiple controllable distributed energy resources (DERs). In [16], a support vector machine (SVM) is used to implement the Volt-VAR control scheme. Linear, polynomial, and radial basis function SVM kernels are compared by the lowest sample mean squared error (MSE). The OPF formulation considers the uncertainty coming from renewable energy sources and load. In [17], the authors have extended their work and presented a machine learning-based method to predict optimal settings of a centralized controller based on historical data. While only inverter-based DER reactive power controller is considered in [16], active power curtailment, controllable load shifting, and battery storage are taken into consideration in [17]. King *et al.* [18] have proposed a machine learning-based approach for transient stability constrained OPF based on critical clearing time constraints. Multilayer feedforward neural network (NN) is used to compute the critical clearing time of the formulated OPF problem. Deep learning is used in [19] to predict OPF results. This approach is applicable if information about the previous system states is available to learners.

Such direct estimations, however, do not precisely match with actual solutions. While a trained learner might provide good estimations for many loading conditions, it might not provide accurate enough solutions for many other demand scenarios. An immense training dataset might be required to reach an acceptable level of accuracy for learners. Even if the accuracy of direct OPF solution estimation is high, a small mismatch between projected and actual solutions may yield a suboptimal or infeasible outcome for the nonlinear, nonconvex ac OPF problem. This makes operators reluctant to deploy them for power systems operation. One may use a combined learning and model-based approach to reduce the possibility of suboptimality and infeasibility. The benefits of learning based warm start to solve ac OPF are presented in [20]. Instead of solving OPF directly with machine learning, the demand information is used as inputs to learners to estimate the OPF solution. This solution is used as a starting point to solve the OPF problem. Although having a warm start enhances solution speed, this method does not reduce the size of the OPF problem that significantly impacts the computational complexity of ac OPF.

An idea recently presented in a few papers is to use machine learning to predict inactive constraints rather than using machine learning tools as black boxes to predict OPF results directly [21]–[27]. In [21] and [22], an approach is presented to learn the mapping from uncertainty realization to the optimal solution. This approach avoids directly mapping the input to the optimal solution and instead uses active constraints at optimality as the mapping output. Deka and Misra [23] present another approach to learn the set of active constraints at the optimal point using classification algorithms. A NN classifier is used for learning the active sets. This article deals with dc OPF and uses only classification learners. The authors of [24] have presented a learning-based method to predict umbrella constraints for an OPF problem. The umbrella constraints are necessary and sufficient constraints to cover the OPF feasible solution. Baker and Bernstein [25], [26] present a learning-based chance-constrained approach to remove constraints with zero probability events from the ac OPF formulation for distribution networks. With statistical learning, the proposed framework reduces the computationally demanding joint chance constraints into a series of single chance constraints. Xavier *et al.* [27], which serves as a modified version of the algorithm presented in [6], uses machine learning to predict redundant transmission constraints, warm start, and an affine subspace that contains the optimal solution of SCUC. A combined learning and analytical model-based scheme are presented in [28] to predict congested transmission lines. A learner predicts generation values, and using linear sensitivity factors are used to estimate line flows.

While these approaches are promising, they mainly focus on dc OPF. More sophisticated yet efficient algorithms are needed to detect inactive constraints of the ac OPF problem. These papers use the demand information and train a classifier(s) to identify the status of constraints. A combination of regression and classification learners may enhance the accuracy of the constraint identification process. Motivated by this, we develop a hybrid regression-classification-based algorithm to identify the status of bus voltage and line flow constraints before solving the ac OPF problem.

III. Hybrid Regression-Classification Algorithm for Inactive Constraints Identification

A. Classical AC OPF Formulation

The considered ac OPF problem, presented by (1a)–(1i), is adopted from [29]. The objective function is to minimize generation costs. Nodal power balance constraints are given by (1b) and (1c). Constraints (1d) and (1e) enforce flow limits at, respectively, line sending and receiving terminals. The upper and lower bounds of generating units are imposed by (1f) and (1g). Inequalities (1h) and (1i) are bus voltage magnitude and angle limits

$$\min f(p) \; = \sum_{g} a_g \cdot p_g^2 + b_g \cdot p_g + c_g \tag{1a}$$

$s.t.$

$$g_p\ (\theta,\ V_m, p_g) = P_{\text{bus}}\ (\theta,\ V_m) + P_d - \ p_g = \ 0 \tag{1b}$$

$$g_q\ (\theta,\ V_m, q_g) = Q_{\text{bus}}\ (\theta,\ V_m) + Q_d - \ q_g = \ 0 \tag{1c}$$

$$h_{ls}\ (\theta,\ V_m) = |F_{ls}\,(\theta,\ V_m)|\ - F_{\max} \leq 0 \tag{1d}$$

$$h_{lr}\ (\theta,\ V_m) = |F_{lr}\,(\theta,\ V_m)|\ - F_{\max} \leq 0 \tag{1e}$$

$$p_g^{\min} \leq p_g \leq p_g^{\max} \quad \forall g \tag{1f}$$

$$q_g^{\min} \leq q_g \leq q_g^{\max} \qquad \forall g \tag{1g}$$

$$V_i^{\min} \leq V_i \leq V_i^{\max} \quad \forall i \tag{1h}$$

$$\theta_i^{\text{ref}} \leq \theta_i \leq \theta_i^{\text{ref}} \qquad \forall i. \tag{1i}$$

B. Constraints Status Identification

The status of inequality constraints is not known before solving OPF. All inequality constraints are included in the original OPF problem. The status of constraints will be known after solving the problem. If at the optimal point x^* , an inequity $h(x) \leq 0$ is satisfied as $h\ (x^*) = \ 0$, this constraint is called active or binding, otherwise inactive. To construct a truncated OPF, inactive inequality constraints should be detected and omitted from the optimization problem before solving the problem. Detecting active and inactive constraints can be cast as a binary classification problem. If $h\ (x^*) = \ 0$, it can be labeled as 1, and if $h(x^*) < 0$, it can be labeled as 0.

Without loss of generality, we focus on identifying the status of bus voltage magnitude and branch flow constraints. These two sets of inequalities have high impacts on OPF computation cost. The total number of voltage magnitude and branch flow constraints is higher than that of other OPF inequalities, e.g., generators' upper and lower bounds. The majority of these two sets of constraints are inactive under various loading conditions. This is not a valid argument for generator limits as many of these controllable devices' constraints might be active under several loading conditions.

The goal is to predict constraints status before solving OPF using only nodal demand values. For brevity, we represent branch flow constraints (1d) and (1e) and voltage magnitude constraints (1h) in compact forms as follows:

$$h_l\,(x) := \{h_{ls}\,(\theta,\; V_m)\,;h_{lr}\,(\theta,\; V_m)\} \tag{2}$$

$$h_v\,(x) := \left\{V_i^{\min} - V_i \le 0; V_i - V_i^{\max} \le 0\right\}. \tag{3}$$

Since the bus voltage and branch flow constraints are inherently different, we train two separate classifiers with one for bus voltage constraints and another for branch flow constraints.

Dataset Preparation: Before solving OPF, demand information is available. The following demand vector D is the input for learners

$$P_d = [p_{d1}, p_{d2}, \ldots, p_{dn}]^T \tag{4a}$$

$$Q_d = [q_{d1}, q_{d2}, \ldots, q_{dn}]^{\ T} \tag{4b}$$

$$D\ = \begin{bmatrix} P_d \\ Q_d \end{bmatrix}. \tag{4c}$$

To cover possible loading situations that may occur during system operation in the training phase, we generate a set of demand scenarios as follows:

$$p_{di}^k = \left[p_{di}^L +\ \eta_p\,(k) \cdot \Delta_{di}\right]\ \ \forall k \tag{5a}$$

$$q_{di}^k = \left[q_{di}^L +\ \eta_q\,(k) \cdot \Delta_{di}\right]\ \ \forall k \tag{5b}$$

$$\Delta_{di} = p_{di}^U\ - p_{di}^L \tag{5c}$$

where $\eta_p(\cdot)$ and $\eta_q(\cdot)$ follows a uniform distribution between 0 and 1. The perturbation range Δ_{di} depends on the possible minimum (P_d^L) and maximum (P_d^U) nodal demand values. For each demand scenario, OPF is solved and active and inactive bus voltage ($A(h_v(x))$) and branch flow constraints ($A(h_l(x))$) are identified and stored for training. Demand scenarios resulting in infeasible OPF are omitted.

Two different nonidentical demand datasets are generated. As shown in Fig. 2, demand scenarios in dataset 1 are used to train a regressor ($\mathcal{L}_\Re$). Demand scenarios in dataset 2, along with their predicted generation values obtained from $\mathcal{L}_\Re$, are the training set for classifiers ($\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$). Using one dataset for all learners means that $\mathcal{L}_\Re$ is trained and then utilized with the same dataset, which is not logical.

Proposed Training Structure: The status of voltage and branch flow constraints depends on demand values and generating units' production. Generation values are not known before solving OPF. As shown in Fig. 2(a), regression learner $\mathcal{L}_\Re$ is dedicated to predicting generation values by reading power demand. The input and target vectors of $\mathcal{L}_\Re$ are demand vector D and $[P_g; Q_g]$, $\forall g$. Several buses have neither load nor generation. Having these buses in nodal demand and generation vectors provides no meaningful information for the learner as their corresponding entries are always zero. Only demand buses are used to form input vector D.

By training one regressor for predicting both $\tilde{P}_g$ and $\tilde{Q}_g$, the learner may capture the interaction between real and reactive powers and better understand generator dynamics. Once trained, $\mathcal{L}_\Re$ will predict real ($\tilde{P}_g$) and reactive ($\tilde{Q}_g$) power generated by each unit for each demand scenario

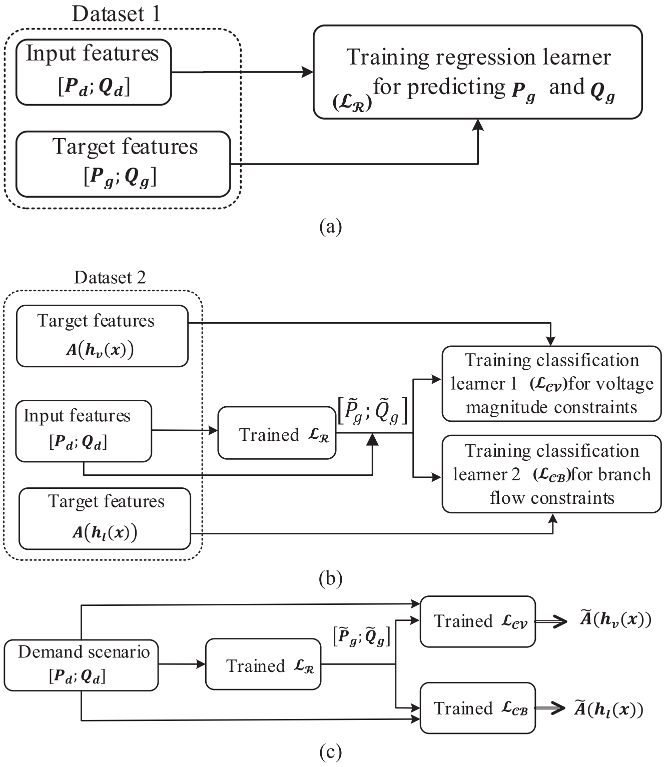

Dataset 1
Input features
$[\boldsymbol{P_d}; \boldsymbol{Q_d}]$
Target features
$[\boldsymbol{P_g}; \boldsymbol{Q_g}]$
Training regression learner
$(\mathcal{L}_\mathcal{R})$ for predicting $\boldsymbol{P_g}$ and $\boldsymbol{Q_g}$
(a)
Dataset 2
Target features
$A(\boldsymbol{h_v}(\boldsymbol{x}))$
Input features
$[\boldsymbol{P_d}; \boldsymbol{Q_d}]$
Target features
$A(\boldsymbol{h_l}(\boldsymbol{x}))$
Trained $\mathcal{L}_\mathcal{R}$
$[\tilde{P}_g; \tilde{Q}_g]$
Training classification learner 1 $(\mathcal{L}_{\mathcal{CV}})$for voltage magnitude constraints
Training classification learner 2 $(\mathcal{L}_{\mathcal{CB}})$for branch flow constraints
(b)
Demand scenario
$[\boldsymbol{P_d}; \boldsymbol{Q_d}]$
Trained $\mathcal{L}_\mathcal{R}$
$[\tilde{P}_g; \tilde{Q}_g]$
Trained $\mathcal{L}_{\mathcal{CV}}$
$\tilde{A}(\boldsymbol{h_v}(\boldsymbol{x}))$
Trained $\mathcal{L}_{\mathcal{CB}}$
$\tilde{A}(\boldsymbol{h_l}(\boldsymbol{x}))$
(c)

Fig. 2. Block diagram of (a) regressor ($\mathcal{L}_\Re$) training procedure, (b) classifiers ($\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$) training procedure, and (c) utilization of trained learners.

$$\tilde{P}_g = [\tilde{p}_{g1}, \tilde{p}_{g2}, \ldots, \tilde{p}_{gn}]^T \tag{6a}$$

$$\tilde{Q}_g = [\tilde{q}_{g1}, \tilde{q}_{g2}, \ldots, \tilde{q}_{gn}]^T \tag{6b}$$

$$\tilde{G} = \begin{bmatrix} \tilde{P}_g \\ \tilde{Q}_g \end{bmatrix}. \tag{6c}$$

Vector D and predicted nodal power generation $\tilde{G}$ are used to form a net nodal power injection vector ($\widetilde{NI}$).

$$\widetilde{NI}_P = [\tilde{p}_{g1} - p_{d1}, \tilde{p}_{g2} - p_{d2}, \ldots, \tilde{p}_{gn} - p_{dn}]^T \tag{7a}$$

$$\widetilde{NI}_Q = [\tilde{q}_{g1} - q_{d1}, \tilde{q}_{g2} - q_{d2}, \ldots, \tilde{q}_{gn} - q_{dn}]^T \tag{7b}$$

$$\widetilde{NI} = \begin{bmatrix} \widetilde{NI}_P \\ \widetilde{NI}_Q \end{bmatrix}. \tag{7c}$$

We train two classifiers for each system, a bus voltage constraint classifier [classifier 1 or $\mathcal{L}_{\mathcal{CV}}$ in Fig. 2(b)] and a branch constraint classifier [classifier 2 or $\mathcal{L}_{\mathcal{CB}}$ in Fig. 2(b)]. The input vector to $\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$ is $\widetilde{NI}$, and their targets are $A(h_v(x))$, and $A(h_l(x))$. As shown in Fig. 2(c), classifiers read the regressor predicted generations to predict constraints' status. The regressor predicts generation values $\tilde{G}$ for each demand scenario in dataset 2. The predicted generation values and demand scenarios in dataset 2 are used to form the net injection vector $\widetilde{NI} = [\begin{smallmatrix} \widetilde{NI}_P \\ \widetilde{NI}_Q \end{smallmatrix}] = [\begin{smallmatrix} \tilde{P}_g - P_d \\ \tilde{Q}_g - Q_d \end{smallmatrix}]$ for training $\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$. By doing so, the classifiers are trained and also utilized by predicted generation values rather than become trained with actual generation and then utilized with predicted generation values. This procedure

Algorithm I: Pseudocode for Training Learners.

```
1. Dataset 1: Generate a set of demand scenarios $P_D(\eta_p)$ and $Q_D(\eta_q)$ by (5), and form $D_1 = \left[\begin{smallmatrix} P_d \\ Q_d \end{smallmatrix}\right]$
2. Solve original OPF for each scenario in $D_1$ and drop infeasible cases
3. Form $G = \left[\begin{smallmatrix} P_g \\ Q_g \end{smallmatrix}\right]$
4. Train $\mathcal{L}_{\Re}$ using $D$ (input) and $G$ (target) in dataset 1
5. Dataset 2: Generate a set of new $P_D(\eta_p)$ and $Q_D(\eta_q)$ by (5) and form $D_2 = \left[\begin{smallmatrix} P_d \\ Q_d \end{smallmatrix}\right]$
6. Solve original OPF for each scenario in $D_2$ and drop infeasible cases
7. Identify $A(h_v(x))$ and $(h_l(x))$ corresponding to each scenario in $D_2$
8. Use $\mathcal{L}_{\Re}$ to predict $\tilde{G} = \left[\begin{smallmatrix} \tilde{P}_g \\ \tilde{Q}_g \end{smallmatrix}\right]$ to each scenario in $D_2$
9. Form $\widetilde{NI} = \left[\begin{smallmatrix} \widetilde{NI}_P \\ \widetilde{NI}_Q \end{smallmatrix}\right] = \left[\begin{smallmatrix} \tilde{P}_g - P_d \\ \tilde{Q}_g - Q_d \end{smallmatrix}\right]$
10. Train $\mathcal{L}_{\mathcal{CV}}$ using $\widetilde{NI}$ as input and $A(h_v(x))$ as target
11. Train $\mathcal{L}_{\mathcal{CB}}$ using $\widetilde{NI}$ as input and $(h_l(x))$ as target
```

would enhance the accuracy of $\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$. The pseudocode to train the learners is represented in Algorithm I.

Utilization Procedure: The utilization procedure of the proposed algorithm is demonstrated in Fig. 2(c). For a given demand, $\tilde{P}_g$ and $\tilde{Q}_g$ are determined by the trained $\mathcal{L}_{\Re}$. The given D and the predicted $\tilde{G}$ will be used to form vector $\widetilde{NI}$ that is the input of the trained $\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$. The output of $\mathcal{L}_{\mathcal{CV}}$ is active bus voltage constraints ($\tilde{A}(h_v(x))$), and $\mathcal{L}_{\mathcal{CB}}$ predicts active branch flow constraints ($\tilde{A}(h_l(x))$). $\tilde{A}(h_v(x))$ and $\tilde{A}(h_l(x))$ will be used to construct a truncated optimization design space and consequently a truncated OPF problem as

$$\min \sum_{g} a_g \cdot p_g^2 + b_g \cdot p_g + c_g \tag{8a}$$

$s.t.$

$$\tilde{A}(h_v\,(x)) \leq 0 \tag{8b}$$

$$\tilde{A}(h_l\,(x)) \leq 0 \tag{8c}$$

$$x \in \chi$$

where χ represents all other constraints except for bus voltage magnitude and branch flow constraints. Suppose all required inequality constraints are predicted correctly. In that case, the truncated OPF problem is equivalent to the original OPF while its size is smaller. The pseudocode to utilize the proposed regression-classification technique to form the truncated OPF is as follows.

One may use $\mathcal{L}_{\mathcal{B}}$ of Fig. 2 to predict $\tilde{P}_g$ and $\tilde{Q}_g$ and then formulate and solve a modified ac power flow instead of a truncated ac OPF. Although solving ac power flow is easier than solving the truncated ac OPF, even a slight error in $\tilde{P}_g$ and $\tilde{Q}_g$ might make ac power flow results suboptimal and, more importantly, endanger power flow feasibility.

Algorithm II: Utilization Algorithm.

```
1. For a given demand vector $D = \left[\begin{smallmatrix} P_d \\ Q_d \end{smallmatrix}\right]$ , run $\mathcal{L}_{\Re}$ to determine $\tilde{G} = \left[\begin{smallmatrix} \tilde{P}_g \\ \tilde{Q}_g \end{smallmatrix}\right]$
2. Form $\widetilde{NI} = \left[\begin{smallmatrix} \widetilde{NI}_P \\ \widetilde{NI}_Q \end{smallmatrix}\right] = \left[\begin{smallmatrix} \tilde{P}_g - P_d \\ \tilde{Q}_g - Q_d \end{smallmatrix}\right]$ using $D$ and $\tilde{G}$
3. Use $\widetilde{NI}$ as input to $\mathcal{L}_{\mathcal{CV}}$ and $\mathcal{L}_{\mathcal{CB}}$ and identify $\tilde{A}(h_v(x))$ and $\tilde{A}(h_l(x))$
4. Formulate truncated AC OPF using $\tilde{A}(h_v(x))$, $\tilde{A}(h_l(x))$, and $\chi$
5. Minimize (8a) subject to (8b), (8c), and $\chi$
```

IV. Selecting Learning Approach and Algorithm

Supervised learning approaches are selected to train $\mathcal{L}_{\mathcal{R}}$, $\mathcal{L}_{\mathcal{CV}}$, and $\mathcal{L}_{\mathcal{CB}}$ in Fig. 2. Various supervised machine learning approaches are available. Among them, NNs have shown promising performance. NNs have outperformed many other machine learning algorithms in recommendation systems, speech and image recognition, natural language processing, etc. SVM with quadratic and Gaussian functions, Gaussian process regression with exponential and quadratic kernels, and ensemble learning with bagging and boosting methods are examined for regression learners. Also, SVM with coarse quadratic and Gaussian functions, the k-nearest neighbor with coarse and weighed techniques, discriminant analysis with linear and quadratic functions, and Naïve Bayes are tested for classification learners. It is observed that while the performance of these approaches is suitable for small power systems, their performance degrades by increasing the size of the system. Also, cases are observed in which these learners failed to map a function between the input and output ac OPF training datasets (these tests and analyses are performed using MATLAB machine learning toolbox).

NNs are used to train $\mathcal{L}_{\mathcal{R}}$, $\mathcal{L}_{\mathcal{CV}}$, and $\mathcal{L}_{\mathcal{CB}}$ for power systems with different sizes, and promising results are obtained for the regressor and classifiers. Hence, we have selected NN for regression and constraints classification. Using activation functions, NN can effectively capture the nonlinearity and complexity of problems, such as ac OPF. A fully connected NN with mini-batch gradient descent is used for $\mathcal{L}_{\mathcal{R}}$, $\mathcal{L}_{\mathcal{CV}}$, and $\mathcal{L}_{\mathcal{CB}}$. For $\mathcal{L}_{\mathcal{R}}$, rectified linear units (ReLU) are used in hidden layers, and linear activation functions are used for the output layer. For $\mathcal{L}_{\mathcal{CV}}$, and $\mathcal{L}_{\mathcal{CB}}$, ReLU is used in hidden layers, and the sigmoid function is used for the output. In the case of linear activation function, the output is proportional to the provided input ($X\,(z) = (mZ)$) whereas, based on the input, the sigmoid activation function provides the output between 0 and 1 ($X\,(z) = \frac{1}{1+e^{-Z}}$). The derivative of the activation function is used in the error backpropagation algorithm, which is a process to optimize each neuron's weights. The loss function of $\mathcal{L}_{\mathcal{R}}$ is the MSE

$$\text{MSE} = \frac{\sum_{k=1}^{K} \left(X^k - \tilde{X}^k\right)^2}{K}. \tag{9}$$

Both real and reactive power are combinedly used to update the weights ($\frac{\sum_{k\,=\,1}^{K}[(P_g^K - \tilde{P}_g^K)^2 + (Q_g^K - \tilde{Q}_g^K)^2]}{K}$). Adam optimizer is

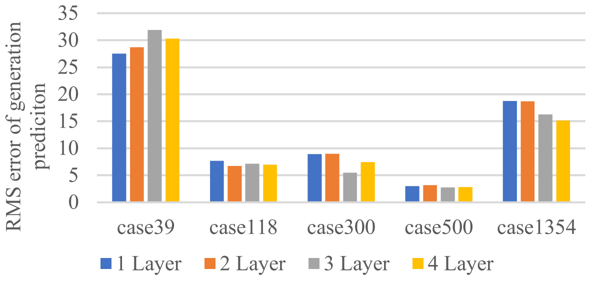


Fig. 3. Root mean square error comparison of different NN regression architectures.

TABLE I
ARCHITECTURE OF TRAINED NNS

Learner	Training parameters	Activation function	Loss function	Optimizer
Regressor $\mathcal{L}_{\mathcal{R}}$	Hidden layer=1 Neuron=256 Epochs=1000 (with early stopping and patience =100), Batch size=100 Validation split=20%	ReLU & Linear	MSE	Adam
Classifier $\mathcal{L}_{\mathcal{CV}}$, $\mathcal{L}_{\mathcal{CB}}$	Hidden layer=1 Neuron=256 Epochs=1000 (with early stopping and patience =100), Batch size=100 Validation split=20%	ReLU & Sigmoid	F2 Loss	Adam

used to find the optimal weight values and train the learners. Various architectures are tested with different numbers of layers, epochs, and batch sizes. Fig. 3 illustrates the results obtained by regressors with different numbers of hidden layers. Merely increasing the number of layers does not improve the prediction accuracy for all systems. One hidden layer is selected for minimalistic learners. Table I depicts the learners' architecture and hyperparameters used in this article. Although we have obtained promising results with these simple architectures, one can use more complex architectures to obtain better results.

Classifier Loss Function: Training an NN is based on solving an optimization problem to find a loss function's best weights. For a typical binary classification problem, the conventional loss function is binary cross-entropy. Although this function works well for many problems, it may not show good performance for imbalanced datasets [30]. Since most voltage and line flow constraints are inactive, the percentage of inactive constraints in dataset 2 is much higher than that of active constraints. Due to this imbalance, the model tends to be overfitted to the class with a higher percentage in the dataset, i.e., inactive class.

To avoid this bias, F_β score is used as the classification loss function. Hyperparameter β controls the importance of precision and recall. Maximizing recall minimizes the number of false negatives (FNs), and maximizing precision reduces false positives (FPs). The objective is to improve recall without hurting precision, which is conflicting. As reliability is critical for power,

TABLE II
NUMBER OF TOTAL CONSTRAINTS AND ACTIVE CONSTRAINTS FOR SEVERAL TEST SYSTEMS

System	Original OPF (Voltage, Branch flow)	Truncated OPF (Active Voltage, Branch flow)	Inactive, Active
case39_epri	78, 92	5, 2	96%, 4 %
case118_ieee	236, 372	12, 2	99%, 1 %
case300_ieee	600,822	32,3	97.5%,2.5%
case500_tamu	1000,1192	16,4	99%,1%
case1354_pegase	2708, 3982	50, 20	99%, 1%

reducing the percentage of FN is desirable. We set $\beta = 2$

$$F_\beta = \left(1+\beta^2\right) \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}} \\ = \frac{\left(1+\beta^2\right) \cdot TP}{\left(1+\beta^2\right) \cdot TP + FP + \beta^2 \cdot FN)} \tag{10}$$

where

$$\text{Precision / Positive predictive value (PPV)} = \frac{TP}{FP + TP} \tag{11a}$$

$$\text{Recall / True positive rate (TPR)} = \frac{TP}{TP + FN}. \tag{11b}$$

Data Scaling: An important preprocessing step before training the learners is data scaling, e.g., normalization and standardization. This step improves the numerical stability of calculations and enhances the prediction accuracy. The data are normalized by (12).

$$X_{\text{normalized}} = \frac{X - X^{\min}}{X^{\max} - X^{\min}}. \tag{12}$$

V. NUMERICAL RESULTS

The proposed algorithm's effectiveness for detecting active and inactive constraints is tested on several small, medium, and large systems. Test systems are adopted from the standard PGLib-OPF benchmark library [31]. MATPOWER interior point solver is used to solve OPF [29]. Python (v3.7.3) based Keras framework (v2.3.1) is used with TensorFlow to trained learners. Simulations are carried out on a personal computer with a 3.70 GHz Intel(R) Xeon(R) CPU, eight cores, and 16 GB of RAM. We have posted our code on GitHub and have uploaded the data used in numerical studies to IEEE DataPort as an open access dataset (DOI: 10.21227/kege-qv50).

A. Average Number of Active and Inactive Constraints

Table II shows the number of voltage and branch constraints for the original OPF and truncated OPF problems. The second column shows the total number of voltage and branch constraints, and the third column depicts the average number of active voltage and branch constraints under various loading conditions. For the 39-bus system, for instance, the total number of voltage and branch constraints are 78 and 92, respectively,

TABLE III
SYSTEM PARAMETERS AND RANGE OF VARIATION OF LOAD

System	NB/NL/NG	Δ_d	No. of scenarios		
			Regressor (dataset1)	Classifier (dataset2)	Testing
case39	39/46/10	70% to 130%	2000	2000	882
case118	118/186/54	70% to 130%	2000	2000	2000
case300	300/411/69	92% to 104%	2000	2000	1641
case500	500/597/90	70% to 109%	2000	2000	3000
case1354	1354/1991/ 260	70% to 110%	1500	1500	1200

* NB/NL/NG stands for the number of Node, Branch, and Generator, respectively.

TABLE IV
INPUT AND OUTPUT LENGTHS OF LEARNERS

System	Regressor ($\mathcal{L}_{\Re}$)		Classifiers ($\mathcal{L}_{\mathcal{CV}}$, $\mathcal{L}_{\mathcal{CB}}$)		
	D	$P_g; Q_g$	NI	h_v	h_l
case39_epri	42	10*2	78	39	46
case118_ieee	189	54*2	236	118	186
case300_ieee	374	69*2	600	300	411
case500_tamu	400	90*2	1000	500	597
case1354_pegase	1332	260*2	2708	1354	1991

out of which, on average, five voltage constraints and two branch flow constraints are active. It is observed that larger systems have a higher percentage of inactive constraints. This shows the potential advantage of detecting active constraints to construct a truncated OPF problem instead of the original OPF. For the 39-bus and 118-bus systems, for instance, the number of constraints of the truncated OPF problem is on average 55% [including all equality and inequality constraints of (1)] less than that of the original OPF.

B. Inactive Constraints Identification by Proposed Hybrid Algorithm

Training: Nodal power demand is varied using uniform random distribution to generate possible demand scenarios over a long operation horizon. Table III shows the load perturbation range (Δ_d) as compared to MATPOWER baseload. The range is obtained by monotonically decreasing and increasing the base-case load until the simulation fails to converge. This range is narrower for the larger systems. OPF is solved for each demand scenario. One regression learner is trained for each system. As shown in Table IV, the length of the output of $\mathcal{L}_{\Re}$ is equal to twice the number of generators, whereas the length of the output of $\mathcal{L}_{\mathcal{CB}}$ ($\mathcal{L}_{\mathcal{CV}}$) is equal to the number of branches (buses). Active voltage and branch flow constraints are labeled as "1," and inactive constraints are labeled as "0" during the preparation of datasets. Two classifiers are trained for each test system. We have used the same architecture for all learners for ease of replicating simulations and to show the proposed algorithm performance with simple machine learning architectures.

Testing: The size of training and test datasets for each studied system is provided in Table III. We use different train-test split ratios. Common ratios are 80%–20%, 70%–30%, and 50%–50%. We have used split ratios with more test scenarios to validate trained learners under various loading conditions. For each test scenario, the original OPF problem is solved to determine the actual active/inactive constraints. The proposed hybrid algorithm is also applied to predict active/inactive constraints. Four primary indices are introduced to interpret predicted results and analyze the accuracy of the proposed algorithm.

1) *True positives* (TP) are cases in which a constraint is predicted to be ACTIVE and its actual status is also ACTIVE.
2) *True negatives* (TN) are cases in which the prediction is INACTIVE and the actual output is INACTIVE.
3) *False positives* (FP) are cases in which the prediction is ACTIVE but the actual output is INACTIVE (type I error).
4) *False negatives* (FN) are cases in which the prediction is INACTIVE but the actual output is ACTIVE (type II error).

In addition, we use the following statistical metrics to analyze the quality of the truncated OPF in detail

$$\text{Accuracy } = \frac{TP+TN}{TP+TN+FP+FN} \tag{13a}$$

$$\text{Misclassification} = \frac{FP+FN}{TP+TN+FP+FN} \tag{13b}$$

$$\text{False negative rate } (\text{FNR}) = 1 - TPR \tag{13c}$$

$$\text{True negative rate } (\text{TNR})\,/\text{specificity } \frac{TN}{TN+FP} \tag{13d}$$

$$\text{False positive rate } (\text{FPR}) \; = \; 1 - \text{TNR} \tag{13e}$$

$$\text{False discovery rate } (\text{FDR}) \; = \; 1 - \text{PPV} \tag{13f}$$

$$\text{Negative predictive value, NPV } = \frac{TN}{FN+TN} \tag{13g}$$

$$\text{False omission rate } (\text{FOR}) \; = \; 1 - \text{NPV}. \tag{13h}$$

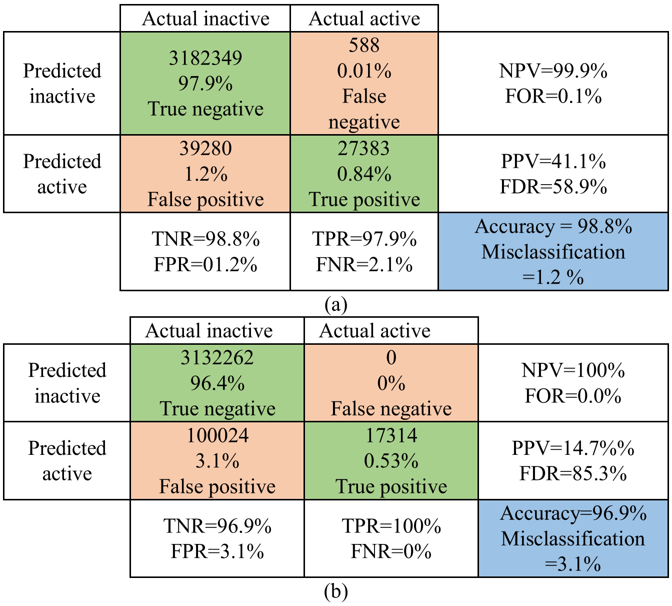
Tables V and VI show these indices for several test systems. We have selected the Pegase 1354-bus test system and constructed a confusion matrix shown in Fig. 4. Each test scenario contains 2708 upper/lower bus voltage magnitude constraints and 3982 sending/receiving branch flow limits. Hence, for 1200 test scenarios, the actual and predicted status of 2708 × 1200 voltage and 3982 × 1200 branch flow constraints are observed to calculate the indices shown in Fig. 4. Green blocks in the second column of Fig. 4(a) and (b) shows that 97.9% of bus voltage constraints and 96.4% of branch constraints are true negatives, which means they are correctly predicted to be inactive. In the third column, green blocks depict that 0.84% and 0.53% of voltage and branch constraints are true positives, which means they are correctly predicted to be active. As shown in orange blocks in the second column, 1.2% of voltage constraints and 3.1% of branch constraints are misclassified to be active. This is the type I error (false positives), meaning these actual inactive constraints are predicted to be active and included in the truncated OPF. This is not critical as these few constraints do not change the truncated feasible space (i.e., do not change the OPF solution)

TABLE V
PREDICTION ACCURACY MEASUREMENTS OF THE PROPOSED ALGORITHM FOR VOLTAGE CONSTRAINTS CLASSIFICATION

systems	FN	FP	TN	TP	NPV	PPV	TPR	TNR	Misclassification	Accuracy
case39_epri	0.01%	4.4%	88%	7.2%	99.8%	61.6%	98.3%	95.1%	4.6%	95.4%
case118_ieee	0.001%	1.0%	9.2%	4.6%	99.8%	80.8%	97.7%	98.8%	1.2%	98.8%
case300_ieee	0.05%	1.7%	92.9%	5.3%	99.9%	75.8%	99.0%	98.2%	1.8%	98.2%
case500_tamu	0.02%	1.1%	97.2%	1.5%	99.9%	57.2%	98.8%	98.8%	1.2%	98.8%
case1354_pegase	0.01%	1.2%	97.9%	0.84%	99.9%	41.1%	97.9%	98.8%	1.2%	98.8%

TABLE VI
PREDICTION ACCURACY MEASUREMENTS OF THE PROPOSED ALGORITHM FOR BRANCH CONSTRAINTS CLASSIFICATION

systems	FN	FP	TN	TP	NPV	PPV	TPR	TNR	Misclassification	Accuracy
case39_epri	0%	0.03%	97.4%	2.5%	99.9%	98.5%	99.9%	99.9%	0.1%	99.9%
case118_ieee	0.02%	0.09%	99.0%	0.84%	99.9%	90.1%	96.8%	99.9%	0.1%	99.9%
case300_ieee	0%	0.04%	99.3%	0.65%	100%	94.3%	100%	99.9%	0.1%	99.9%
case500_tamu	0%	0.58%	98.9%	0.41%	99.9%	41.7%	98.8%	99.4%	0.6%	99.4%
case1354_pegase	0%	3.1%	96.4%	0.53%	100%	14.7%	100%	96.9%	3.1%	96.9%



	Actual inactive	Actual active	

Predicted inactive	3182349
	97.9%
	True negative
		588
		0.01%
		False negative
			NPV=99.9%
			FOR=0.1%

Predicted active	39280
	1.2%
	False positive
		27383
		0.84%
		True positive
			PPV=41.1%
			FDR=58.9%

	TNR=98.8%
	FPR=01.2%
		TPR=97.9%
		FNR=2.1%
			Accuracy = 98.8%
			Misclassification =1.2 %

(a)

	Actual inactive	Actual active	

Predicted inactive	3132262
	96.4%
	True negative
		0
		0%
		False negative
			NPV=100%
			FOR=0.0%

Predicted active	100024
	3.1%
	False positive
		17314
		0.53%
		True positive
			PPV=14.7%%
			FDR=85.3%

	TNR=96.9%
	FPR=3.1%
		TPR=100%
		FNR=0%
			Accuracy=96.9%
			Misclassification =3.1%

(b)

Fig. 4. Confusion matrices for the Pegase 1354-bus system (a) voltage constraints and (b) branch flow constraints.

and have no considerable impact on the computational burden of the truncated OPF. The type II error (FN), meaning actual active constraints are predicted to be inactive, is undesirable. As shown in orange blocks in the third column of confusion matrices, the type II error is close to 0%. TPR for voltage and branch flow constraints is 97.9% and 100%, showing that roughly most actual active constraints are predicted to be active. TNR pertaining to voltage and branch constraints is 98.8% and 96.9%, respectively, showing the percentage of actual inactive constraints predicted to be inactive. NPV for both voltage and branch flow constraints is roughly 100% showing that most predicted inactive constraints are truly inactive. The misclassified constraints are mainly FP that means no important information is lost from the feasible space of the truncated OPF. Therefore, the solution of the constructed truncated OPF will be similar to the solution of the complete OPF formulation.

TABLE VII
COST GAP OF TRUNCATED OPF

System	Cost gap
case39_epri	4e-06%
case118_ieee	3e-07%
case300_ieee	5.9e-05
case500_tamu	7.5e-07 %
case1354_pegase	3e-05%

TABLE VIII
ITERATION NUMBERS AND TIME-SAVING

Systems	Solution time (s) (Total/avg)		Number of iterations (Avg)		Time-saving
	OPF	T-OPF	OPF	T-OPF	
case39	53/0.06	35/0.04	16	14	33%
case118	200/0.10	140/0.07	15	14	30%
case300	583/0.355	361/0.22	35	25	38%
case500	1170/0.39	750/0.25	25	18	35%
case1354	2640/2.20	1800/1.50	42	38	32%

Tables V and VI show that the FN index for all cases is negligible. We have observed a few misclassified voltage constraints. A detailed analysis reveals that these constraints are not heavily binding. Although no FN misclassification is observed for most of the studied cases, it is not guaranteed that the solution of truncated OPF always matches that of the original OPF. In such cases with nonzero FN, the solution of truncated OPF might be infeasible for the original OPF. An iterative constraints generation approach can be used along with the predicted constraints to ensure the solution feasibility.

The average cost gap is used as an index to measure how close the truncated and original OPF solutions are

$$\text{Cost Gap\%} = \frac{\left|f^{T\text{-OPF}} - f^{\text{OPF}}\right|}{f^{\text{OPF}}} \times 100. \tag{14}$$

The smaller the index is, the more accurate the solution of the truncated OPF will be. The values reported in Table VII show that the truncated OPF (T-OPF) solution is very close to that of the original OPF.

Table VIII shows the number of iterations of the interior point method and computation time. OPF is solved for all test scenarios (the number of scenarios is given in Table III). The total runtime, the average runtime, and the average number of iterations per scenario are reported. The time for learners to identify active/inactive constraints is comparatively much lower than OPF runtime, and thus, is neglected. The time-saving values are in comparison with the original OPF. The number of iterations does not decrease significantly. However, since the number of function evaluations per iteration reduces by omitting inactive constraints, the solution time per iteration and the total time decrease. The average time of each iteration can be calculated by dividing the total time by the number of iterations. For instance, for the IEEE 118-bus system, the average time of each iteration of the interior point method decreases from 6.7 ms for the original OPF to 5 ms for the truncated OPF, a 30% time-saving.

In summary, Tables VII and VIII show the promising advantage of the proposed algorithm for reducing the ac OPF problem's computation time while providing a very high accurate solution.

VI. Conclusion

This article presents a hybrid regression-classification algorithm to identify the active and inactive voltage and branch flow constraints for OPF before solving the optimization problem. It is observed that the majority of voltage and branch flow constraints are inactive, even if the system load changes, and have no impact on the OPF solution. The proposed learning algorithm identifies inactive inequality constraints and creates a truncated OPF problem. This algorithm reduces the size of the OPF problem and its computation costs. The simulation studies show that the proposed algorithm can efficiently and quickly separate active and inactive bus voltage and branch flow constraints based on reading the predicted nodal real and reactive power demand. The results show that more than 99% of voltage and branch constraints are predicted correctly, and omitting them results in a significant time-saving for solving ac OPF. Further analysis of the very small fraction (less than 1%) of misclassified constraints shows that they are not heavily binding.

We have tested several learning algorithms, generated diverse samples to ensure that the learners observe various patterns in the training phase, and trained learners with different hyperparameters to obtain high-quality results with a low false-negative percentage. Another reason for the low FN percentage is the small number of active constraints in power systems optimization problems.

VII. Future Work

Advanced approaches, such as generative adversarial networks [32]–[34], can be used to produce more realistic operating scenarios to form a training database. Other constraints such as transformers constraints, phase shifter constraints, load shedding constraints, power electronic converter constraints, capacitor banks, FACTS devices, and battery storage constraints can be included in OPF, and classifiers can be used to identify inactive constraints and drop them from the optimization formulation. In addition, the proposed algorithm can be applied to other power system scheduling problems, such as unit commitment, to reduce their computational burden.

A research direction is to investigate strategies for penalizing false-negative classes in learners' objective functions to reduce the possibility of misclassification of true active constraints. This would be useful for problems with a high percentage of active constraints as compared to total constraints. Another research direction is to develop combined learning techniques and system models to consider grid topology and generation cost changes in active/inactive constraints prediction. This direction is suitable for the application of the proposed algorithm on electricity market problems. In addition, to enhance the solution speed for dc OPF, one may identify the status of all inequality constraints and then solve the first-order optimality conditions based on the system of linear equations instead of solving a truncated dc OPF using optimization techniques. Predicting the sets of active and inactive constraints in the presence of uncertainties, such as renewable sources, is another interesting research path.

References

[1] M. B. Cain, R. P. O'Neill, and A. Castillo, "History of optimal power flow and formulations," Federal Energy Regulatory Commission, pp. 1–36, 2012.

[2] N. Nguyen, S. Almasabi, A. Bera, and J. Mitra, "Optimal power flow incorporating frequency security constraint," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 6508–6516, Nov./Dec. 2019.

[3] Q. Zhai, X. Guan, J. Cheng, and H. Wu, "Fast identification of inactive security constraints in SCUC problems," *IEEE Trans. Power Syst.*, vol. 25, no. 4, pp. 1946–1954, Nov. 2010.

[4] A. J. Ardakani and F. Bouffard, "Identification of umbrella constraints in DC-based security-constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3924–3934, Nov. 2013.

[5] R. Madani, J. Lavaei, and R. Baldick, "Constraint screening for security analysis of power networks," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 1828–1838, May 2016.

[6] A. S. Xavier, F. Qiu, F. Wang, and P. R. Thimmapuram, "Transmission constraint filtering in large-scale security-constrained unit commitment," *IEEE Trans. Power Syst.*, vol. 34, no. 3, pp. 2457–2460, May 2019.

[7] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA, USA: MIT Press, 2009.

[8] S. Wang, P. Dehghanian, L. Li, and B. Wang, "A machine learning approach to detection of geomagnetically induced currents in power grids," *IEEE Trans. Ind. Appl.*, vol. 56, no. 2, pp. 1098–1106, Mar./Apr. 2020.

[9] A. Rosato, M. Panella, R. Araneo, and A. Andreotti, "A neural network based prediction system of distributed generation for the management of microgrids," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 7092–7102, Nov./Dec. 2019.

[10] F. Hasan, A. Kargarian, and A. Mohammadi, "A survey on applications of machine learning for optimal power flow," in *Proc. IEEE Texas Power Energy Conf.*, 2020, pp. 1–6.

[11] Y. Sun *et al.*, "Local feature sufficiency exploration for predicting security-constrained generation dispatch in Multi-area power systems," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl.*, 2018, pp. 1283–1289.

[12] T. Navidi, S. Bhooshan, and A. Garg, "Predicting solutions to the optimal power flow problem," Stanford Univ., Stanford, CA, USA, 2016.

[13] G. Dalal, E. Gilboa, S. Mannor, and L. Wehenkel, "Unit commitment using nearest neighbor as a short-term proxy," in *Proc. Power Syst. Comput. Conf.*, 2018, pp. 1–7.

[14] R. Dobbe, O. Sondermeijer, D. Fridovich-Keil, D. Arnold, D. Callaway, and C. Tomlin, "Toward distributed energy services: Decentralizing optimal power flow with machine learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1296–1306, Mar. 2020.

[15] O. Sondermeijer, R. Dobbe, D. Arnold, C. Tomlin, and T. Keviczky, "Regression-based inverter control for decentralized optimal power flow and voltage regulation," *arXiv:1902.08594*, 2019.

[16] F. Bellizio, S. Karagiannopoulos, P. Aristidou, and G. Hug, "Optimized local control for active distribution grids using machine learning techniques," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2018, pp. 1–5.

[17] S. Karagiannopoulos, P. Aristidou, and G. Hug, “Data-driven local control design for active distribution grids using off-line optimal power flow and machine learning techniques,” *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6461–6471, Nov. 2019.

[18] R. T. A. King, X. Tu, L.-A. Dessaint, and I. Kamwa, “Multi-contingency transient stability-constrained optimal power flow using multilayer feedforward neural networks,” in *Proc. IEEE Can. Conf. Elect. Comput. Eng.*, 2016, pp. 1–6.

[19] F. Fioretto, T. W. Mak, and P. Van Hentenryck, “Predicting AC optimal power flows: Combining deep learning and Lagrangian dual methods,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 1, pp. 630–637.

[20] K. Baker, “Learning warm-start points for AC optimal power flow,” in *Proc. IEEE 29th Int. Workshop Mach. Learn. Signal Process.*, 2019, pp. 1–6.

[21] Y. Ng, S. Misra, L. A. Roald, and S. Backhaus, “Statistical learning for DC optimal power flow,” in *Proc. Power Syst. Comput. Conf.*, 2018, pp. 1–7.

[22] S. Misra, L. Roald, and Y. Ng, “Learning for constrained optimization: Identifying optimal active constraint sets,” *arXiv:1802.09639*, 2018.

[23] D. Deka and S. Misra, “Learning for DC-OPF: Classifying active sets using neural nets,” in *Proc. IEEE Milan PowerTech*, 2019, pp. 1–6.

[24] A. J. Ardakani and F. Bouffard, “Prediction of umbrella constraints,” in *Proc. IEEE Power Syst. Comput. Conf.*, 2018, pp. 1–7.

[25] K. Baker and A. Bernstein, “Joint chance constraints reduction through learning in active distribution networks,” in *Proc. IEEE Glob. Conf. Signal Inf. Process.*, 2018, pp. 922–926.

[26] K. Baker and A. Bernstein, “Joint chance constraints in AC optimal power flow: Improving bounds through learning,” *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6376–6385, Nov. 2019.

[27] Á. S. Xavier, F. Qiu, and S. Ahmed, “Learning to solve large-scale security-constrained unit commitment problems,” *INFORMS J. Comput.*, 2020, doi: 10.1287/ijoc.2020.0976.

[28] F. Hasan and A. Kargarian, “Combined learning and analytical model based early warning algorithm for real-time congestion management,” in *Proc. IEEE Texas Power Energy Conf.*, 2020, pp. 1–6.

[29] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, “MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.

[30] P. Branco, L. Torgo, and R. Ribeiro, “A survey of predictive modeling on imbalanced domains,” *ACM Comput. Surv.*, vol. 49, no. 2, 2016, doi: 10.1145/2907070.

[31] S. Babaeinejadsarookolaee *et al.*, “The power grid library for benchmarking ac optimal power flow algorithms,” *arXiv:1908.02788*, 2019.

[32] Y. Chen, Y. Wang, D. Kirschen, and B.Zhang, “Model-free renewable scenario generation using generative adversarial networks,” *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3265–3275, May 2018.

[33] Y. Chen, X. Wang, and B. Zhang, “An unsupervised deep learning approach for scenario forecasts,” in *Proc. Power Syst. Comput. Conf.*, 2018, pp. 1–7.

[34] I. Goodfellow *et al.*, “Generative adversarial nets,” in *Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

Fouad Hasan (Student Member, IEEE) received the B.Sc. degree in electrical engineering (power system) from the Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, in 2015. He is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA, USA. His research interests include optimization, power systems operation, electricity market, and machine learning.

Amin Kargarian (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Mississippi State University, Starkville, MS, USA, in 2014. He was a Postdoctoral Research Associate with the Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA, USA, in 2014–2015. He is currently an Assistant Professor with the Electrical and Computer Engineering Department, Louisiana State University, Baton Rouge, LA, USA. His research interests include power systems optimization and machine learning.

Javad Mohammadi (Senior Member, IEEE) received the Ph.D. degree from the Electrical and Computer Engineering Department, Carnegie Mellon University (CMU), Pittsburgh, PA, USA, in 2016. As a graduate student, he was a recipient of the Innovation Fellowship from the Swartz Center for Entrepreneurship. He is currently a Special Faculty in electrical and computer engineering with CMU. His research interests include distributed decision making in networked cyber-physical systems, including energy systems, retail operation, and electrified infrastructures. His grid modernization efforts are supported by ARPA-E and the Department of Energy. Thai Beverage Public Company and Carnegie Mellon Thailand have supported his research on retail analytics and intelligence. His research on building energy management has received financial support from the State of Pennsylvania and Target Corporation.