A New False Data Injection Attack Detection Model for Cyberattack Resilient Energy Forecasting

Amirhossein Ahmadi (*Student Member, IEEE*), Mojtaba Nabipour (*Student Member, IEEE*), Saman Taheri, Behnam Mohammadi-Ivatloo (*Senior Member, IEEE*), and Vahid Vahidinasab (*Senior Member, IEEE*)

***Abstract*—As power systems are gradually evolving into more efficient and intelligent cyber-physical energy systems with the large-scale penetration of renewable energies and information technology, they become increasingly reliant upon more accurate and complex forecasting. The accuracy and generalizability of the forecasting rest to a great extent upon the data quality, which is very susceptible to cyberattacks. False data injection (FDI) attacks constitute a class of cyberattacks that could maliciously alter a large portion of supposedly-protected data, which may not be easily detected by existing operational practices, thereby deteriorating the forecasting performance causing catastrophic consequences in the power system. This paper proposes a novel data-driven FDI attack detection mechanism to automatically detect the intrusions and thus enrich the reliability and resiliency of energy forecasting systems. The proposed mechanism is based on cross-validation, least-squares and z-score metric providing accurate detections with low computational cost and high scalability without utilizing neither system's models nor parameters. The effectiveness of the proposed detector is corroborated through six representative tree-based wind power forecasting models. Experiments indicate that corrupted data injected into input, output and input-output data is properly located and removed, whereby the accuracy and generalizability of the final forecasts are recovered.**

***Index Terms*—Cyberattack, False data injection, Forecasting, Machine learning, Cross-validation.**

I. Introduction

Along with population and economic growth, the exploitation of fossil fuel resources continues to increase worldwide, thereby accelerating the gradual fossil fuel depletion and increasing pollution density. These issues, besides the low energy efficiency of conventional power systems, have paved the way for the remarkable proliferation of renewable energy sources (RESs) into power grids [1], [2]. The European Union, as an example, has expressed the ambitious goal of making a 43% of RESs portfolio share by 2030– in the so-called "European Energy Union" [3]. As such, RESs have become a defining feature of modern power systems due to increasing technical efficiency, addressing environmental concerns and lowering costs [4].

Despite all benefits, RESs like wind and solar are tied with volatility, both temporally and geographically [5]. Thus, they are contingent upon high variability and uncertainty, which makes RES penetration challenging. The main issues here are those of extra flexibility and reliability that should be provided once RESs are added to the power systems. To compensate for these issues, several solutions are introduced/adopted, e.g., sub-hourly scheduling and dispatching, flexibility resources integration, and the large-scale movement toward energy storage, demand response programs, and natural gas units [6]. Another way to account for RESs' intermittency is the accurate forecasting of renewable generation. While the associated costs for these different options are very system-dependent and time-evolving, renewable energy forecasting is considered as the lowest-cost and easiest-to-implement thereof [7]. Moreover, accurate forecasting guarantees higher reliability of the system operation and, therefore, penetration of RESs into the power grid [8]. The importance of providing accurate forecasts has led to extensive research conducted in the existing literature.

Forecasting methods can be classified into probabilistic or deterministic methods. Probabilistic forecasts, which are also called interval forecasting, provide estimates of forecast uncertainty that can be very helpful, provided that the system has a way of using additional information. The deterministic forecasting, which is also called point estimate, provides one output for each time interval [9]. Both probabilistic and deterministic models can be further categorized based on the forecast input and output, time-scale, and the associated forecasting method [10]. A comprehensive review of state-of-the-art forecasting methods is presented in [11]. While a large variety of methods such as persistence, physical and statistical methods have been employed previously, machine learning (ML) as a subdivision of statistical methods is extensively utilized (around 38% of the literature according to [12]) due to their capability of learning the uncertainty and high variability associated with RESs.

ML-based algorithms learn patterns from historical data and forecast accordingly [13]. Therefore, the accuracy of forecasting depends directly on the quality of input data. The data quality can be deteriorated either unintentionally using less accurate sensors and less effective measuring techniques or intentionally using data integrity attacks. Unintentionally deteriorated data quality effects on long-term wind power forecasting are explored in [14], using tree-based ML algorithms. Speaking of intentional deterioration of data, as pointed out in [15], the strong tight between data quality and energy forecasting makes ML-based very susceptible to cybersecurity issues. One such cybersecurity issue relates to FDI attacks, through which adversaries inject false data into the historical records and undermine the prediction quality drastically. With the advent and advance of information technologies, the data feeding forecasting systems can be compromised by cyber intrusions through different ways. For example, real-time forecasting data is heavily reliant on

A. Ahmadi was with the Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran (e-mail: e.n.amirhossein@aut.ac.ir).

M. Nabipour is with the Department of Mechanical Engineering, Tarbiat Modares University, Tehran, Iran (e-mail: mojtaba.nabipour@modares.ac.ir).

S. Taheri is with Department of Mechanical Engineering, Purdue University Indianapolis, Indianapolis, US (e-mail: Taheri@purdue.edu).

B. Mohammadi-Ivatloo is with the Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran and also with Information Technologies Application and Research Center, İstanbul Ticaret University, Istanbul, Turkey (e-mail: bmohammadi@tabrizu.ac.ir, bmivatloo@ticaret.edu.tr).

V. Vahidinasab is with the School of Science and Technology, Nottingham Trent University, Nottingham, U.K. (e-mail: vahid.vahidinasab@ntu.ac.uk).

the communication, control, and computational infrastructures of power grids, as well as their hardware, all of which are susceptible to cyber-attacks. Aggregation of data often needs a variety of data sources, which creates a large attack surface. Additionally, the cryptographic methods used to protect the aggregated data may be deciphered if the attackers had higher access to develop them. Moreover, the extended period of data retention necessitates data transfer, which presents additional security issues. Authors in [16] showed that FDI attacks could alter a great proportion of supposedly-protected input data with large magnitudes without even being detected by existing operational practices such as anomaly detection. In early spring 2018, a cyberattack brought down the server of a major load forecasting service provider, affecting the operations of dozens of large electric companies in the United States [17]. Therefore, it is imperative to develop an effective detection method to detect FDI attacks and lay a foundation for the defense mechanism.

The current researches on FDI detection approaches have taken various directions based on several factors, including defense strategies, attack principles, and attack scenarios and goals. Although these directions vary immensely, two core themes can be extracted: model-based and data-driven FDI detection algorithms. The first category focuses on improving the system's state estimation by using the its physical model and parameters and some measurements, which can be further classified into static and dynamic state estimators. Static estimation approaches such as weighted least squares [18] are built upon the supposition that systems' states can be computed with steady-state and deterministic models. Dynamic state estimation approaches, with Kalman filtering as the primary model [19], are developed to deal with the stochastic nature of the system. Both steady-state and dynamic estimation FDI detection algorithms contain numerous other algorithms and applications, as discussed by a comprehensive review in [20]. The problem with these model-based algorithms, however, is their need for the system model, which in turn, dictates extensive computation, low scalability, detection delay, and possible divergence [21].

Data-driven FDI attack detectors are introduced as the second category to provide more accurate detections, less computational cost, and higher scalability without utilizing neither system's models nor parameters. Cao *et al.* [22] explored the invasion pathway of the FDI attacks against power systems and proposed a novel FDI attack detection framework utilizing ensemble learning. Al-Abassi *et al.* [23] presented an ensemble deep learning-based FDI attack detection mechanism for the industrial control systems. Xue and Wu [24] introduced a new active learning-based FDI attack detection method for the cyber-physical systems. Besides, general regression neural networks [25], deep belief networks [26], reinforcement learning [27], long short term memory networks [28], and support vector machines [29] have also been used on the task of FDI attack detection. A detailed review of the relative performance of ML algorithms for FDI applications is accessible in [30]. While anomaly detection methods are extensively explored in previous studies, the cyberattack resiliency issue associated with forecasting systems has yet remained a persistent challenge for the forecasting community.

Luo *et al.* [31] studied data integrity attacks on forecasting systems and asked how to develop robust ML-based forecasting methods under data integrity attacks. Luo and Hong [32] presented three robust load forecasting models under data integrity attacks based on alleviating the effects of large residuals using iteratively re-weighted least squares (l_2-norm) and also replacing l_2 norm with l_1 norm. Numerical tests demonstrated the superiority of their proposed L_1 regression model with regard to multiple linear regression, artificial neural networks and support vector regression models. Yue [33] proposed an optimized anomaly detection system for load forecasting under cyber-attacks based on a combination of heuristically organized time series aggregation and current point anomaly detectors such as second-order difference Chebyshev Inequality-based methods. The concept is expanded upon and evaluated in [34] for forecast data that has been compromised by various cyberattack models. Luo *et al.* [35] present a real-time anomaly detector for very short-term load forecasting built on a Vanilla benchmark model expansion, a dynamic regression model and an adaptive anomaly threshold. In [36], an ML-based anomaly detector is implemented for load forecasting under cyberattacks. Using k-means clustering, the expected load data is first used to recreate the benchmark and scaling data. Based on the cumulative distribution function and statistical features of the scaling results, the Naive Bayes classification is then used to approximate the unique attack template. Finally, dynamic programming is used to quantify the frequency as well as the parameter of a single cyberattack on load forecasting results. Yue *et al.* [17] introduced a descriptive analytics-based anomaly detection approach for a cyber-secure load forecasting that detects long series anomalies effectively. Zheng *et al.* [37] proposed a robust load forecasting method founded on denoising variational autoencoder-based anomaly detection and combining iteratively re-weighted least squares regression, Huber regression and random forest. In [38], an adversarial ML approach is suggested for cyberattack-resistant load forecasting that is capable of detecting a broad spectrum of attacks without detecting outliers. Kozitsin *et al.* [39] introduced a new computationally simple technique based on the auto-regressive integrated moving average model for both anomaly detection and forecasting systems.

However, numerical experiments reveal that none of the presented robust approaches provide accurate forecasts under strong data integrity attacks [40]. Hence, there is a strong need to study the effect of false data attacks on the accuracy of ML-based forecasting models. Previous works mostly explored cybersecurity issues associated with load forecasting while there exist little works investigating generic forecasting systems using ML algorithms. Besides, performance comparison of decision tree ensembles subject to contaminated datasets is neglected. These algorithms offer highly accurate, stable, and interpretable prediction models especially for tabular datasets with a small number of variables. Generalization, which highly depends on the quality of representativeness of data, should also be investigated when contaminated datasets emerge. Detecting malevolently manipulated not only input, but also output and input-output data is also necessary to fully address the issue, which is also overlooked. The main contributions of this paper can be summarized as:

- We propose a novel step-by-step framework that can detect exogenous false data injected into forecasting systems. In contrast to commonly-used anomaly detection techniques like Naïve methods that are unable to detect intelligently injected false data, it is a model-free detection technique based on well-established cross-validation, R^2 values and z-score metric that are effective and easy to employ. It can work with any learning algorithms and can detect attacks injected on input, output and input-output data (Section II).

- We aim to identify the most robust tree-based forecasting model subject to contaminated input datasets. Decision tree ensembles including decision tree, bagging, random forest, AdaBoost, gradient boosting and XGBoost, which are not considered in the cyber-resilient forecasting literature, are trained with the proposed mechanism and compared with robust models such as iteratively re-weighted least squares regression models and an L_1 regression model (Section III).
- We show that the proposed method can recover the generalizability of the forecasting models. This is done by testing those models against data measured at locations different from what the models had been trained with (Section III).

Lastly, Section IV concludes the paper and elaborate on some future research directions.

II. Attack Templates and Proposed FDI Attack Detection Technique

The sophistication and rate of cyberattacks are continually growing, forcing researchers and practitioners to test different systems and evaluate the risk associated with specific situations like data integrity attacks. Simulation is a primary key in assessing data integrity attacks due to its ability to model different scenarios and interactions between attackers and cyber systems. In this section, we start by describing cybersecurity scenarios that can influence forecasting. Then, we introduce an algorithm that is capable of detecting erroneous data generated under those scenarios.

A. Cyberattack Templates

The FDI attacks are a class of cyberattacks aiming to destabilize the system by injecting exogenous false data into the system. Such attacks are usually conducted by attacking servers that accommodate the original data or by directly altering data sensors' reading. Various templates can be incorporated to model data manipulation scenarios during data integrity attacks targeting forecasting systems [32], [36], [38]. (I) Pulse attacks by which ground truth data are manipulated to lower or higher values at a short time interval. (II) Random attacks that add values, generated by random functions, to the actual values. (III) Ramping attacks aim to destabilize the system by gradual manipulation of the ground truth measurements to lower/higher levels via a ramp function. (IV) Scaling attacks through which a proportion of system data is increased or decreased by a significant percentage of the original values, which can be modeled as $\bar{\mathcal{D}}_t = (1+\alpha_s\%)\times\mathcal{D}_t$ for $t_s < t < t_e$, where t_s and t_e are the start and end times of the cyberattack, respectively, α_s represents the scale factor and $\mathcal{D}$ is the original dataset while $\bar{\mathcal{D}}$ is the corrupted dataset. Based on the results reported in [36], scaling attacks are more challenging than pulse, random, and ramping attacks to detect. Hence, the effectiveness of the proposed mechanism is just evaluated on detecting scaling attacks. Fig. 1 indicates an example of the a scaling attack contaminating 20% of the temperature records.

B. Proposed Cyberattack Detection Method

ML algorithms usually have to overcome underfitting or overfitting issues to provide reliability, validity, and generalizability. To estimate the degree over which ML models are over-fitted or under-fitted, one can use the whole dataset for both training and testing. This technique, which is also called "re-substitution validation," usually tends to create overfitting problems. To mitigate the issue, another accepted practice is to split data into two parts. On one part, training is performed while the generalization performance is tested against the other part. This technique is known as "hold-out validation". A major factor here is to choose the ratio over which data is split into the chunks. The choice of data splitting ratio is tricky as the sizes of the chunks can highly affect the generalization ability of the model.

It is also possible to create several training/testing chunks instead of only two. Here, data is segmented equally into k distinct chunks. Out of these k data folds, one is used for testing purposes, and the other $k-1$ parts concern the training process. This training and evaluation are repeated k times; one of the folds is picked each time as the testing set. This method is called "k-fold cross-validation," and is depicted schematically in Fig. 2. The performance metrics obtained from all the k iterations are then averaged to report an ultimate performance estimation. The k-fold cross-validation has established its performance over re-substitution and hold-out validation techniques.

A typical system is usually made up of input variables $\mathbf{x}_i = (x_i^1, x_i^2, ..., x_i^n)^T \in \Re^n$, where n is the number of input variables, and the output variable $y_i \in \Re^m$, where m is the number of output variables. In least-squares-based anomaly detection mechanisms, L_2-norm $\|y-\hat{y}\|_2$ is used to detect whether outliers exist or not, where $\hat{y} = h(\mathbf{x})$ is the estimated value using the state estimation technique and $h(\cdot): \Re^n \longrightarrow \Re^m$ is a vector-valued nonlinear function. Attacker can malevolently manipulate input variables ($\mathbf{x} \longrightarrow \mathbf{x}+\alpha \Longrightarrow \hat{y} \longrightarrow \hat{y}+\mathbf{e}$), or output variables ($y \longrightarrow y+e$) or both of them simultaneously. Based on these detectors, given a threshold τ, outliers exist if the L_2-norm of residuals is larger than τ (i.e., $\|y-\hat{y}\|_2 = \|e\|_2 > \tau$).

The same procedure can be adopted for forecasting system, where $\hat{y} = h(\mathbf{x})$ is the forecasted value and $h(\cdot): \Re^n \longrightarrow \Re$ denotes the forecasting model, where the attacker can inject false data in input variables or output variable. k-fold cross-validation can be adopted as a new mechanism to detect cyberattacks in forecasting systems. As can be seen in Fig. 2, the performance of the k-fold-based forecasting is heavily dependent on the quality and purity of the test set at each iteration, such that a slight deviation in the test set leads to a remarkable loss in the forecasting accuracy. Provided that the whole dataset is clean, the performance metric of each distinct iteration does not deviate heavily from the average accuracy. On the other hand, if one iteration shows a significant performance deviation compared to the average performance obtained by all the other iterations, the corresponding test set associated with that iteration might be corrupted in part or as a whole. As such, the performance estimation associated with the k-fold cross-validation can be used to recognize parts of data that are not behaving normally. Indeed,

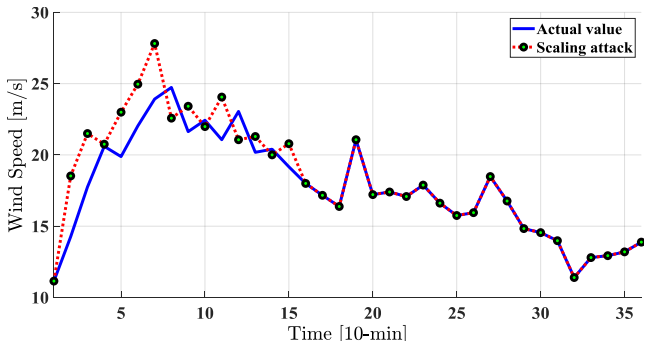

Fig. 1. An example of scaling attack on records

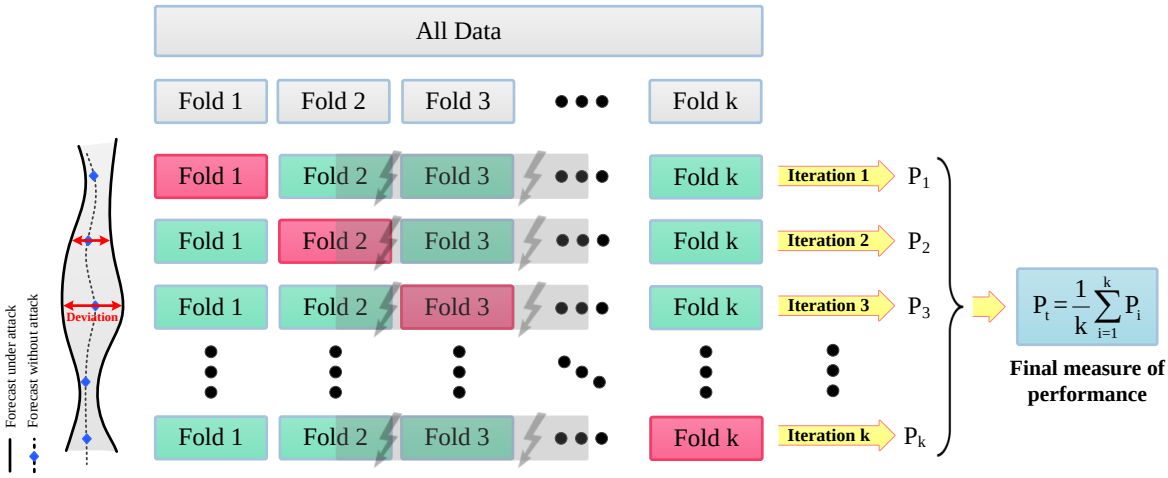

Fig. 2. Schematic of the k-fold cross-validation method.

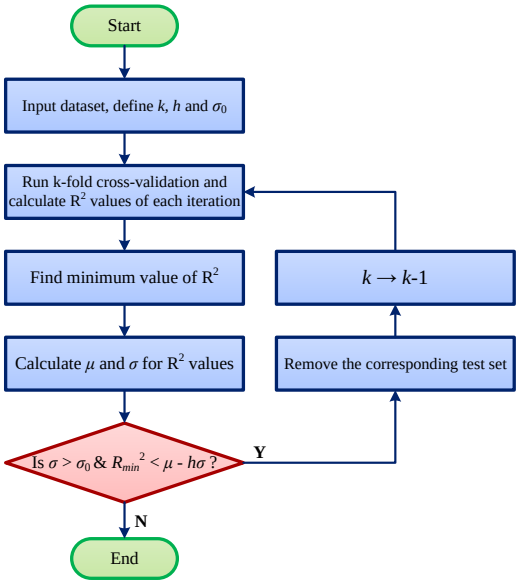

Fig. 3. Flowchart of the proposed cyberattack detection method.

k-fold cross-validation can be used to detect whether outliers exist or not and if exist to locate where it is by utilizing the coefficient of determination (R^2) value, as follows, instead of L_2-norm $\|y-\hat{y}\|_2$.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{N}(y_i-\bar{y})^2} = 1 - \frac{\|y-\hat{y}\|_2^2}{\|\bar{y}-\hat{y}\|_2^2} \quad (1)$$

On the other hand, the z-score is a well-established anomaly detection metric that measures how many standard deviations (σ) a data point (x) is from the data mean (μ), which can be formulated as $z=(x-\mu)/\sigma$. When computing the z-score for each sample on the data set a threshold must be specified, which as a rule of thumb, can be 2.5, 3, 3.5 or more standard deviations. On these bases, we propose an algorithm that can assess data integrity of forecasting systems and enhance their overall performance by detecting inputs that are deteriorated.

The proposed algorithm is demonstrated with the flowchart shown in Fig. 3. The main idea behind the projected k-fold-based cyberattack detection method is first to evaluate the performance of all k-fold models generated at each iteration and then compare their performance one by one with the average performance of all the other iterations. If the z-score of R^2_{min}, i.e. $(\mu - R^2_{min})/\sigma$, is larger than a given threshold (h), the corresponding test set will be detected as cyberattack. The procedure will continue up until all test sets violating the threshold be removed from the dataset. Here, the tuning of k plays a pivotal role in reaching a compromise between accuracy and cost. Higher values of k give more detecting resolution, yet, at the expense of more computational cost, and vice versa.

III. Representative Forecasting Models and Simulation Results

Generally, ML models can be grouped into supervised, unsupervised, semi-supervised, and reinforcement learning in line with the type and volume of supervision they receive during the training process. Supervised learning, in which the desired labels are fed to the algorithm, is the most common category, which can be classified into classification and regression. Several techniques have been developed to deal with supervised learning tasks, each of which has its implications and applications. While the proposed approach can work with the other machine learning models like neural networks, tree-based ensembles are selected here as one of the most popular and successful supervised learning approaches. One major advantage of tree-based models is their capability of learning complex and nonlinear relationships, which makes them adaptable to various kinds of problems in the ML area. Also, they incorporate predictive algorithms with high accuracy, swift performance, and straightforward interpretation. Here, decision tree, random forest, boosting, gradient boosting, extreme gradient boosting, and bagging are adopted as representative tree-based models.

A. Decision tree

Decision tree learning is a simple but powerful method for classifying targets or forecasting values. One advantage of decision tree models is their ability to learn abstract decision rules extracted from data features, without being relied on intense data preparation. Learned-trees embody a single root node that branches into several new nodes. That procedure is then repeated for descending nodes,

which generates a subtree rooted at every new node. A tree-shaped model is finally created that is robust to noisy data and also capable of mapping expressive functions. Having a tree-shaped configuration improves human interpretation of learned-trees as they can be re-represented by groups of if-then-else rules.

B. Random forest

Random forest is another versatile ML method made by a large number of decision trees. Random forest tends to reduce the associated error with decision trees by aggregating the forecasts from a cluster of predictors. Three major steps are then conducted: training instances are randomly selected when making trees, some subsets of features are nominated to split nodes, and carefully-chosen subsets of all features are employed for splitting nodes of decision trees. Each tree learns from a randomly selected sample of the input instances during the training process.

C. Boosting

Boosting is another ensemble method that combines a series of predictors to make an ultimate powerful learner. Each model is trained according to its predecessor and with the aim of improving the overall performance of an ensemble. To boost the overall performance, a new tree added to the ensemble should account for its predecessor's error. This is done by tweaking the weight of instances that are strongly underfitted by the former predictor. Once all models are sequentially added to the ensemble, a model is made that usually outperforms each of the single weak predictors in terms of generalization ability. AdaBoost is a widely-utilized method among boosting algorithms.

D. Gradient boosting

Gradient boosting (GBoosting) method is another boosting method that creates a strong predictor, normally a decision tree, out of conceivably several weak prediction models. Contrary to other boosting algorithms like AdaBoost, Gradient Boosting aims at fitting the new estimator to the residual errors rather than increasing the relative weights made by the previous learners.

E. XGBoost

XGBoost is developed and designed as an optimized version of the gradient boosting method. While XGBoost applies the same principle of combining weak learners, parallel and distributed computing and efficient memory allocation improve the performance, accuracy, and scalability of the method. Efficient imputation of missing values, built-in cross-validation, advanced pruning capability, and catch awareness are other strengths of the XGBoost method.

F. Bagging

Bagging is also an ensemble learning method; yet, with a different approach to combine a diverse set of estimators. Instead of using different predictors, as in random forest and boosting methods, bagging uses a communal algorithm for each predictor but train them on a random sampling of a small subset of the main dataset. The subsets are replaceable, training instances can be re-sampled multiple times for each estimator, and predictions are performed after a majority voting mechanism. The core strength of a bagging estimator is its ability to obtain the best trade-off between the bias and variance of the dataset.

G. Performance metrics

To assess the effectiveness of the proposed models, we use various performance indices with respect to accuracy. Following paragraphs introduce those performance indices.

1) Mean absolute error (MAE)

MAE, which evaluates the mean absolute difference between predictions and observations, is expressed in (2) as

$$\mathrm{MAE}=\frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i-y_i|. \tag{2}$$

It is worthy of mentioning that because MAE has not a differentiable function, most ML algorithms that use gradient descents have a hard time incorporating MAE as the evaluation metric. To compensate for this problem, other performance metrics should be considered.

2) Root mean square error (RMSE)

RMSE, as expressed in (3), can consider the error's direction by measuring the root of the mean of the distance between predictions and observations.

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i-y_i)^2} \tag{3}$$

To make the RMSE metric more sensible when it is used in RESs models, normalized RMSE (nRMSE) is often proposed, whose formula is depicted in (4).

$$\mathrm{nRMSE}=\frac{1}{P_{inst}}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i-y_i)^2} \tag{4}$$

where P_{inst} is the installed capacity of the wind or solar power plant.

H. Experimental results

Tree-based algorithms are tested against contaminated data used for forecasting wind power generation of 1 MW wind turbines installed in the northeast region of Iran. Model hyper-parameters are the maximum depth of each tree, estimator, learning rate and the number of trees to grow, which are set to 10, decision tree, 0.1 and 500, respectively. The aim here is to find the most resilient model and also reveal the effect of the proposed mechanism on the associated prediction accuracies. Table I shows the results of a comparative study between decision tree ensembles [14] and iteratively re-weighted least squares with l_1 and l_2 norms [32] applied on the dataset described in Table II. Datasets including 18 months of wind speeds as inputs and wind powers as the output were collected with a 10-min sampling time measured at the height of 40 m. As can be seen, all models showed acceptable performance with small errors and high coefficients of determination. Nevertheless, XGBoost and L_1 regressor are the most and the least accurate models, respectively.

Thereafter, the effect of the scaling attack with a normal distribution, a mean of 20, and a standard deviation of 100 is explored on the forecasting accuracy of models. Table I summarizes the performance metrics of the representative forecasting models when 20% of the whole data is contaminated. Results indicate a remarkable degradation in the performance metrics, especially for L_1 and L_2 regressors proposed in [32], which means all models own weak robustness against FDI attacks. It is worth mentioning that in the normal condition, both of hold-out and cross-validation approaches produce similar forecasting performance. However, as the salient point of the results, there is a marked difference between results obtained using the hold-out method and those obtained through the cross-validation method in the presence of cyberattacks, where the latter led to an aggravated situation. This is because of the fact that *k*-fold cross-validation ensures that every fold has the

chance of appearing in training and test set, where contaminated folds produce inaccurate performance in the test position, thereby distinctly reducing the average performance of the model.

Herein, three different case studies are designed to analyze the effectiveness of the proposed detector, including scaling attack on input data, output data and both of them. Starting with first case study, three scenarios are considered to explore the impact of attack time of occurrence on the detection process as schematically depicted in Fig. 4. The first and ideal scenario represents when all the corrupted data fall entirely into $n < k$ consecutive folds (here two folds). The second case denotes when the attacker manipulates half of a fold, full of the next fold and half of the other consecutive fold, and the third one exemplifies when 10% of a fold, full of the next fold and 90% of the other consecutive fold is corrupted under the cyberattack. Fig. 5 compares the wind power forecasting under normal condition and scaling attack. As can clearly seen, scaling attack on data caused remarkable deviations from the normal operation for XGBoost forecasts, in which contaminating the input data is more effective than corrupting the output data. This makes sense for wind power forecasting due to the exponential correlation between wind speed and wind power. Therefore, it is imperative to develop an effective FDI attack detection mechanism to automatically detect the intrusions and thus enrich the reliability of forecasting.

Table III lists the results for XGBoost algorithm equipped with the proposed detector for all considered scenarios. As can be seen, when a contaminated fold is chosen as test set, the R^2 value is quite distinct from the others, which increases the standard deviation of the values. Taking it into account and by using the threshold condition, the proposed approach readily detects and gets rid of all contaminated folds. As can be seen from Table III, the algorithm

TABLE I
FORECASTING ACCURACY OF REPRESENTATIVE TREE-BASED MODELS

Algorithm	Attack	Method	MAE	RMSE	R^2
Decision tree	×	-	7.81	27.41	0.9995
	✓	Hold-out	245.39	810.25	0.7771
	✓	CV	302.98	916.87	0.5465
Bagging	×	-	7.08	23.57	0.9996
	✓	Hold-out	238.93	695.37	0.8358
	✓	CV	280.29	781.55	0.6649
Random forest	×	-	7.09	23.57	0.9996
	✓	Hold-out	233.58	665.66	0.8495
	✓	CV	277.55	764.99	0.6830
AdaBoost	×	-	7.67	24.96	0.9996
	✓	Hold-out	347.78	711.68	0.8280
	✓	CV	444.08	856.15	0.6118
Gboosting	×	-	6.02	23.82	0.9996
	✓	Hold-out	234.78	635.87	0.8627
	✓	CV	275.78	723.73	0.7151
XGBoost	×	-	**5.52**	**20.44**	**0.9997**
	✓	Hold-out	**231.45**	**633.23**	**0.8712**
	✓	CV	**270.18**	**719.95**	**0.7310**
L_1 **regressor**	×	-	443.28	509.63	0.9156
	✓	Hold-out	572.17	854.33	0.6863
	✓	CV	662.13	938.20	0.4694
L_2 **regressor**	×	-	408.72	479.72	0.9256
	✓	Hold-out	613.23	902.21	0.6223
	✓	CV	721.02	1018.91	0.4194

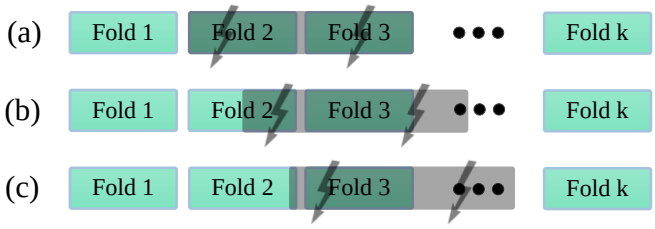

Fig. 4. Schematic of different attack scenarios:(a) 100-100% (b) 50-100-50% (c) 10-100-90%.

first splits the dataset into ten folds and consecutively took one of them as the test set in order to compute its R^2 value. Then, all values are compared to find the minimum, mean, and standard deviation. Finally, if the test set with the minimum R^2 value meet the predefined conditions, the detection process will be finished; if not, the process will be continued by removing the corresponding fold. To preserve more untouched data and increase the resolution of the proposed method, one can split data into more folds at the expense of more computational cost. Fig. 6 illustrates wind power forecasting accuracy obtained by XGBoost algorithm under scaling attacks on input data, in which the model is equipped with the proposed detection mechanism. As can be realized, corrupted data is successfully detected and removed from the dataset, therefore, the forecasting accuracy of the model is successfully recovered.

Second case considers when attackers get access to the output data of the forecasting system tamper with this data. Fig. 5c illustrates scaling attack on the output data leading to notable deviations from the normal operation for XGBoost forecasts. Table IV lists the results obtained from XGBoost model utilizing the proposed framework for all considered scenarios. Again, when a contaminated fold is chosen as the test set, the R^2 value is quite distinct from the counterparts, and the proposed framework effectively detected and removed compromised folds. The third case takes into account simultaneous FDI attacks on both input and output data, as shown in Fig. 5d. Table V presents details of results obtained by applying the proposed framework for detecting and removing compromised input-output data through defined scenarios. Clearly, the proposed framework extracted the false data from dataset and, thus, improved the accuracy of the forecasts. In short, injecting false data into input data makes more uncertainty and, thus, more impacts on the accuracy of wind power forecasting in comparison with attacking output data. Moreover, results indicate that the more corrupted data spread over dataset, the more pure data removed from dataset, thereby requiring a higher resolution for the detection. To further corroborate the advantage of the presented method, Table VI provides the forecasting accuracy of XGBoost model equipped with the proposed method and a combined robust forecasting model equipped with denoising variational autoencoder (DVAE-CRF) [37] for scaling attacks on output data. It can be clearly seen that the proposed method outperformed its counterpart in terms of less MAE and RMSE besides more R^2 values.

A generalizable model is a neither underfit nor overfit model aiming to make sensible predictions based on unseen validation datasets. The generalizability can be highly degraded in the case of data quality deterioration. In the previous study [28], the generalizability of long-term wind power forecasting models was confirmed by testing the trained model against data measured at a different location.

TABLE II
DESCRIPTIVE STATISTICS OF EMPLOYED DATASETS.

Site	Location	Time interval (10-min)	Samples	Min (A)	Mean (A)	Max (A)	Std
Ghadamgah	36.104° north 59.066° east	01/09/2015 11:40 - 05/03/2017 10:20	79334	0	4.85	20.80	3.85
Davarzan	36.210° north 56.524° east	17/07/2015 14:00 - 08/04/2017 10:10	83068	0	3.17	22.30	2.58
Jangaal	34.421° north 59.132° east	02/09/2015 10:20 - 13/04/2016 10:50	83692	0	3.68	18.50	1.99
Khaf	34.567° north 60.148° east	08/07/2015 13:50 - 26/01/2017 10:50	78852	0	10.45	31.20	6.066

TABLE III
THE EVOLUTION OF XGBOOST ALGORITHM EQUIPPED WITH THE PROPOSED DETECTOR USING CORRUPTED INPUT DATA

Scenario	Round	R^2 values	$R^2_{min} < \mu - 2\sigma$	$\sigma > 0.1$	Action
1	1	[0.2163, -1.0483, 0.9473, 0.9222, 0.9400, 0.9395, 0.9774, 0.9692, 0.9208, 0.8856]	✓	✓	Remove
	2	[0.0843, 0.9365, 0.9295, 0.9671, 0.9670, 0.9855, 0.9839, 0.9682, 0.9019]	✓	✓	Remove
	3	[0.9996, 0.9993, 0.9997, 0.9998, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
2	1	[0.6721, -0.5332, 0.0157, 0.9197, 0.9505, 0.9359, 0.9762, 0.9830, 0.9255, 0.9260]	✓	✓	Remove
	2	[0.5637, -0.1751, 0.9369, 0.9658, 0.9589, 0.9897, 0.9843, 0.9411, 0.9289]	✓	✓	Remove
	3	[0.5359, 0.9655, 0.9770, 0.9754, 0.9949, 0.9951, 0.9274, 0.9661]	✓	✓	Remove
	4	[0.9994, 0.9997, 0.9999, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
3	1	[0.9358, -0.8314, -0.5022, 0.9650, 0.9595, 0.9215, 0.9770, 0.9820, 0.9753, 0.9583]	✓	✓	Remove
	2	[0.9507, -0.9285, 0.97752, 0.9803, 0.9576, 0.9902 0.9946, 0.9920, 0.9734]	✓	✓	Remove
	3	[0.9418, 0.9966, 0.9971, 0.9886, 0.9996, 0.9985, 0.9990, 0.9970]	✓	×	Keep

TABLE IV
THE EVOLUTION OF XGBOOST ALGORITHM EQUIPPED WITH THE PROPOSED DETECTOR USING CORRUPTED OUTPUT DATA

Scenario	Round	R^2 values	$R^2_{min} < \mu - 2\sigma$	$\sigma > 0.1$	Action
1	1	[0.4600, 0.5455, 0.9104, 0.8698, 0.7912, 0.7242, 0.8402, 0.8280, 0.8215, 0.9158]	✓	✓	Remove
	2	[0.5519, 0.9752, 0.9530, 0.9756, 0.9735, 0.9814, 0.9632, 0.9739, 0.9867]	✓	✓	Remove
	3	[0.9996, 0.9993, 0.9997, 0.9998, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
2	1	[0.6351, 0.5520, 0.7336, 0.9610, 0.8450, 0.8540, 0.9588, 0.9267, 0.9398, 0.9670]	✓	✓	Remove
	2	[0.6380, 0.7641, 0.9617, 0.8988, 0.8758, 0.9624, 0.9412, 0.9640, 0.9820]	✓	✓	Remove
	3	[0.8215, 0.9953, 0.9950, 0.9971, 0.9951, 0.9918, 0.9899, 0.9957]	✓	✓	Remove
	4	[0.9994, 0.9997, 0.9999, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
3	1	[0.9442, 0.5214, 0.6356, 0.9546, 0.9698, 0.9599, 0.9813, 0.9788, 0.9636, 0.9825]	✓	✓	Remove
	2	[0.9706, 0.6506, 0.9732, 0.9882, 0.9909, 0.9856, 0.9823, 0.9856, 0.9928]	✓	✓	Remove
	3	[0.9748, 0.9946, 0.9991, 0.9968, 0.9966, 0.9939, 0.9953, 0.9995]	✓	×	Keep

TABLE V
THE EVOLUTION OF XGBOOST ALGORITHM EQUIPPED WITH THE PROPOSED DETECTOR USING CORRUPTED INPUT-OUTPUT DATA

Scenario	Round	R^2 values	$R^2_{min} < \mu - 2\sigma$	$\sigma > 0.1$	Action
1	1	[0.0760, -0.3932, 0.8900, 0.8087, 0.9052, 0.8509, 0.9490, 0.9558, 0.9403, 0.9124]	✓	✓	Remove
	2	[0.1724, 0.8347, 0.9005, 0.9305, 0.8900, 0.9670, 0.9608, 0.9244, 0.8854]	✓	✓	Remove
	3	[0.9996, 0.9993, 0.9997, 0.9998, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
2	1	[0.3755, -0.2299, 0.1211, 0.9322, 0.9295, 0.9125, 0.9800, 0.9715, 0.9669, 0.8892]	✓	✓	Remove
	2	[0.3696, -0.6066, 0.9414, 0.9468, 0.9555, 0.9835, 0.9782, 0.9857, 0.9040]	✓	✓	Remove
	3	[0.3723, 0.9580, 0.9597, 0.9611, 0.9883, 0.9749, 0.9944, 0.9506]	✓	✓	Remove
	4	[0.9994, 0.9997, 0.9999, 0.9999, 0.9999, 0.9994, 0.9998]	×	×	Keep
3	1	[0.8877, -0.3489, -0.5599, 0.9398, 0.9404, 0.9308, 0.9788, 0.9801, 0.9373, 0.9232]	✓	✓	Remove
	2	[0.9131, -0.4986, 0.9704, 0.9540, 0.9542, 0.9891, 0.9891, 0.9673, 0.9620]	✓	✓	Remove
	3	[0.9122, 0.9876, 0.9986, 0.9939, 0.9979, 0.9986, 0.9880, 0.9992]	✓	×	Keep

However, as Table VII shows, data integrity attacks can considerably deteriorate the generalizability by injecting false data into the dataset. Hence, the effectiveness of the proposed method in preserving the generalizability of the models under data integrity attacks is investigated here. Simulation results for various algorithms are summarized in Table VII, where the generalizability of models are demonstrated through predicting wind power at locations different from the model-trained location, i.e. Ghadamgah. Fig. 7 demonstrated that XGBoost model equipped with the proposed method could predict the generated power at the other locations with the accuracy of 5%-6% error in the case of MAE, except for Khaf wind farm that has a very diverse wind profile. This case validates the capability of the

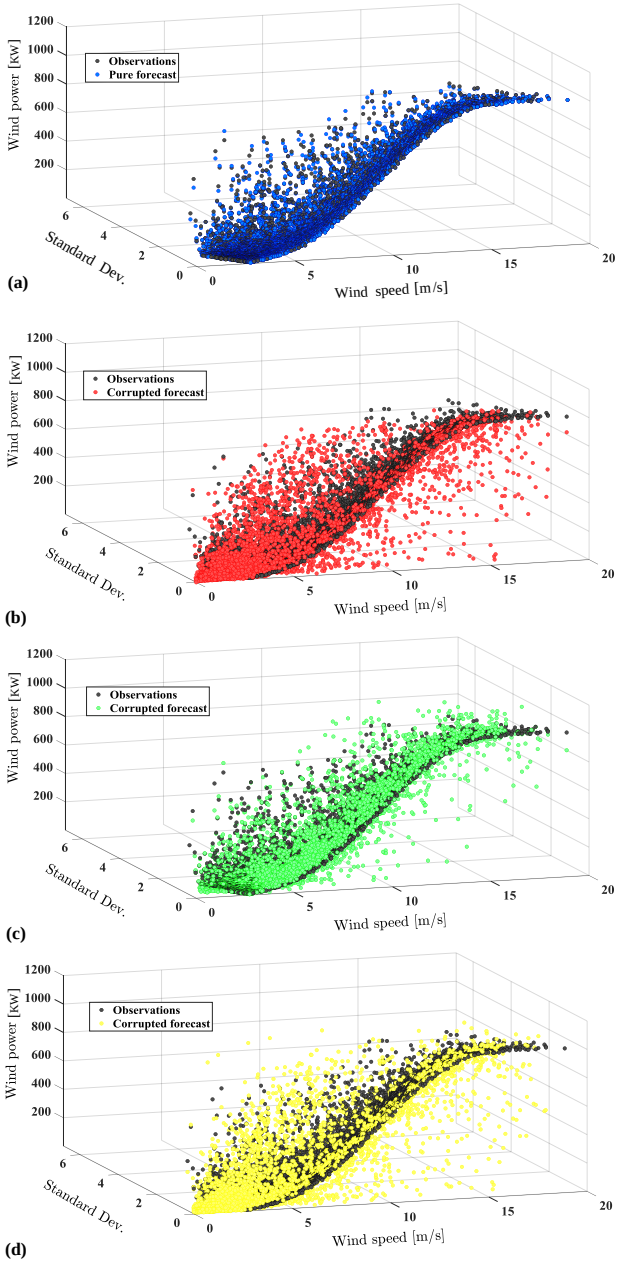

Fig. 5. Wind power forecasts under (a) normal condition, (b) scaling attacks on the input data (c), scaling attacks on the output data and (d) scaling attacks on the input-output data.

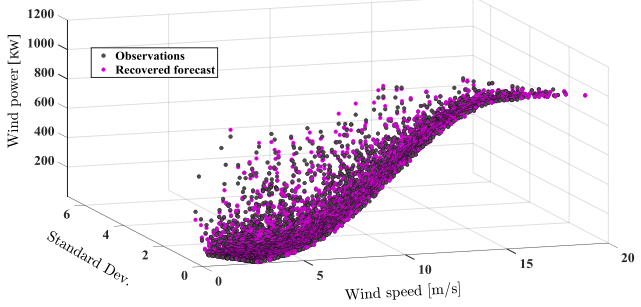

Fig. 6. Forecasting accuracy of equipped XGBoost model under the scaling attack.

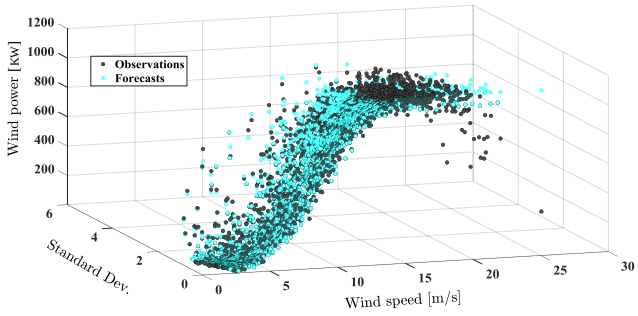

Fig. 7. Forecasting accuracy of recovered XGBoost model applied on Khaf.

proposed methodology in keeping the generalizability under cyber-attacks and making accurate predictions based on unseen datasets.

I. Bad vs false data detection

A critical feature of any model for detecting false data injection attacks is its ability to distinguish between bad data (measurement errors, communication interruptions, and so on) and false data injections. Bad data can come from many sources in different models and make data explicitly deviated from other observations, which can be detected by existing fault detectors [41]. We conduct two experiments to determine if the suggested methodology is capable of distinguishing between faulty and false data injections. In the first experiment, we first identify outliers (poor data) during the preprocessing step, resulting in a total of 1,848 outliers out of 79,334 samples (2.3 percent). To this end, without loss of generality, we used z-score metric to identify the outliers in the inputs. Then, we incorporate the erroneous data samples into the training phase to observe how the proposed detection system performs. The first row of Table VIII contains the acquired findings, where the suggested method identifies 3.9 percent of the occurrences of bad data as false data injection, while the rest samples are processed properly. This is due to the fact that bad data can be scattered over the dataset affecting the overall performance of the cross-validation not a specific iteration with specific fold. Due to the low amount of incorrect data discovered, the algorithm's overall accuracy suffers, which can be tackled by coupling widely used anomaly detectors like z-score metric. On the other side, this removes the possibility that system operators would confuse bad data with data injection assaults. In the second experiment, we replicate data scaling attacks before to the preprocessing step to see if preprocessing can detect data injected samples. Two of the 10 folds that cover all the data have been corrupted by data assaults (15,866 samples out of 79,334). The preprocessing phase detects just 6.3 percent of the intentionally inserted data; the remainder is ignored. The preprocessing technique, we think, can detect only the tail of the distribution associated with the injected data. This necessitates the use of sophisticated algorithms that have been specifically trained to identify purposeful data injection assaults.

TABLE VI
FORECASTING ACCURACY OF DIFFERENT MODELS

Algorithm	MAE	RMSE	R^2
Proposed XGBoost	5.52	20.44	0.9996
DVAE-CRF	212.27	704.43	0.7422

TABLE VII
FORECASTING ACCURACY OF XGBOOST APPLIED ON OTHER LOCATIONS

Proposed method	Location	Attack	MAE	RMSE	R^2
Not applied	**Khaf**	×	26.34	122.41	0.9945
		✓	811.91	983.76	0.5455
	Davarzan	×	6.32	25.43	0.9966
		✓	678.12	794.31	0.7549
	Jangaal	×	4.79	13.72	0.9984
		✓	703.40	816.14	0.7064
Applied	**Khaf**	×	24.27	90.09	0.9958
		✓	50.96	180.16	0.9944
	Davarzan	×	5.10	22.04	0.9997
		✓	5.44	27.43	0.9996
	Jangaal	×	3.67	11.57	0.9998
		✓	3.89	12.55	0.9998

TABLE VIII
DETECTION ACCURACY OF PROPOSED MECHANISM VS PREPROCESSING

	Bad data	False data	Normal data
Bad data	–	3.9%	96.1%
False data	6.3%	–	93.7%

IV. CONCLUSIONS

This paper addressed the reliability and resiliency of energy forecasting systems against FDI attacks. A new FDI attack detection framework was presented based on cross-validation, R^2 values and z-score metric to provide cyberattack resilient energy forecasts. The proposed method splits the dataset into k folds, iteratively takes one fold as test set and the remaining as train test, evaluates coefficient of determination, as a new least square criteria, for each iteration and finally, detects intrusions based on pre-specified thresholds and remove them from the dataset. Decision tree ensembles including decision tree, bagging, random forest, AdaBoost, gradient boosting and XGBoost, which are not considered in the cyber-resilient forecasting literature, are compared with robust models such as iteratively re-weighted least squares regression models and an L_1 regression model. Simulation results verified the better robustness of the decision tree ensembles regarding its counterparts. After training with the proposed mechanism, they also compared with a recently proposed combined robust forecasting model equipped with denoising variational autoencoder. Experiments substantiate the effectiveness and superiority of the presented method in providing accurate detections under attacks on input, output and input-output data with low computational cost and high accuracy and scalability without utilizing neither system's models nor parameters. Results revealed that the generalizability of the final forecasts is also recovered by detecting and removing the corrupted data.

The proposed framework requires proper tuning of three parameters, i.e k, h and σ_0. The adjusting of k has a key role in reaching a compromise between accuracy and cost, in which the more folds are defined, the more detecting resolution has resulted at the expense of the more computational cost, and vice versa. h is another decisive factor in accurately detecting anomalies, which shall be appropriately selected. The higher values of k results in a higher accuracy and computational cost, and vice versa. σ_0 is the early stopping threshold that prevents the proposed algorithm from overrunning. Based on conducted experiments, adopting ten folds with $h=2$ and $\sigma=0.1$ yields an acceptable trade-off between the wind power forecasting accuracy and computational cost.

The future work will focus on more and diverse types of attacks to verify the reliability and resiliency of the presented defense mechanism. It also can be employed for different applications ranging from dynamic line rating forecasting to demand response forecasting. Furthermore, the recovery of the contaminated data instead of removing it besides using an adaptive number of folds is also taken into account as one of future works.

V. ACKNOWLEDGEMENT

This publication is supported by award NPRP12S- 0125-190013 from the QNRF-Qatar National Research Fund, a member of The Qatar Foundation. The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the QNRF.

REFERENCES

[1] R. Deng, G. Xiao, R. Lu, H. Liang, and A. V. Vasilakos, "False data injection on state estimation in power systems—attacks, impacts, and defense: A survey," *IEEE Transactions on Industrial Informatics*, vol. 13, pp. 411–423, April 2017.

[2] S. Taheri, M. Jooshaki, and M. Moeini-Aghtaie, "Long-term planning of integrated local energy systems using deep learning algorithms," *International Journal of Electrical Power & Energy Systems*, vol. 129, p. 106855, 2021.

[3] Z. Kazemi, A. A. Safavi, F. Naseri, L. Urbas, and P. Setoodeh, "A secure hybrid dynamic-state estimation approach for power systems under false data injection attacks," *IEEE Transactions on Industrial Informatics*, vol. 16, pp. 7275–7286, Dec 2020.

[4] S. Taheri, R. Ghoraani, A. Pasban, M. Moeini-Aghtaie, and A. Safdarian, "Stochastic framework for planning studies of energy systems: a case of ehs," *IET Renewable Power Generation*, vol. 14, no. 3, pp. 435–444, 2020.

[5] A. Ahmadi, M. Nabipour, B. Mohammadi-Ivatloo, and V. Vahidinasab, "Ensemble learning-based dynamic line rating forecasting under cyberattacks," *IEEE Transactions on Power Delivery*, vol. 37, no. 1, pp. 230–238, 2022.

[6] F. Zhang, H. A. D. E. Kodituwakku, J. W. Hines, and J. Coble, "Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 4362–4369, July 2019.

[7] K. Ganz, M. Hinterstocker, and S. von Roon, "Toward intelligent industrial informatics: A review of current developments and future directions of artificial intelligence in industrial applications," *IEEE Industrial Electronics Magazine*, vol. 14, pp. 57–72, June 2020.

[8] X. Luo, J. Sun, L. Wang, W. Wang, W. Zhao, J. Wu, J. Wang, and Z. Zhang, "Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy," *IEEE Transactions on Industrial Informatics*, vol. 14, pp. 4963–4971, Nov 2018.

[9] R. R. Appino, J. Á. G. Ordiano, R. Mikut, T. Faulwasser, and V. Hagenmeyer, "On the use of probabilistic forecasts in scheduling of renewable energy sources coupled to storages," *Applied energy*, vol. 210, pp. 1207–1218, 2018.

[10] H. Liu, C. Chen, X. Lv, X. Wu, and M. Liu, "Deterministic wind energy forecasting: A review of intelligent predictors and auxiliary methods," *Energy Conversion and Management*, vol. 195, pp. 328–345, 2019.

[11] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "A review of deep learning for renewable energy forecasting," *Energy Conversion and Management*, vol. 198, p. 111799, 2019.

[12] M. U. Yousuf, I. Al-Bahadly, and E. Avci, "Current perspective on the accuracy of deterministic wind speed and power forecasting," *IEEE Access*, vol. 7, pp. 159547–159564, 2019.

[13] S. Taheri and A. Razban, "Long-term planning of integrated local energy systems using deep learning algorithms," *Building and Environment*, vol. 205, p. 108164, 2021.

[14] A. Ahmadi, M. Nabipour, B. Mohammadi-Ivatloo, A. M. Amani, S. Rho, and M. J. Piran, "Long-term wind power forecasting using tree-based learning algorithms," *IEEE Access*, vol. 8, pp. 151511–151522, 2020.

[15] S. Ahmed, Y. Lee, S.-H. Hyun, and I. Koo, "Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2765–2777, 2019.

[16] Q. Yang, D. An, R. Min, W. Yu, X. Yang, and W. Zhao, "On optimal pmu placement-based defense against data integrity attacks in smart grid," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1735–1750, 2017.

[17] M. Yue, T. Hong, and J. Wang, "Descriptive analytics-based anomaly detection for cyber-secure load forecasting," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 5964–5974, 2019.

[18] G. Chaojun, P. Jirutitijaroen, and M. Motani, "Detecting false data injection attacks in ac state estimation," *IEEE Transactions on Smart Grid*, vol. 6, pp. 2476–2483, Sep. 2015.

[19] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using kalman filter," *IEEE Transactions on Control of Network Systems*, vol. 1, pp. 370–379, Dec 2014.

[20] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Transactions on Smart Grid*, vol. 8, pp. 1630–1638, July 2017.

[21] A. S. Musleh, G. Chen, and Z. Y. Dong, "A survey on the detection algorithms for false data injection attacks in smart grids," *IEEE Transactions on Smart Grid*, vol. 11, pp. 2218–2234, May 2020.

[22] J. Cao, D. wang, Z. Qu, M. Cui, P. Xu, K. Xue, and K. Hu, "A novel false data injection attack detection model of the cyber-physical power system," *IEEE Access*, vol. 8, pp. 95109–95125, 2020.

[23] A. Al-Abassi, H. Karimipour, A. Dehghantanha, and R. M. Parizi, "An ensemble deep learning-based cyber-attack detection in industrial control system," *IEEE Access*, vol. 8, pp. 83965–83973, 2020.

[24] W. Xue and T. Wu, "Active learning-based xgboost for cyber physical system against generic ac false data injection attacks," *IEEE Access*, vol. 8, pp. 144575–144584, 2020.

[25] R. Xu, R. Wang, Z. Guan, L. Wu, J. Wu, and X. Du, "Achieving efficient detection against false data injection attacks in smart grid," *IEEE Access*, vol. 5, pp. 13787–13798, 2017.

[26] H. Wang, G. Wang, G. Li, J. Peng, and Y. Liu, "Deep belief network based deterministic and probabilistic wind speed forecasting approach," *Applied Energy*, vol. 182, pp. 80–93, 2016.

[27] Z. Wang, H. He, Z. Wan, and Y. L. Sun, "Coordinated topology attacks in smart grid using deep reinforcement learning," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2020.

[28] A. Ayad, H. E. Z. Farag, A. Youssef, and E. F. El-Saadany, "Detection of false data injection attacks in smart grids using recurrent neural networks," in *2018 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, Feb 2018.

[29] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Systems Journal*, vol. 11, pp. 1644–1652, Sep. 2017.

[30] Q. Wang, W. Tai, Y. Tang, and M. Ni, "Review of the false data injection attack against the cyber-physical power system," *IET Cyber-Physical Systems: Theory & Applications*, vol. 4, no. 2, pp. 101–107, 2019.

[31] J. Luo, T. Hong, and S.-C. Fang, "Benchmarking robustness of load forecasting models under data integrity attacks," *International Journal of Forecasting*, vol. 34, no. 1, pp. 89–104, 2018.

[32] J. Luo, T. Hong, and S.-C. Fang, "Robust regression models for load forecasting," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5397–5404, 2018.

[33] M. Yue, "An integrated anomaly detection method for load forecasting data under cyberattacks," *2017 IEEE Power & Energy Society General Meeting*, pp. 1–5, 2017.

[34] M. Yue, "Evaluation of a data analytic based anomaly detection method for load forecasting data," *In 2018 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5, 2018.

[35] J. Luo, T. Hong, and M. Yue, "Real-time anomaly detection for very short-term load forecasting," *Journal of Modern Power Systems and Clean Energy*, vol. 6, no. 2, pp. 235–243, 2018.

[36] M. Cui, J. Wang, and M. Yue, "Machine learning-based anomaly detection for load forecasting under cyberattacks," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5724–5734, 2019.

[37] R. Zheng, J. Gu, Z. Jin, H. Peng, and Y. Zhu, "Load forecasting under data corruption based on anomaly detection and combined robust regression," *International Transactions on Electrical Energy Systems*, vol. 30, no. 7, p. p.e12103, 2020.

[38] Z. Tang, J. Jiao, P. Zhang, M. Yue, C. Chen, and J. Yan, "Enabling cyberattack-resilient load forecasting through adversarial machine learning," *2019 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1–5, 2019.

[39] V. Kozitsin, I. Katser, and D. Lakontsev, "Online forecasting and anomaly detection based on the ARIMA model," *Applied Sciences*, vol. 11, p. 3194, 2021.

[40] Y. Zhang, F. Lin, and K. Wang, "Robustness of short-term wind power forecasting against false data injection attacks," *Energies*, vol. 13, no. 15, p. 3780, 2020.

[41] S. Taheri, A. Ahmadi, B. Mohammadi-Ivatloo, and S. Asadi, "Fault detection diagnostic for hvac systems via deep learning algorithms," *Energy and Buildings*, vol. 250, p. 111275, 2021.

Amirhossein Ahmadi (Student Member, IEEE) received the B.Sc. degree in electrical engineering-control systems from the University of Tabriz, Tabriz, Iran, in 2012, and the M.Sc. degree from Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2015. He was an instructor in Industrial Control Laboratory, Amirkabir University of Technology and a research assistant with the Department of Electrical Engineering, University of Tabriz. He is currently a PhD student in electrical and software engineering at the University of Calgary, Calgary, Canada. His main area of research is renewable energies integration, storage systems, electrical vehicles, forecasting and deep learning.

Mojtaba Nabipour (Student Member, IEEE) received the B.Sc. degree in mechanical engineering from Noshirvani University of Technology, Babol, Iran, in 2009, and the M.Sc. degree from Tarbiat Modares University, Tehran, Iran, in 2015. He is currently a research assistant with the Department of Mechanical Engineering, Tarbiat Modares University. His main area of research is additive manufacturing, composite materials, renewable energies, machine learning and deep learning.

Saman Taheri received the B.Sc. degree in mechanical engineering from University of Tehran, Iran, in 2015 and completed his M.S. degree at Sharif University of Technology, Iran, in 2018. He is currently a Ph.D. student at the Department of Mechanical and Energy Engineering at Purdue School of Engineering, Indian University-Purdue University In Indianapolis. His research interests include renewable energies integration, machine learning, optimization, energy and building, and energy conversion.

Behnam Mohammadi-Ivatloo (Senior Member, IEEE) received the B.Sc. degree (Hons.) in electrical engineering from the University of Tabriz, Tabriz, Iran, in 2006, and the M.Sc. and Ph.D. degrees (Hons.) from the Sharif University of Technology, Tehran, Iran, in 2008 and 2012, respectively. He is currently a Professor with the Faculty of Electrical and Computer Engineering, University of Tabriz and consulting member of the Information Technologies Application and Research Center, İstanbul Ticaret University, Istanbul, Turkey. His main area of research is operation and planning of intelligent energy systems.

Vahid Vahidinasab (Senior Member, IEEE) received his PhD in electrical engineering from the Iran University of Science and Technology, Tehran, Iran, in 2010. He is a Senior Lecturer (Assistant Professor) in Electrical Power Engineering at Nottingham Trent University, Nottingham, UK and the Director of the GridLab. His research interests include power and energy systems modelling, analysis and control, smart grids, energy systems integration, and energy markets. Dr Vahidinasab is an Associate Editor for the IEEE Transactions on Industry Applications.