Energy Reports 10 (2023) 2194–2205

Contents lists available at ScienceDirect

Energy Reports

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

Research Paper

Data-centric approach for online P-margin estimation from noisy phasor measurements

Felipe Proença de Albuquerque [a], Rafael Nascimento [a], Luisa H.B. Liboni [c], Ronaldo F. Ribeiro Pereira [b], Eduardo Coelho Marques da Costa [a,*]

[a] USP - University of São Paulo, Department of Electronic Systems Engineering, Polytechnic School, São Paulo, Brazil
[b] UFAC - Federal University of Acre, Acre, Brazil
[c] IFSP - Federal Institute of Education, Science, and Technology of São Paulo, São Paulo, Brazil

ARTICLE INFO

Article history:
Received 27 June 2023
Received in revised form 22 August 2023
Accepted 2 September 2023
Available online xxxx

Keywords:
Voltage stability
Load P margin estimation
Machine learning
Noise on phasor measurements
Explainable AI
Regression methods

ABSTRACT

A new estimation method for load P-margin of transmission systems is proposed by using machine learning techniques. The estimation solution uses a reduced number of features as inputs to the machine learning algorithm and does not rely on power flow measurements, avoiding using time-varying grid parameters. The method involves investigating the performance of several machine learning algorithms to undertake the estimation task and explore different data transformation processes, including an optimized feature selection scheme, enabling an enhanced performance of the machine learning algorithms. Moreover, the method comprises the use of different Explainable-AI approaches to better understand the behavior of the solution. The method's performance for different noise levels is widely studied by employing a noise model available in the recent technical literature. The mean absolute percentage error - MAPE and the root mean square error - RMSE are calculated for performance assessment. Numerical examples of the proposed technique are presented using the IEEE 14-bus test system, considering normal and contingency (N-1,N-2) conditions for a wide range of load cases.

1. Introduction

Voltage stability is crucial for the safe and reliable operation of power transmission grids, and have been gaining importance in recent years since modern systems operate very closely to the collapse point (Thasnas and Siritaratiwat, 2019) due to the rising demand, which makes the integrated power system heavily loaded (Goh et al., 2015). The term voltage stability refers to the ability of a power system to maintain steady voltages in all buses after being subject to a disturbance from a given initial condition (Kundur et al., 2004). There are several types of voltage stability. Considering the definition introduced by Kundur et al. (2004), this work focuses on long-term voltage stability (LTVS), which is a phenomenon in a scale of minutes associated with the inability of the system to provide adequate reactive support in at least one bus, line, or even entire areas of the power transmission system. Concerning the reasons which lead the system to instability, it is worth highlighting the excessive load in which the network operates and the outages of transformers (Löf et al., 1993). The sequence of voltage instability events leads to voltage collapse, ending in the blackout of the entire power system.

There are several methods to analyze LTVS, including modal analysis, P-V and Q-V curves, singular value decomposition of the state matrix that composes the entire system, sensitivity analysis, voltage stability indices based on the values obtained in the standard power flow, data mining approaches, and the continuation power flow (Chandra et al., 2016). Theoretically, the analysis of LTVS can be performed by increasing the system load up until the voltage collapse point in a bus. The voltage variation as the load increases can be shown by plotting the P-V curve, where the MW distance from the operating point to the critical collapse point is called the load active power (*LPM*) margin.

The *LPM* is a good quantitative indicator for evaluating how close the voltage instability condition is and provides information about the possibility of increasing the load of the system (Zhou et al., 2010). In addition, LPM is important as it is a load management and network demand study that provides necessary information to improve the voltage profile, economic dispatch (as well as system efficiency) and system stability. Besides, LPM is considered in other studies in order to match stochastic output power of renewable sources and manage electric vehicles (Al-Alawi and Islam, 2004; Aazami et al., 2011; Wang et al., 2015a).

Concerning the determination of *LPM*, the most commonplace method found in the literature is the continuation power flow (CPF). In such method, the initial step is obtained by using

* Corresponding author.
E-mail address: educosta@usp.br (E.C.M. da Costa).

https://doi.org/10.1016/j.egyr.2023.09.016

the standard power flow solution for the system. In sequence, the method applies a sequence of predictors and correctors to trace the P-V curve until the collapse point. Thus, the difference between the load at the collapse point and the initial load is defined as the load margin (Pourbagher et al., 2022). Nevertheless, it is difficult to use the CPF for a large network in real-time applications due to the long iterative solution involved, which leads to a high computational effort and eventual divergent cases.

In recent years, data availability has increased from modern monitoring equipment. Phasor Measurement Units (PMU) at some bars of the system enable measurement of real-time phasors, frequency, and power with a high reporting rate (Joshi and Verma, 2021). In this sense, such measurements can be incorporated into the voltage stability analysis by using state-of-the-art machine learning techniques.

The methods applied to LTVS analysis by employing a machine learning approach use standard measurements, obtained from the power flow equations, and phasor measurements obtained as features for the margin estimation. Nevertheless, the studies ignore that the grid parameters might vary through time, especially in overhead transmission lines (Milojević et al., 2018; Wang et al., 2015b). In this sense, the power flow measurements might not reflect the system's current state. Besides, the phasor measurements are modeled in a simplified way by using the Total Vector Error (TVE) approach, without considering the random aspect of the noise (Brown et al., 2016) and the different noise sources in data acquisition (Wehenkel et al., 2020). Also, the current methods found in the literature use many features without presenting an effective feature selection scheme that improves performance. Moreover, the rise of Explainable AI techniques allow for building more reliable models in which the user might understand the behavior of the features behind the black-box machine learning model.

Indeed, machine-learning-based decision-making is trending in today's era of abundant data, and Explainable AI techniques can increase the trustiness and understandability of such models. This way, the control operations may be done considering the most important features of the model, allowing the operator to predict some changes in the output and performance after a change in a specific feature.

Some papers have been proposed in the literature in order to deal with *LPM* estimation with machine learning solutions for the LTVS problem. In Zhou et al. (2010), the LTVS is investigated in the New England 39-bus test system by using an artificial neural network (ANN) with voltage magnitudes and phase angles of all buses and as its inputs. The authors in Dharmapala et al. (2020) propose a study of the *LPM* comparing some well-established machine learning regression tools. Different voltage stability indexes obtained from the actual system state are used as input features. This approach poses a higher computational effort since it is necessary to get the voltage stability indices before applying the method. Another solution was presented by Suganyadevi and Babulal (2014), where the accuracy of the support vector regression method is compared with the ANN accuracy. In Bento (2022), the author presents another solution for load margin stability estimation based on ANN, in that case, the method comprises the requirements of voltage stability and small-signal stability.

The solution shown in Li and Ajjarapu (2017) uses convolutional neural networks (CNN) to recognize complex patterns, especially with high nonlinearities, which is the case of tracking the saddle–node bifurcation (SNB) of the collapse point. However, convolutional neural networks demand a high computational burden, and the system dimension can be too large. An Extreme Learning Machine (ELM) approach to estimate the *LPM* is introduced by Villa-Acevedo et al. (2020). In Villa-Acevedo et al. (2020), the input features include voltage magnitudes and angles, power flows, and active and reactive power injections for all buses which, e.g., take into account the error on the parameter estimation, this can increase the final estimation error. Nevertheless, the time to optimize the hyperparameters, which compose the model, increases rapidly if the initial adjustment presents poor adherence. It is worth highlighting the contribution of Malbasa et al. (2017), where some machine learning tools are implemented considering the uncertainty of machine learning models and the computational burden of training and voltage instability prediction.

Actually, the number of features available in the entire system can represent a great issue in machine-learning-based solutions. In this sense, the authors in Mohammadi and Dehghani (2015) propose a method to reduce the number of features by using the Principal Component Analysis (PCA) technique, and thereafter the selected features are given to decision trees for classification and security assessment of power systems. In Bahmanyar and Karami (2014), the authors develop a method base on the Gram–Schmidt orthogonalization process along with an ANN-based sensitivity technique to use a less amount of input variables to estimate the *LPM* with sufficient accuracy.

In this context, the paper introduces a novel machine learning approach that relies solely on power injections and phasor measurements, thereby avoiding the utilization of measurements influenced by time-varying grid parameters. Regarding phasor measurements, a robust noise modeling approach was employed, encompassing both systematic and random errors characterized by different distributions. The solution was derived through a comparison of multiple machine learning algorithms, demonstrating its robustness across various noise levels. Lastly, the paper presents an exploration of Explainable AI methods in Load P-Margin estimation, yielding a feature selection approach that effectively reduces the dimensionality of the feature space and enhances the method's performance.

The remaining work is structured as follows. Section 2 presents a formulation of the problem of Load Active Margin estimation. Section 3 shows the developed process of building the database. Section 4 describes the methodology of deal with the problem as a supervised learning solution. Section 5 presents numerical results considering the test system IEEE 14 bus. Section 6 introduces the Explainable AI techniques in the studied problem. Section 7 shows the feature selection method derived from the Explainable AI method. Finally, Section 8 concludes the paper.

2. Problem formulation

The *LPM* is defined as the distance between the power at the voltage collapse situation and the actual power for the current point of operation, given the considered direction of load increase (Van Cutsem and Vournas, 2007; Zhou et al., 2010). Fig. 1 illustrates the *LPM* characteristics. This margin is calculated for each load bus, which shows to be an intuitive and understandable indicator of proximity to the voltage collapse point. By using such an indicator, it is possible to evaluate if some bus of the system operates close to instability, which allows the operator to take action in order to avoid voltage collapse.

Theoretically, the load margin can be calculated using the standard load flow repeated until the collapse point. However, the common power flow method diverges near to the voltage collapse point, since the power flow Jacobian becomes singular (Nirbhavane et al., 2021). In order to avoid such singularity, the CPF uses a sequence of predictor–corrector steps to trace the curve from the base case obtained from the standard load flow solution to the collapse point.

The value of the loading margin depends on the behavior of the load and generation in the entire system. The study of the

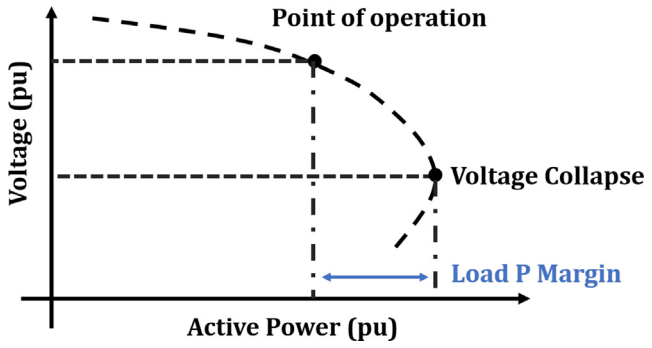

Fig. 1. Representation of the Load P Margin.

loading margin cannot be simplified as a simplistic case in which a load of a given bar is varied and other parts of the system are considered constant (Sode-Yome et al., 2006). To accommodate the increase in the load and generation, the dynamics can be described as

$$P_{L,i}(\lambda) = P_{L0,i}(1+\lambda) \tag{1}$$

$$Q_{L,i}(\lambda) = Q_{L0,i}(1+\lambda) \tag{2}$$

$$P_{G,i}(\lambda) = P_{G0,i}(1+\lambda), \tag{3}$$

where: λ defines the increasing in the load/generation, $P_{L,i}(\lambda)$ is the active load power at bus i, $Q_{L,i}(\lambda)$ is the reactive power at bus i, $P_{G,i}(\lambda)$ is the active generation power at bus i, and $P_{L0,i}$, $Q_{L0,i}$, and $P_{G0,i}$ are the initial conditions for these variables.

Using the proposed CPF alternative formulation for the increase in the load and generation indexes, the load flow equations can be rewritten as

$$P_{G,i}(\lambda) - P_{L,i}(\lambda) - |V_m|_i \sum_{j=1}^{n} |V_m|_j |Y|_{ij} \cos(\delta_i - \delta_j - \theta_{ij}) = 0 \tag{4}$$

$$Q_{G,i}(\lambda) - Q_{L,i}(\lambda) - |V_m|_i \sum_{j=1}^{n} |V_m|_j |Y|_{ij} \sin(\delta_i - \delta_j - \theta_{ij}) = 0, \tag{5}$$

where $|V_m|_i$ and δ_i are voltage magnitude and voltage angle at bus i, respectively. $|Y|_{ij}$ and θ_{ij} are admittance magnitude and admittance angle between buses i and j.

To use the continuation power flow method, the standard power flow equations are reformulated to incorporate the load parameter λ. Therefore the vector equation can be summarized by

$$\boldsymbol{F}(\boldsymbol{\delta}, \mathbf{V}, \lambda) = \mathbf{0}, \qquad 0 \leq \lambda \leq \lambda_{critical}. \tag{6}$$

In (6) $\boldsymbol{\delta}$ is the vector that represents the bus voltage angles, and $\boldsymbol{V}$ denotes the bus voltage magnitudes. The base case $(\boldsymbol{\delta_0}, \boldsymbol{V_0}, \lambda_0)$ is obtained using the standard power flow method. The next step of the solution is obtained by applying a continuation algorithm over the reformulated system of equations. Finally, the problem is solved until the critical point is obtained using a sequence of the predictor–corrector scheme to find solutions for different load levels. The complete description of the method can be found in Ajjarapu and Christy (1992).

Some voltage phasors are known for the case in which PMU measurements are available. Thus, the number of states to estimate is reduced. Besides, these values can be inserted into (6) in order to simplify the system dimension that can be solved for each load condition (Kumar et al., 2021).

3. Assembling of the database

The database was generated by obtaining various Continuation Power Flow (CPF) results under both normal and contingency situations for the analyzed system using the MATPOWER tool (Zimmerman et al., 1997). The database is composed of several CPF results under normal and contingency situations for the analyzed system. The study performed solutions for the N-1 and N-2 contingency situations, giving special attention to the outage of lines and transformers close to the target bus. It is recalled that N-1 is the situation in which the system operates without one element and N-2 with a lack of two elements.

Different load conditions are generated by applying a perturbation in the base case. According to Zhou et al. (2010), during a period of 24-h, a ±30 % variation of the load is considered suitable. In such work, the varying load is modeled by a uniform distribution. A normal distribution could be better to represent the load variation during a day since it presents less frequency for extreme values. The maximum variation of the load can be represented in the Gaussian distribution through the standard deviation:

$$P(\mu - 3\sigma \leq X \leq \mu + 3\sigma) = 0.9973, \tag{7}$$

where $P(\ )$ is the probability function, μ is the mean, X is the random variable, and σ is the standard deviation. Thus, for a specific active power at bus i, it is possible to write:

$$P_{L0,i}(1+0.3) = \mu + 3\sigma \Rightarrow \mu = P_{L0,i}, \ \sigma = 0.1P_{L0,i}, \tag{8}$$

for its maximum load.

Therefore, the different load active powers, load reactive powers, and generator active powers are given by

$$P_{L,i} \sim \mathcal{N}(P_{L0,i}, 0.1P_{L0,i}) \tag{9}$$

$$Q_{L,i} \sim \mathcal{N}(Q_{L0,i}, 0.1Q_{L0,i}) \tag{10}$$

$$P_{Gi} \sim \mathcal{N}(P_{G0,i}, 0.1P_{G0,i}). \tag{11}$$

Nevertheless, some generated points can be unfeasible, i.e, the solution of the standard power flow could not exist. To verify which points were useful, the load flow was performed, and when there was an existing solution, the result was stored as the initial point for the continuation power flow, i.e, the values of $P_{L0,i}$, $Q_{L0,i}$, $i = 1, 2, \ldots, n_{PQ}$, where n_{PQ} is the number of load buses, and $P_{G0,j}$, $i = 1, 2, \ldots, n_{gen}$, where n_{gen} is the number of buses with generation. After that, the CPF is performed until the collapse point of the bus i, corresponding to the value $P_{L\max,i}$. This way, the *LPM* is calculated as:

$$LPM = P_{L\max,i} - P_{L0,i}$$

The pseudocode of the algorithm used to build the database is presented in Algorithm 1.

4. Machine learning aspects

To obtain the *LPM*, the problem can be formulated as a regression problem with the following input features:

- Active and Reactive Power injections at each bus,
- Power Flow measurements between buses,
- Voltage phasors at buses with PMUs.

The target variable, or labels, correspond to the LPM of a specific bus.

Let $\mathcal{D}$ be the entire training data set. The elements of this set in supervised learning are defined as:

$$\mathcal{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\} \subseteq \mathcal{R}^d \times \mathcal{C} \tag{12}$$

where:

- $\mathcal{R}^d$ is the d-dimensional feature space;
- $\boldsymbol{x}_i$ is the input vector of the ith sample;
- y_i is the label of the ith sample;
- $\mathcal{C}$ is the label space (in regression problems $\mathcal{C} = \mathbb{R}$).

Algorithm 1 Generation Dataset

1: Designate the bus for analysis, called the target bus;
2: Create the matrix $\boldsymbol{M}$ of valid cases;
3: Define the total number of cases N_{cases};
4: State the initial values of $P_{L0,i}, Q_{L0,i}, i = 1, 2, \ldots, n_{PQ}$, and $P_{G0,j}, j = 1, 2, \ldots, n_{gen}$;
5: Generate N_{cases} different cases using: $P_{L,i} \sim \mathcal{N}(P_{L0,i}, 0.1P_{L0,i})$, $Q_{L,i} \sim \mathcal{N}(Q_{L0,i}, 0.1Q_{L0,i})$, $P_{Gj} \sim \mathcal{N}(P_{G0,j}, 0.1P_{G0,j})$.
6: Choose whether the case is normal, a contingency of type $N-1$, or $N-2$.
7: **if** $N-1$ or $N-2$ **then**
8: Remove one or two elements connected to target bus.
9: **end if**
10: **for** $k = 1, 2, \ldots, N_{cases}$ **do**
11: **if** $\{P_{L,i}(k), Q_{L,i}(k), P_{Gj}(k)\}, i = 1, 2, \ldots, n_{PQ}$, and $j = 1, 2, \ldots, n_{gen}$ generates a valid power flow **then**
12: Run the CPF until the target bus reaches its collapse point.;
13: Calculate $LPM = P_{L\max,target} - P_{L0,target}$;
14: Store the variables in $\boldsymbol{M}$.
15: **end if**
16: **end for**

The solution aims to get a function h that belongs to hypothesis class $\mathcal{H}$, such that $h(\boldsymbol{x}) = y$ for an instance $\boldsymbol{x} \notin \mathcal{D}$. The process of learning $h \in \mathcal{H}$ is summarized as follows

$$\text{Learning: } h^*(\cdot) = \text{argmin}_{h(\cdot)\in\mathcal{H}} \frac{1}{|\mathcal{D}_{\text{TR}}|} \sum_{(\boldsymbol{x},y)\in\mathcal{D}_{\text{TR}}} \ell(\boldsymbol{x}, y|h(\cdot))$$

where $\mathcal{D}_{TR}$ is the sample of $\mathcal{D}$ used to train the model, the operator $|\mathcal{S}|$ gives the number of elements of a set $\mathcal{S}$, $\ell(\boldsymbol{x}, y|h(\cdot))$ is the loss function of $(\boldsymbol{x}, y)$ given the function $h(\,.\,)$. Here, the loss function is the mean square error, given by

$$\ell(\boldsymbol{x}, y|h(\cdot)) = \frac{1}{n}\sum_{i=1}^{n}(h(\boldsymbol{x}_i) - y_i)^2. \tag{13}$$

The performance in the test database will be evaluated using the mean absolute percentage error (MAPE)

$$MAPE = \frac{100}{|\mathcal{D}_{Te}|}\sum_{i=1}^{|\mathcal{D}_{Te}|}\frac{|y_i - h(\boldsymbol{x})|}{|y_i|} \tag{14}$$

and the root mean square error (RMSE)

$$RMSE = \sqrt{\frac{1}{|\mathcal{D}_{Te}|}\sum_{i=1}^{|\mathcal{D}_{Te}|}(y_i - h(\boldsymbol{x}))^2} \tag{15}$$

where $\mathcal{D}_{Te}$ is the sample of $\mathcal{D}$ used to test the model.

The entire database was divided into training data and testing data, following a proportion of 80% for training and 20% for testing. This procedure ensures that the model remains completely blind to the testing data, making it possible to consider the testing data as unseen by the model. The model is obtained using the training data and applying K-fold cross-validation, where the model is trained on k-1 folds and validated on the remaining fold. The folds are randomly generated. This approach allows for obtaining the best model that can effectively generalize the regression.

4.1. Validation metrics

There are several indices to evaluate the performance of a regression method. It is possible to cite: Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and R squared (R^2) or adjusted R^2. It is important to emphasize that the R squared is useless in general problems, with strong nonlinearities as it is the problem of load margin estimation, since this index presents the following characteristics (Berk, 2004):

- R-squared does not measure goodness of fit;
- R-squared does not measure predictive error;
- R-squared does not allow comparing models using transformed responses;
- R-squared does not measure how one variable explains another.

In this sense, this work avoided to use the R squared, since even with a high value it is not possible to affirm that the method presents a good fit (Berk, 2004). Rather, the Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) were chosen in order to evaluate the solution.

5. Numerical analysis

5.1. Simulation structure

The methods to estimate the *LPM* was developed using the IEEE 14-bus test system. This system is composed of 5 generators, 9 load buses with static loads, and one fixed shunt. Generators at buses 3, 6, and 8 are operated as synchronous condensers, according to Fig. 2. Due to the large cost of implementation, it is unfeasible to consider PMUs on all buses in the system. However, there is a minimum number of PMUs in order to guarantee the complete observability of the entire power system. A comprehensive review brought by Yuill et al. (2011) showed that the suitable number of PMUs in the IEEE 14-bus system is three, whereas Hajian et al. (2007) proposes that the optimal location is at the buses 2, 6, and 9, which was considered in this work. The bus chosen to study was bus 4 which is a critical bus since it was connected to generation, transformers, and lines.

The cases to train the model were generated by using the software MATLAB and the tool MATPOWER (Zimmerman et al., 2011) where it is possible to model the test system by means of the admittance matrix of the system, available in Freris and Sasson (1968), and create several studies of the load margin by performing the continuation power flow, including normal and contingency conditions. Initially, it was created 10000 operation points with their corresponding load margin after the application of the CPF in order to provide sufficient information to the models. Besides, it is known that the size and performance characteristics of each system is correlated with number of samples (Kalogirou, 2000; Villa-Acevedo et al., 2020; Dharmapala et al., 2020; Zhou et al., 2010). So, it was chosen 10000 operation points because this number fits well the probability density function of IEEE 14 bus. This number of operation points is a good initial condition.

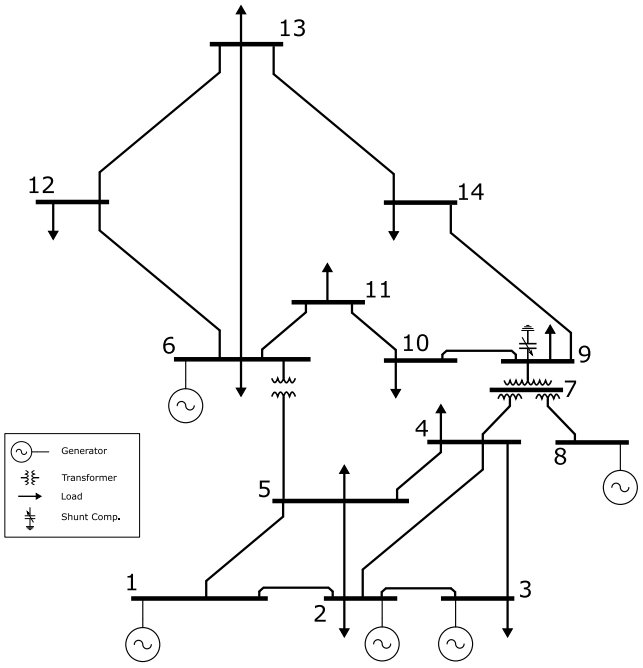

Fig. 2. IEEE 14 bus system.

5.2. Feature analysis

In a data mining approach, where the data is the center of the solution instead of the model, the selection of suitable features is a crucial aspect. In a field application, simulated in this work, the data available for the model could be considered:

- Active and Reactive Power injections in each bus;
- Power Flow measurements between each bus;
- Voltage phasors in buses with PMUs.

However, the power flow measurements require calculations that could lead to a large computational cost depending on the system dimension. Besides, these measurements are obtained by using the network parameters that including transmission lines. It is known in the literature that the stored parameters of a transmission line could differ to the actual value up to 30% (Kusic and Garrison, 2004). This way, insert measurements that depend on these parameters might introduce intrinsic errors into the machine learning model. A way to deal with this issue is to avoid the use of power flow measurements. Therefore, the features were chosen as the active/reactive power injections in each bus and voltage phasors in buses with PMUs which represents a novelty in the load estimation works, since the previous papers use the load flow measurements as inputs in machine learning solutions.

5.3. Estimation normal conditions

In order to prove the robustness of the proposed method, the test was divided into two situations: normal conditions and contingency conditions. In the algorithm, it is possible to determine the studied case by using a flag inserted when the database was built. It is worth highlighting that the training process is not affected by the flag, the only use of this differentiation is to compute the validation metrics for each case. Using the training data, the first qualitative analysis that can be done is the correlation matrix, the result is presented in Table 1. The Pearson correlation coefficient between the random variables X and Y is calculated as shown in Eq. (16).

$$\rho = \frac{Cov(X, Y)}{\sqrt{Var(X)Var(Y)}}, \tag{16}$$

where $Cov(X, Y)$ is the covariance between X and Y, $Var(X)$ represents the variance of the random variable X, and $\rho \in [-1, 1]$.

The first important aspect shown in the result is the high correlation among the phasor measurements. Another point is that these measurements also present a high correlation with the target. However, this behavior does not imply a high influence on the machine learning model, since the correlation coefficient only analyzes a two by two relation, whereas the model is a mixed relation of the features. Besides, in order to simplify the visualization, some features do not appear in the correlation matrix, this occurs because these features present a small variation in the training database, or do not have a relevant correlation with the target. This way, for that fold, these features can be neglected, since they can be described by a combination of other features.

In the initial analysis, it was utilized an Auto-ML framework called PyCaret (Ali, 2020), which incorporates various machine learning algorithms such as multiple linear regression, tree-based methods, and ensemble algorithms. This framework allowed to select the best model that suits the data's behavior from the training database. For this analysis, it was employed a validation approach with $k = 5$ folds to assess the results and compare different machine learning algorithms. In practice, typical choice of K is between 5 and 10. Burman (1989) argued that $K \leq 5$ might cause an issue. The performance comparison of these algorithms, utilizing the cross-validation process, is presented in Table 2. Additionally, the table illustrates the computational burden for both the training and testing processes. All the computations were carried out using one Intel Core i5 9600k, 8 GB RAM, a mATX Z390 m chipset family and no GPU were used in these processes. Finally, the solution's flowchart is presented in Fig. 3.

Table 2 shows that the Extra Trees Regressor presents the best performance in the training database. Therefore, this algorithm was chosen to use in the test database, remembering that the trained model does not have the information about the test database. The result for the test base is shown in Table 3.

Concerning the Extra-Trees regressor, it is a generalization of the Random Forest regressor introduced in Geurts et al. (2006). This algorithm belongs to a class of tree-based ensemble methods in which several trees are used at the same time in order to deal with classification and regression problems. Basically, there are two main differences between the Extra-Tree regressor and the Random Forest. First, the Extra Tree uses the entire database to train the different decision trees in the ensemble, whereas the Random forest applies the method in a random subset of features. The other difference lies in the way that the best feature and the corresponding value on the node are obtained in the node-splitting step (John et al., 2015). These differences lead the Extra Tree regressor to present less overfitting and better performance compared with the standard Random Forest algorithm (Geurts et al., 2006). The best machine learning model chosen in the Auto-ML approach was built using the standard library for machine learning tools in python called *scikit-learn*. In that library, it was possible to search for the best parameters of the model considering as metric the mean percentage absolute error (MAPE), such process is called grid search. The result of the grid search is presented in Appendix.

The mean error or MAPE presented in this work has the same magnitude that the one obtained in Zhou et al. (2010), but without using power flow measurements and artificial neural network which leads to a large computational cost compared with standard machine learning methods.

Table 1
Correlation Matrix in the training database.

Measure	V_9	θ_2	θ_6	θ_9	P_{L6}	$P_{L,11}$	$P_{L,12}$	$P_{L,13}$	Q_4	LM
V_9	1	0.98	0.94	0.97	−0.033	0.099	−0.012	0.03	0.12	0.95
θ_2	0.98	1	0.97	0.99	0.053	0.004	0.014	0.025	0.0088	0.98
θ_6	0.94	0.97	1	0.99	0.2	0.04	0.08	0.11	0.12	0.95
θ_9	0.97	0.99	0.99	1	0.096	0.054	0.038	0.067	0.053	0.98
P_{L6}	−0.033	0.053	0.2	0.096	1	−0.18	−0.17	−0.19	0.22	0.065
$P_{L,11}$	0.099	0.004	0.04	0.054	−0.18	1	−0.11	−0.12	0.14	0.018
$P_{L,12}$	−0.012	0.014	0.08	0.038	−0.17	−0.11	1	−0.11	0.13	0.025
$P_{L,13}$	0.03	0.025	0.11	0.067	−0.19	−0.12	−0.11	1	0.14	0.074
Q_4	0.12	0.0088	0.12	0.053	0.22	0.14	0.13	0.14	1	0.074
LM	0.95	0.98	0.95	0.98	0.065	0.018	0.025	0.039	0.074	1

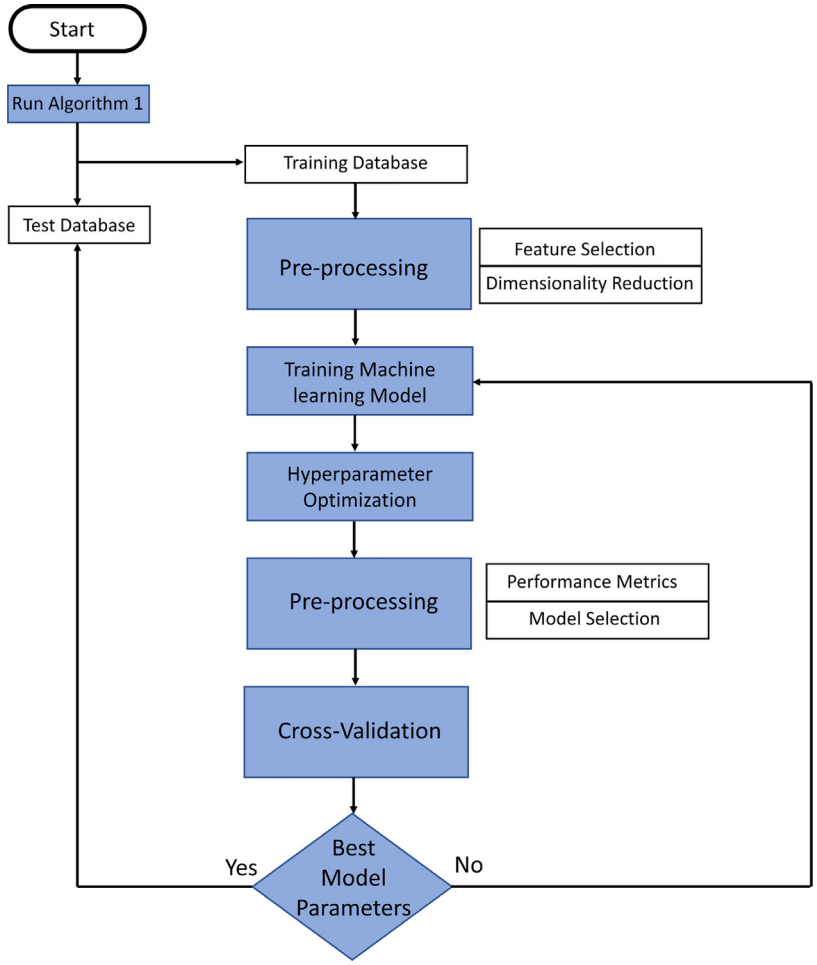

Fig. 3. Flowchart of the proposed solution.

In order to improve the performance of the method, it is possible to carry out some transformations in the base. The first suitable operation in the database in regression analysis is to remove multicollinearity which occurs when independent variables in the regression model are highly correlated to each other. This might lead to a certain problems in the regression since it could be hard to interpret the model and also creates an overfitting problem because of the unstable estimation of the coefficients in the model (Daoud, 2017). A simple way to remove the multicollinearity is to establish a threshold for the correlation coefficient (ρ) between features, in this work it is fixed in 0.9.

Another data transformation popular in machine learning solutions is data standardization where the data is scaled or transformed to make an equal contribution of each feature. Some works in the literature prove the efficacy of this transformation (Singh and Singh, 2020; Shanker et al., 1996). There are several methods for normalization, in this work was employed Z-score, where each instance, $x_{i,n}$ of the data is transformed into $x'_{i,n}$ as follows:

$$x'_{i,n} = \frac{x_{i,n} - \mu_i}{\sigma_i} \tag{17}$$

where μ and σ denote the mean and standard deviation of *i-th* feature respectively. The resulted correlation matrix after the data transformation processes is shown in Table 4. In that Table, it is possible to see that the remaining features present a small two by two correlation which shows the lack of multicollinearity.

The new performance in the training database after the use of the data transformation (standardization and remove of multicollinearity) is presented in Table 5. In general, for all algorithms, the performance in the training database is worse with the data transformation which is not a bad thing, since an extremely high

Table 2
Performance of the algorithms in the training database.

Algorithm	RMSE (MW)	MAPE (%)	Computational Burden (s)
Extra Trees	0.5573	0.58	2.6820
Random Forest	1.2033	0.99	6.4100
Light Gradient Boosting	1.2645	1.42	1.7200
Decision Tree	1.9947	1.11	1.1460
K-Neighbors	2.3958	9.31	1.1020
Gradient Boosting	2.4027	3.43	3.2360
Linear Regression	8.0773	23.48	1.4960
Bayesian Ridge	8.0777	23.48	1.700
AdaBoost	12.79	42.54	1.7620
Ridge	21.33	79.82	1.0940
Huber	23.73	89.99	1.1920
Lasso	24.61	91.3	1.2640
Elastic Net	33.49	92.4	1.0940
CPF	–	–	3.365

Table 3
Performance of the Extra Trees Regressor in the test database.

Algorithm	RMSE (MW)	MAPE (%)
Extra Trees Regressor	1.7195	0.8598

Table 4
Correlation Matrix after removing the multicollinearity.

Measure	θ_2	P_{L6}	$P_{L,11}$	$P_{L,12}$	$P_{L,13}$	Q_4
θ_2	1	0.053	0.004	0.014	0.025	0.0088
P_{L6}	−0.053	1	−0.18	−0.17	−0.19	0.22
$P_{L,11}$	0.004	−0.18	1	−0.11	−0.12	0.14
$P_{L,12}$	−0.014	−0.17	−0.11	1	−0.11	0.13
$P_{L,13}$	0.025	−0.19	−0.12	−0.11	1	0.14
Q_4	0.0088	0.22	0.14	0.13	0.14	1

Table 5
Performance of the algorithms in the training database after data transformation.

Algorithm	RMSE (MW)	MAPE (%)
Extra Trees Regressor	0.9218	0.91
Light Gradient Boosting Machine	1.1743	2.26
Gradient Boosting Regressor	1.977	2.95
Random Forest Regressor	1.555	1.37
K-Neighbors Regressor	2.388	9.16
Decision Tree Regressor	2.275	1.55
AdaBoost Regressor	13.347	44.06
Bayesian Ridge	21.044	67.08
Ridge Regression	25.43	81.23
Huber Regressor	31.44	92.09
Lasso Regression	33.33	94.51
Elastic Net	38.92	98.02

Table 6
Performance of the Extra Trees Regressor in the test database after data transformation.

Algorithm	RMSE (MW)	MAPE (%)
Extra Trees Regressor	0.8472	0.6895

performance in the training database might represent overfitting. This fact can be observed in Table 6, where the performance in the test database is presented. Such table shows that the performance in the test database is better with the data transformation process which proves the greater ability of the model to generalize the result.

5.4. Estimation contingency situations

The contingency situations are defined as the situation where the power system operates with an outage of one or more components since the generated situation is credible, that is if the system might remain operating without the components (Bulat et al., 2021). This work analyzes the load margin estimation in which the system operates with N-1 and N-2 devices, giving relevant importance to outages around the bus analyzed (bus 4) since that region shows a more influence on the load margin for the studied bus. The machine learning algorithm employed to estimate the load P margin under contingency was the Extra-Tree regressor, since that algorithm showed the best performance for the first analysis. The performance in the train and test set is presented in Table 7.

Table 7
Performance of the Extra Trees Regressor in the database for contingency situations.

Database	Algorithm	RMSE (MW)	MAPE (%)
Train	Extra Trees Regressor	2.071	0.923
Test	Extra Trees Regressor	14.402	6.5074

Table 8
Performance of the algorithms in the database using N-1 and N-2 flags.

Database	Algorithm	RMSE (MW)	MAPE (%)
Train	Extra Trees Regressor	1.784	0.887
Test	Extra Trees Regressor	12.333	6.0512

Table 9
Performance of the algorithms in the database using polynomial features.

Database	Polynomial Features	RMSE (MW)	MAPE (%)	Total computational time (s)
Test	No	12.333	6.0512	2.6820
Test	Yes	10.515	5.7564	10.435

Comparing with the previous analysis, this result presented a worse performance for both training and test data. A way to deal with such poor performance is to create a new feature to help the learning information process of the method. In this sense, it was created two new features: a flag for the case of N-1 contingency situation and a flag for N-2, i.e it was incorporated into the database if in the studied case there was a lack of 1 or 2 elements. Besides, if N-1 and N-2 are both equal to zero then the case corresponds to a normal condition. Using this new set of features, the performance in the entire database becomes better which is represented in Table 8.

It is possible to improve the performance of the method in the contingency database by using others features transformation processes. A simple method of data transformation process is the use of polynomial combinations of the initial features. In that process, it is generated new features as polynomial combinations with a maximum degree previously fixed. For example, supposing the existence of features X_1, X_2, if we choose polynomial transformation with degree 2, the resulting dataset will contain the features $X_1, X_2, X_1X_2, X_1^2, X_2^2, X_1^2X_2^2$. At the same time that the creation of new features might improve the performance of the machine learning solution, certainly it will result in a more computational burden. For that reason, it is presented in Table 9 the performance of the Extra Tree Regressor considering polynomial features with degree 2 and the computational time used to train the model and predict the values in the test base. As it was expected the computational time increased compared with the case without the polynomial features, which could be an issue for large systems.

5.5. Comparison with previous work in the literature

To highlight the effectiveness of the proposed method, a comparative analysis will be conducted against existing literature.

It is crucial to underscore that, whenever feasible, the method was replicated within the identical computational environment as the proposed approach, allowing for a meaningful comparison of computational burdens. In Suganyadevi and Babulal (2014), a solution based on support vector machines within a regression framework was proposed. In that context, through the application of grid search for kernel types and parameters C and γ, the best result achieved was a Mean Square Error (MSE) of 1.30466. In contrast, utilizing the same metric, the proposed solution demonstrated a significantly lower MSE of 0.3160. In terms of computational burden, the SVM solution required 9.63 s for completion, while the Extra Tree method took only 2.6820 s. These timings encompass both the training and testing processes, considering the same test system.

The paper presented in Kumar et al. (2021) has shown a solution based on Artificial Neural Network, an extreme powerful deep learning tool. Such solution requires the use of Particle Swarm Optimization algorithm to find the meta-parameters and tune the ANN. In that case, the maximum observed error was 0.28%, which it is greater that presented in that paper. Implemented in the same computational environment as the proposed solution and using the parameters described in Kumar et al. (2021), the algorithms took 254 s or 320 epochs to reach the mentioned performance.

Another solution using ANN is presented by Zhou et al. (2010), the Mean Absolute Error for the best feature set was 0.1205%, whereas the computational time considering the presented meta-parameters was 44.5 s.

A solution based on coupled single-port Thevenin equivalent model and the cubic spline extrapolation technique is presented by Su and Liu (2015). Considering no measurement errors and the best result, the method proposed by the authors reaches 1.78%. The authors did not give information about the computational burden.

The solution presented by Dharmapala et al. (2020) uses several Voltage Stability Indices (VSIs) as inputs to an machine learning models to predict the Voltage Stability Margin. Considering the similar database employed in the presented work, the best result was obtained using the Random Forest Regressor (RFR) with a R^2 of 0.9981. Using the same metric of performance, the Extra Tree Regressor presents a R^2 of 0.9983. Also, the time to get the solution using RFR is 8.3 s.

Essentially, when comparing the proposed approach with other methods in the literature, it is plausible to assert that within the realm of machine learning algorithms, the Extra Tree Regressor, categorized as a bagging algorithm, emerges as a favorable choice in terms of both performance and computational efficiency. A better result might be accomplished using deep learning; however, this implies a significant computational cost and requires additional methods to initialize the network. The conclusion reached is consistent with recent papers in machine learning that assess the efficacy of algorithms when dealing with tabular databases (Grinsztajn et al., 2022).

5.6. Estimation load margin considering noise

The noise modeling in phasor measurements is a crucial aspect in problems that deal with data obtained from PMUs. Several methods have been proposed in the literature to introduce noise, i.e random errors, in phasor measurements. The main part of them either represents noise as additive term in each equation (Li et al., 2017; Du and Liao, 2012; Dasgupta and Soman, 2013) or assume unrealistic approximations, for example, that the projections in the real and imaginary axes ever follow a Gaussian distribution independent of noise level (Wehenkel et al., 2020) which was shown that is not true for all cases (De Albuquerque et al., 2021).

Table 10
Maximum errors for different classes of ITs.

Class of operation	Max. magnitude error (α) [%]	Max. phase error (β) [rad]
0.1	0.1	1.5×10^{-3}
0.2	0.2	3×10^{-3}
0.5	0.5	9×10^{-3}
1.0	1	18×10^{-3}

Table 11
Performance of the Extra Tree regressor in the database considering noise.

Base	Algorithm	RMSE (MW)	MAPE (%)
Train	Extra Trees Regressor	1.58971	0.65987
Test	Extra Trees Regressor	2.06476	1.09796

A rigorous approach for noise modeling on phasor measurements was presented in Pegoraro et al. (2019). Consider a generic phasor measurement given in polar coordinates $\bar{w} = \bar{X}e^{\bar{\phi}}$, the random error is inserted into the model asa zero mean Gaussian distribution upon the magnitude, denoted by $\Delta_{\rho,x}$, and a different normal distribution in the phase, described by $\Delta_{\phi,x}$. Besides, the systematic error is modeled as a constant term for each sample, i.e, it is considered as an uniform distribution for the magnitude ξ_ρ and another for the phase ξ_ϕ.

This way, a field phasor measurement might be modeled accurately by:

$$w = (\bar{X} + \Delta_{\rho,x} + \xi_\rho)e^{j(\bar{\phi}+\Delta_{\phi,x}+\xi_\phi)}$$

To produce a realistic setup the following assumption was used according to Pegoraro et al. (2019):

- Voltage transformers are assumed to be of class 0.5, considering Table 10. The errors and deviations have uniform distribution, in which the standard deviation is equal to $\Delta/\sqrt{3}$, where Δ is the absolute value of the maximum deviation. For example, in the class 0.5, $\Delta = 0.5\%$. Basically, this defines the systematic errors in phasor measurements;
- As for PMUs, a maximum total vector error (TVE) of 0.1% has been considered by splitting it into a maximum amplitude error of 0.1% and a maximum phase angle error of $10^{-3}rad$.

The random error is given by a Gaussian distribution and the error of the PMUs is provided in maximum values. It is possible to insert the information of the maximum value into the random model by controlling the standard deviation of the distribution. More specifically, the standard deviation is chosen as the maximum value of the error divided by three. This way, as the measures follow a Gaussian distribution, the measurements belong to an interval bounded for the maximum value with a probability of 0.9973 (Wehenkel et al., 2020). It is important to emphasize that this proposed paper presents rigorous noise modeling in accordance with the state of the art in noise modeling on phasor measurements (Wehenkel et al., 2020; De Albuquerque et al., 2021; Brown et al., 2016; Dobakhshari et al., 2020). Rather, the previous papers present in the literature (Dharmapala et al., 2020; Zhou et al., 2010) consider the noise by using the total vector error approach which consists of a model only for the PMU and does not consider the random aspect of the error and even do not consider the systematic errors in phasor measurements.

Considering the measurement errors process described in this section, the algorithm with the best result in the base case was applied to the database considering a generic case, without a differentiation between normal and contingency conditions which allows an analysis for the noise influence in both conditions. Such result is shown in Table 11.

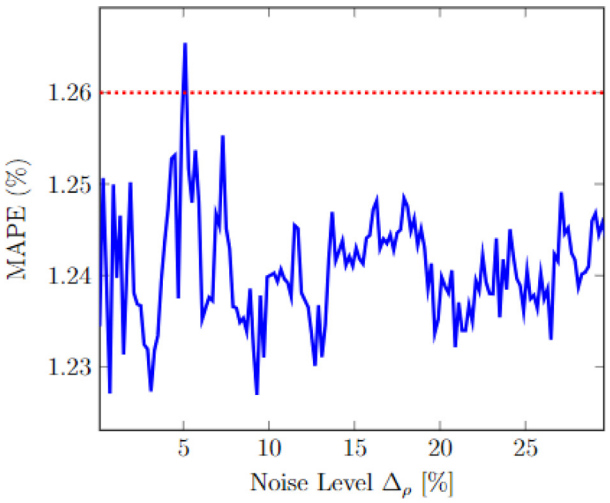

Fig. 4. Influence of magnitude noise level on regression performance.

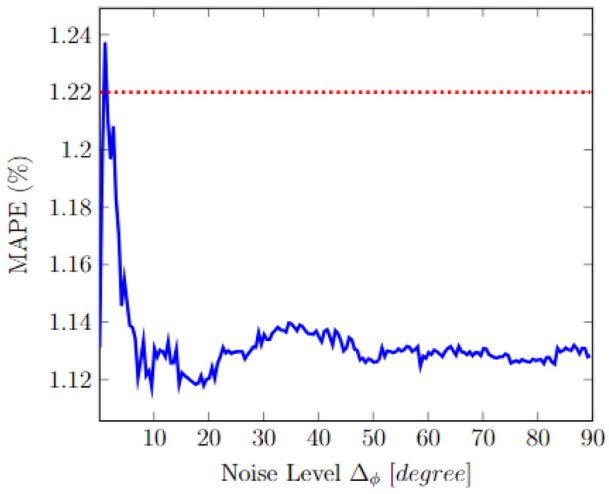

Fig. 5. Influence of phase noise level on regression performance.

Comparing the result without and considering noise, it was observed a difference in performance, in which the case with noise presented a worse performance. The same behavior was obtained in previous papers in the literature that deal with load margin estimation by means phasor measurements (Zhou et al., 2010; Dharmapala et al., 2020). This fact occurs because it is more difficult to generalize a model when the data is affected by inaccuracies.

Despite the works presenting a result considering measurement error, the study of the noise influence is not carried out. In this sense, the study of the noise influence in the estimation method is proposed in this work, considering several noise levels for the magnitude (α) and phase (β) in the random errors, introduced in Table 10. For the first study, the noise level is fixed for the phase in $\beta = 18 \times 10^{-3}$ rad and vary the noise level for the magnitude from 0.1 % to 30%, the result is presented in Fig. 4. The behavior observed in Fig. 4 shows the power of machine learning methods of regression concerning noise since the Extra Tree Regressor presents a good performance even with high noise levels. Similar behavior is shown in Fig. 5, where the influence of noise level in phase is analyzed and the noise level of magnitude is fixed in $\alpha = 1$ %. It is important to note that the noise levels considered are much greater than the ones observed in the field case. The purpose of this study is to show that even with extreme corrupted measurements, the regression method might present an acceptable performance for steady-state applications.

6. Explainable AI

The spread of machine learning solutions in several fields of knowledge is not followed by the use of such solutions in real applications. The main reason for this fact is that the models are built to focus only on the performance indexes, such as accuracy for classification problems. Although the evaluation metrics play a relevant role in the machine learning solution, it is not possible to trust in a black box approach only by observing the output, because for certain problems or tasks it is not enough to get the prediction. The model must also explain how it came to the prediction since a correct prediction only partially solves your original problem. Additionally, learning how the model deal with data can improve the knowledge of the problem, predict when the solution might fail, and even learn how the features interact with each other inside the machine learning algorithm.

The interpretability can be achieved using two different approaches: by restricting the complexity of the machine learning model (intrinsic) or by applying methods that analyze the model after training (post hoc). Intrinsic interpretability refers to machine learning models that are considered interpretable due to their simple structure, such as short decision trees or sparse linear models. Post hoc interpretability refers to the application of interpretation methods after model training. In a decision tree-based solution, for instance, it is possible to use the increase in the Gini information throughout all splits to compute the feature importance for the entire set of features, obtaining a 0 to 100 score for each one.

Generally, feature importance is a technique that calculates a score for each feature given a machine learning model. This measure represents crucial importance to understanding the interaction among features inside a machine learning algorithm and also allows to interpret the model behavior. There are some methods in the literature to calculate feature importance. Usually, it is removed the supposedly informative features from the input and it is looked at how the classifier degrades. Some common methods are regularization, greedy search, averaged input gradient, permutation importance (Hooker et al., 2018; Wojtas and Chen, 2020; Altmann et al., 2010; Hastie et al., 2009). In this work, it was chosen the permutation importance. In a nutshell, that method calculates the increase or decrease in error when the values of a feature are permuted. If such permutation causes a greater variation in the error, this feature is important for the model. Using this approach, it is possible to correct biased measurements of feature importance and deal with high cardinality features (Altmann et al., 2010). The result presented in Fig. 6 shows the feature importance analysis by using the permutation method.

Another relevant method used to explain machine learning models is Shapley values, a method based on cooperative game theory in which a group of players come together to consume a service, and this incurs some cost. The Shapley value distributes this cost among the players. There is a correspondence between cost-sharing and the attribution problem: the cost function is analogous to the model, the players to base features, and the cost-shares to the attributions (Sundararajan and Najmi, 2020). The Shapey values are widely used in machine learning solutions due to the fact the method satisfies certain properties (Efficiency, Symmetry, Dummy and Linearity) and the solution is unique (Hart, 1989), i.e, for all trials the output for feature importance results in the same. Furthermore, the Shapey values is model-agnostic, i.e does not require any knowledge of the inner workings of the black box model, only access to the data and

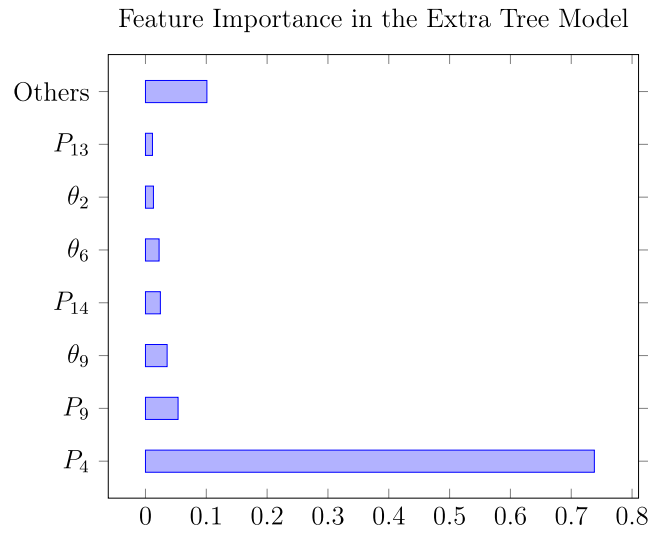

Fig. 6. Analysis of feature importance by permutation method.

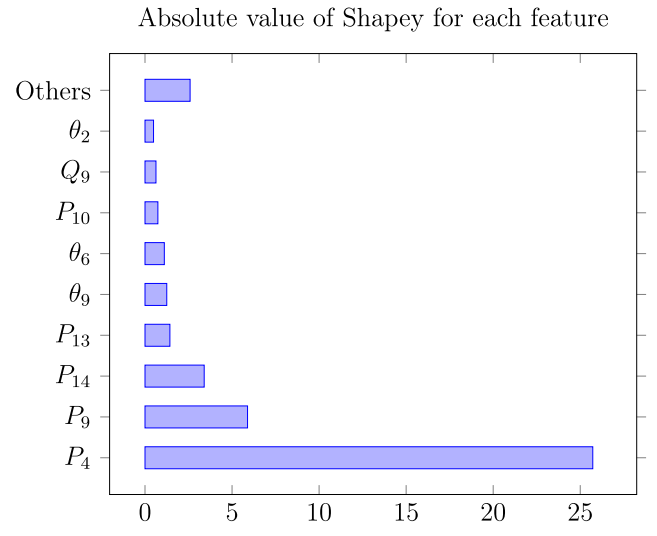

Fig. 7. Shapey values for the machine learning model.

the predictions of the model is necessary, in contrast to model-specific methods, which are applicable only to a single type of algorithm (Messalas et al., 2019). The result of Shapey values analysis for the data is presented in Fig. 7. Comparing Figs. 6 and 7, it is possible to observe a similar behavior in which the injection power in the studied bus (P_4) is the most relevant feature compared with others, such feature is followed by P_9 that is directed connected to the studied bus, the importance of the remaining features varies according to the employed method. However, it is important to emphasize the importance of buses 13 and 14 to obtain the load margin.

6.1. Sensitivity analysis

Utilizing the outcomes derived from Explainable AI techniques, it becomes feasible to conduct a sensitivity analysis on the features. This analysis takes into account the potential alteration of the target variable when each feature is perturbed.

The features consist of power injections and phasor measurements. For the power injections, the uncertainty can be represented by a Gaussian distribution with a standard deviation ranging from ±5% to ±10% of the mean values, as discussed

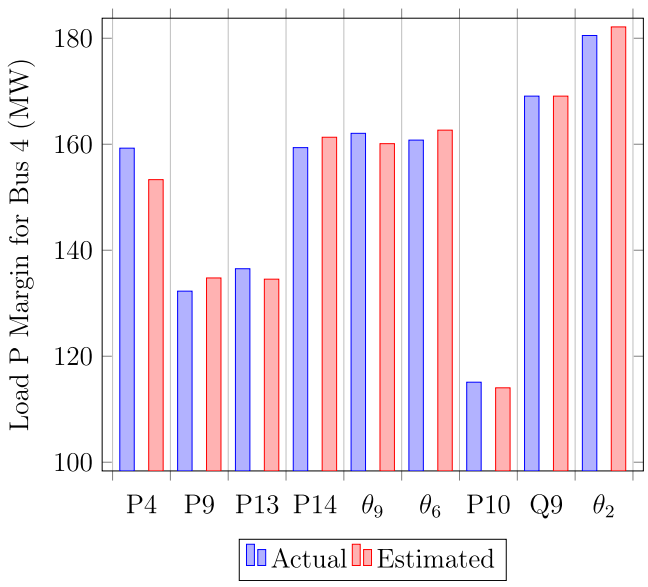

Fig. 8. Sensitivity analysis of the input variables.

Table 12
Performance of the Extra Tree regressor in the database considering only P_4.

Base	Algorithm	RMSE (MW)	MAPE (%)
Train	Extra Trees Regressor	10.0906	3.0052
Test	Extra Trees Regressor	28.3475	15.2106

in Singh et al. (2022). To account for the worst-case scenario, a value of $\sigma = 10\%$ was chosen. Regarding the phasor measurements, the uncertainty was modeled as the average value of the random error discussed in Section 5.6. The obtained result is depicted in Fig. 8. This figure illustrates the real and estimated values for the LPM of bus 4. It is noteworthy that the most significant variation was observed when introducing an inaccuracy of 30% in P4. In this case, the corresponding relative error was 3.73%. This analysis demonstrates the robustness of the proposed method against perturbations in the features .

7. Feature selection

The power P_4 is the most significant feature of the model, which lead us to think that it is possible to use only this feature to estimate the load margin. Nevertheless, Table 12 shows the lack of performance for the machine learning method using only such feature.

This way, in order to improve the performance of the method and at the same time reduce the dimension of the space generated by the features, it was proposed a method of feature selection based on the developed feature importance scheme. The common feature selection methods are based on correlation coefficient threshold or shuffling methods for a specific model. Here, it is proposed a method based on the result of permutation feature importance, in which a threshold is evaluated as a percentage of the mean feature importance presented in Fig. 6, i.e, considering the mean feature importance as $\bar{f}$, the best threshold wanted is a value $r \cdot \bar{f}$, in which the MAPE is minimum. In the proposed scheme, a feature with importance higher than the threshold is considered for the model, and the ones with a value less than the threshold are neglected. This procedure is done in the training database and after the result is calculated in the test database, which guarantees the feasibility of the method.

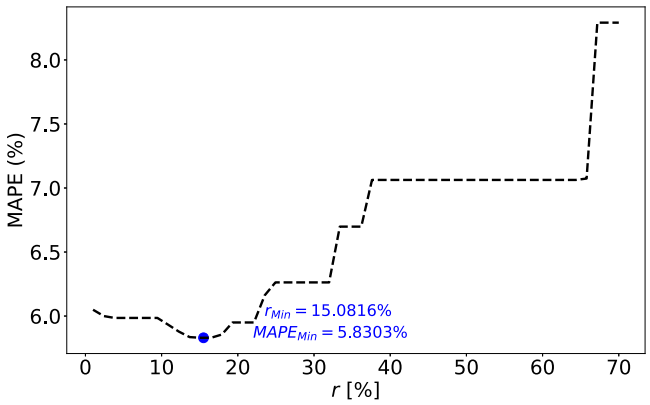

Fig. 9. Proposed feature selection method.

The Fig. 9 shows the result for the MAPE using several values of threshold. The best threshold obtained was $r = 0.150816$ resulting in a MAPE equal to 5.8303%. In that case, the features used for the model were $V_2, V_6, V_9, \theta_2, \theta_6, \theta_9, P_1, P_4, P_9, Q_9, P_{10}, P_{13}, P_{14}$. Therefore, the initial estimation uses 34 features and this number is reduced them to 13 features. Besides, the performance of the method with a feature selection scheme presented a better performance compared with the standard approach.

8. Conclusion

A new approach for load margin estimation is proposed in this paper. The proposed method is based on a machine learning solution considering different well-established algorithms, ensemble methods, and data transformation processes did not use in the technical literature before. Such method is tested in the IEEE 14-bars test system considering normal and contingency conditions for N-1 and N-2 grid elements. In addition, the database was built by using a random generation of load cases considering a load variation of ± 30 % which covers a wide range of loads in steady-state condition.

The number of features employed in the method was reduced compared with the common methods previously developed. Besides, the work avoided the use of power flow measurements due to the fact that these measurements depend on the grid parameters which might vary in time and need to be estimated to guarantee a more reliable power flow calculation.

It has been shown that it was possible to obtain a method with high performance without the use of power flow measurements using power injections and mainly phasor measurements of some buses. In this sense, the solution was developed considering an optimal PMU placement in some bars of the system which makes the solution more realistic than supposing PMUs are available in all buses.

Considering the state of the art for artificial intelligence solutions, the paper presented two methods of Explainable AI in order to improve the reliability and interpretability of the machine learning solution, allowing the operator to comprehend certain decisions. Furthermore, the feature importance scores were used to create a feature selection scheme that improved the performance of the standard method and reduced the space generated by the features.

Also, the influence of the noise in the estimation method was investigated considering different noise levels for the magnitude and phase. Concerning the noise, the solution presented a stable and predictable behavior with a acceptable performance even for extreme high noise levels do not observed in field applications.

Table A.13
Hyperparameters used in the Extra-Tree Regressor.

Parameter	Value
Number of estimators	150
Maximum depth	12
Maximum features per split	$\sqrt{\text{number of features}}$
Minimum samples per leaf	20
Minimum samples per split	15
Random state	42

CRediT authorship contribution statement

Felipe Proença de Albuquerque: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing – original draft, Writing – review & editing. **Rafael Nascimento:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing – review & editing. **Luisa H.B. Liboni:** Conception and design of study, Acquisition of data, Writing – original draft, Writing – review & editing. **Ronaldo F. Ribeiro Pereira:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing – original draft, Writing – review & editing. **Eduardo Coelho Marques da Costa:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing – original draft, Writing – review & editing.

Declaration of competing interest

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

Data availability

Data will be made available on request

Acknowledgments

CAPES - *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior*, Brazil.

CNPq - National Council for Scientific and Technological Development, Brazil (Grant 402830/2021-0).

FAPESP - São Paulo Research Foundation, Brazil (Grant 2021/01325-7). All authors approved the version of the manuscript to be published.

Appendix. Hyperparameters used in the Extra-Tree regressor

The following hyperparameters was used to build the machine learning model. It is worth emphasize that the parameters do not show in Table A.13 was considered as the default value given by the *scikit-learn* library.

References

Aazami, R., Aflaki, K., Haghifam, M.R., 2011. A demand response based solution for LMP management in power markets. Int. J. Electr. Power Energy Syst. 33 (5), 1125–1132.

Ajjarapu, V., Christy, C., 1992. The continuation power flow: a tool for steady state voltage stability analysis. IEEE Trans. Power Syst. 7 (1), 416–423.

Al-Alawi, A., Islam, S., 2004. Demand side management for remote area power supply systems incorporating solar irradiance model. Renew. Energy 29 (13), 2027–2036.

Ali, M., 2020. PyCaret: An open source, low-code machine learning library in Python, pyCaret version 1.0.0. https://www.pycaret.org.

Altmann, A., Toloşi, L., Sander, O., Lengauer, T., 2010. Permutation importance: a corrected feature importance measure. Bioinformatics 26 (10), 1340–1347.

Bahmanyar, A., Karami, A., 2014. Power system voltage stability monitoring using artificial neural networks with a reduced set of inputs. Int. J. Electr. Power Energy Syst. 58, 246–256.
Bento, M.E., 2022. Monitoring of the power system load margin based on a machine learning technique. Electr. Eng. 104 (1), 249–258.
Berk, R.A., 2004. Regression Analysis: A Constructive Critique, Vol. 11. Sage.
Brown, M., Biswal, M., Brahma, S., Ranade, S.J., Cao, H., 2016. Characterizing and quantifying noise in PMU data. In: 2016 IEEE Power and Energy Society General Meeting. PESGM, IEEE, pp. 1–5.
Bulat, H., Franković, D., Vlahinić, S., 2021. Enhanced contingency analysis—a power system operator tool. Energies 14 (4), 923.
Burman, P., 1989. A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. Biometrika 76 (3), 503–514.
Chandra, A., Pradhan, A.K., Sinha, A.K., 2016. A comparative study of voltage stability indices used for power system operation. In: 2016 21st Century Energy Needs-Materials, Systems and Applications. ICTFCEN, IEEE, pp. 1–4.
Daoud, J.I., 2017. Multicollinearity and regression analysis. In: Journal of Physics: Conference Series, Vol. 949. IOP Publishing, 012009.
Dasgupta, K., Soman, S., 2013. Line parameter estimation using phasor measurements by the total least squares approach. In: 2013 IEEE Power & Energy Society General Meeting. IEEE, pp. 1–5.
De Albuquerque, F.P., Da Costa, E.C.M., Pereira, R.F.R., Liboni, L.H.B., De Oliveira, M.C., 2021. Nonlinear analysis on transmission line parameters estimation from noisy phasorial measurements. IEEE Access 10, 1720–1730.
Dharmapala, K.D., Rajapakse, A., Narendra, K., Zhang, Y., 2020. Machine learning based real-time monitoring of long-term voltage stability using voltage stability indices. IEEE Access 8, 222544–222555.
Dobakhshari, A.S., Abdolmaleki, M., Terzija, V., Azizi, S., 2020. Online non-iterative estimation of transmission line and transformer parameters by scada data. IEEE Trans. Power Syst. 36 (3), 2632–2641.
Du, Y., Liao, Y., 2012. On-line estimation of transmission line parameters, temperature and sag using PMU measurements. Electr. Power Syst. Res. 93, 39–45.
Freris, L., Sasson, A., 1968. Investigation of the load-flow problem. In: Proceedings of the Institution of Electrical Engineers, Vol. 115. IET, pp. 1459–1470.
Geurts, P., Ernst, D., Wehenkel, L., 2006. Extremely randomized trees. Mach. Learn. 63 (1), 3–42.
Goh, H., Chua, Q., Lee, S., Kok, B., Goh, K., Teo, K., 2015. Evaluation for voltage stability indices in power system using artificial neural network. Procedia Eng. 118, 1127–1136.
Grinsztajn, L., Oyallon, E., Varoquaux, G., 2022. Why do tree-based models still outperform deep learning on typical tabular data? Adv. Neural Inf. Process. Syst. 35, 507–520.
Hajian, M., Ranjbar, A.M., Amraee, T., Shirani, A.R., 2007. Optimal placement of phasor measurement units: Particle swarm optimization approach. In: 2007 International Conference on Intelligent Systems Applications to Power Systems. pp. 1–6. http://dx.doi.org/10.1109/ISAP.2007.4441610.
Hart, S., 1989. Shapley value. In: Game Theory. Springer, pp. 210–216.
Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Vol. 2. Springer.
Hooker, S., Erhan, D., Kindermans, P.-J., Kim, B., 2018. Evaluating feature importance estimates. In: NeurIps.
John, V., Liu, Z., Guo, C., Mita, S., Kidono, K., 2015. Real-time lane estimation using deep features and extra trees regression. In: Image and Video Technology. Springer, pp. 721–733.
Joshi, P.M., Verma, H., 2021. Synchrophasor measurement applications and optimal PMU placement: A review. Electr. Power Syst. Res. 199, 107428.
Kalogirou, S.A., 2000. Applications of artificial neural-networks for energy systems. Appl. Energy 67 (1), 17–35. http://dx.doi.org/10.1016/S0306-2619(00)00005-2.
Kumar, S., Tyagi, B., Kumar, V., Chohan, S., 2021. PMU based voltage stability measurement under contingency using ANN. IEEE Trans. Instrum. Meas.
Kundur, P., Paserba, J., Ajjarapu, V., Andersson, G., Bose, A., Canizares, C., Hatziargyriou, N., Hill, D., Stankovic, A., Taylor, C., Van Cutsem, T., Vittal, V., 2004. Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definitions. IEEE Trans. Power Syst. 19 (3), 1387–1401. http://dx.doi.org/10.1109/TPWRS.2004.825981.
Kusic, G., Garrison, D., 2004. Measurement of transmission line parameters from scada data. In: IEEE PES Power Systems Conference and Exposition, Vol. 2004. IEEE, pp. 440–445.
Li, S., Ajjarapu, V., 2017. Real-time monitoring of long-term voltage stability via convolutional neural network. In: 2017 IEEE Power & Energy Society General Meeting. IEEE, pp. 1–5.
Li, C., Zhang, Y., Zhang, H., Wu, Q., Terzija, V., 2017. Measurement-based transmission line parameter estimation with adaptive data selection scheme. IEEE Trans. Smart Grid 9 (6), 5764–5773.
Löf, P.-A., Hill, D.J., Arnborg, S., Andersson, G., 1993. On the analysis of long-term voltage stability. Int. J. Electr. Power Energy Syst. 15 (4), 229–237.
Malbasa, V., Zheng, C., Chen, P.-C., Popovic, T., Kezunovic, M., 2017. Voltage stability prediction using active machine learning. IEEE Trans. Smart Grid 8 (6), 3117–3124.
Messalas, A., Kanellopoulos, Y., Makris, C., 2019. Model-agnostic interpretability with Shapley values. In: 2019 10th International Conference on Information, Intelligence, Systems and Applications. IISA, IEEE, pp. 1–7.
Milojević, V., Čalija, S., Rietveld, G., Ačanski, M.V., Colangelo, D., 2018. Utilization of PMU measurements for three-phase line parameter estimation in power systems. IEEE Trans. Instrum. Meas. 67 (10), 2453–2462.
Mohammadi, H., Dehghani, M., 2015. PMU based voltage security assessment of power systems exploiting principal component analysis and decision trees. Int. J. Electr. Power Energy Syst. 64, 655–663.
Nirbhavane, P.S., Corson, L., Rizvi, S.M.H., Srivastava, A.K., 2021. TPCPF: Three-phase continuation power flow tool for voltage stability assessment of distribution networks with distributed energy resources. IEEE Trans. Ind. Appl. 57 (5), 5425–5436.
Pegoraro, P.A., Brady, K., Castello, P., Muscas, C., von Meier, A., 2019. Compensation of systematic measurement errors in a PMU-based monitoring system for electric distribution grids. IEEE Trans. Instrum. Meas. 68 (10), 3871–3882.
Pourbagher, R., Derakhshandeh, S.Y., Golshan, M.E.H., 2022. An adaptive multi-step Levenberg–Marquardt continuation power flow method for voltage stability assessment in the ill-conditioned power systems. Int. J. Electr. Power Energy Syst. 134, 107425.
Shanker, M., Hu, M.Y., Hung, M.S., 1996. Effect of data standardization on neural network training. Omega 24 (4), 385–397.
Singh, V., Moger, T., Jena, D., 2022. Uncertainty handling techniques in power systems: A critical review. Electr. Power Syst. Res. 203, 107633.
Singh, D., Singh, B., 2020. Investigating the impact of data normalization on classification performance. Appl. Soft Comput. 97, 105524.
Sode-Yome, A., Mithulananthan, N., Lee, K.Y., 2006. A maximum loading margin method for static voltage stability in power systems. IEEE Trans. Power Syst. 21 (2), 799–808.
Su, H.-Y., Liu, C.-W., 2015. Estimating the voltage stability margin using PMU measurements. IEEE Trans. Power Syst. 31 (4), 3221–3229.
Suganyadevi, M., Babulal, C., 2014. Support vector regression model for the prediction of loadability margin of a power system. Appl. Soft Comput. 24, 304–315.
Sundararajan, M., Najmi, A., 2020. The many shapley values for model explanation. In: International Conference on Machine Learning. PMLR, pp. 9269–9278.
Thasnas, N., Siritaratiwat, A., 2019. Implementation of static line voltage stability indices for improved static voltage stability margin. J. Electr. Comput. Eng. 2019.
Van Cutsem, T., Vournas, C., 2007. Voltage Stability of Electric Power Systems. Springer Science & Business Media.
Villa-Acevedo, W.M., López-Lezama, J.M., Colomé, D.G., 2020. Voltage stability margin index estimation using a hybrid kernel extreme learning machine approach. Energies 13 (4), 857.
Wang, C., Miller, C.J., Nehrir, M.H., Sheppard, J.W., McElmurry, S.P., 2015a. A load profile management integrated power dispatch using a Newton-like particle swarm optimization method. Sustain. Comput. Inform. Syst. 8, 8–17.
Wang, Y., Xu, W., Shen, J., 2015b. Online tracking of transmission-line parameters using scada data. IEEE Trans. Power Deliv. 31 (2), 674–682.
Wehenkel, A., Mukhopadhyay, A., Le Boudec, J.-Y., Paolone, M., 2020. Parameter estimation of three-phase untransposed short transmission lines from synchrophasor measurements. IEEE Trans. Instrum. Meas.
Wojtas, M., Chen, K., 2020. Feature importance ranking for deep learning. Adv. Neural Inf. Process. Syst. 33, 5105–5114.
Yuill, W., Edwards, A., Chowdhury, S., Chowdhury, S., 2011. Optimal PMU placement: A comprehensive literature review. In: 2011 IEEE Power and Energy Society General Meeting. IEEE, pp. 1–8.
Zhou, D.Q., Annakkage, U.D., Rajapakse, A.D., 2010. Online monitoring of voltage stability margin using an artificial neural network. IEEE Trans. Power Syst. 25 (3), 1566–1574.
Zimmerman, R.D., Murillo-Sánchez, C.E., Gan, D., 1997. Matpower: A Matlab Power System Simulation Package, Vol. 1. Power Systems Engineering Research Center, Ithaca NY, Manual.
Zimmerman, R.D., Murillo-Sánchez, C.E., Thomas, R.J., 2011. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. IEEE Trans. Power Syst. 26 (1), 12–19. http://dx.doi.org/10.1109/TPWRS.2010.2051168.