21, rue d'Artois, F-75008 PARIS
http : //www.cigre.org

CIGRE US National Committee
2022 Grid of the Future Symposium

Proof-of-Concept Studies in Machine Learning based CVR Calculations

Z. HOSSEINI, A. KHODAEI
PLUG LLC
USA

W. FAN, A. HAQUE, P. PABST
Commonwealth Edison Company
USA

M. MAHOOR
Independent Researcher
USA

SUMMARY

Electric utilities implement Conservation Voltage Reduction (CVR) to reduce energy consumption and peak demand by lowering the voltage at the distribution system. This is a cost-effective way to improve system energy efficiency and to provide benefits to their customers. Identifying the impact of CVR is carried out through Measurement and Verification (M&V), which determines the CVR factor, energy baseline, and energy savings. The commonly used M&V methodologies, including regression, comparison, and load modeling-based M&V, are data-intensive and depend heavily on the quality of the data. This paper leverages machine learning based techniques to develop an outside-the-box methodology for CVR M&V that can quantify CVR impacts using only feeder characteristics. The machine learning based models used are Artificial Neural Networks (ANN), Support Vector Regression (SVR) and K-Nearest Neighbors (KNN). The performance of the proposed models is demonstrated through actual CVR data from a utility in the US. For the case study presented, Mean Absolute Percentage Error (MAPE) for the baseline energy forecast is around 10% and for the energy savings calculation is around 14%.

KEYWORDS

Conservation voltage reduction, machine learning, energy saving

Ashraful.Haque@comed.com

1. INTRODUCTION

The unparalleled transformation of the electric power grid to a more digital, decentralized, and decarbonized grid is supported heavily by comprehensive grid monitoring and control deployments. The enhanced monitoring and control facilitate multiple viable practices, most notably Volt-Var Optimization (VVO) and Conservation Voltage Reduction (CVR). CVR is the act of reducing energy consumption and peak demand through lowering the voltage at the distribution grid and following the ranges provided by the ANSI C84.1 standard [1], [2] or voltage limits identified by the utilities' regulatory bodies. Under CVR, the voltage across a distribution feeder operates toward the lower range of regulated voltages by coordinating feeder equipment, including the Load Tap Changers (LTC), voltage regulators, and capacitor banks.

CVR has gained renewed interest in recent years as a cost-effective way to deliver energy efficiency benefits to customers with no negative impacts on customers' appliances and without requiring them to opt into the program [3], [4]. A significant challenge in CVR application remains the measurement and verification (M&V) of its effects. The energy savings benefits of CVR are identified by M&V in order to study the benefit-cost ratio or to report energy savings to the public utility commissions. However, a lack of benchmark load measurement complicates assessing CVR's real impact [5]. This is further complicated by the challenges in differentiating load and energy reductions due to variations in temperature and customer usage patterns over time.

Based on the benchmarking report in [3], utilities primarily leverage three CVR assessment methods: comparison-based, regression-based, and simulation-based. The comparison-based methods leverage operational data under CVR (treatment) and non-CVR (control) conditions and accordingly determine the CVR factor by comparing these two cases. There are two general categories for comparison-based methods, including correlated-feeder and correlated-weather. The primary benefit of comparison-based methods is being easy to implement. However, on the downside, other factors, such as the weather differences in the correlated-weather approach or load differences in the correlated-feeder approach, may add noise to the measurements and result in erroneous CVR calculations. Given the commonly small CVR effect, this noise may completely mask the CVR. Moreover, the time-dependency nature of the CVR factor may be lost as the data is averaged.

Regression-based methods model load and nodal voltage as a function of various factors, including temperature and CVR impact. The CVR factor is calculated by generating this function using data associated with CVR-on and CVR-off conditions. In regression-based methods, physical interpretations are potentially embedded in the regression models so electric utilities can understand the model behavior based on impact factors. A benefit of regression-based methods is that physical interpretations are potentially embedded in the regression models. As a disadvantage, regression models may have higher estimation errors than the CVR effects, thus masking the effect. Moreover, these models are primarily linear, while the loads are known to be nonlinear.

Simulation-based methods simulate the load consumption in CVR-off conditions and further use this model within power flow calculations to find the difference with measured load consumption and accordingly calculated CVR factor. Simulation-based methods show high precision if the load models are highly accurate while allowing the system to run continually. However, building models for all existing and emerging load components may be cost prohibitive. The application of aggregated load models at the circuit level is a potential solution.

The developed models are further not adaptive to dynamic changes of feeders and load behaviors. In addition, the power flow model requires a load-voltage sensitivity (CVR factor) that is unknown until the analysis is complete. A constant CVR factor can be used for the power flow model, and by refining the CVR factor, energy savings would be calculated based on the new CVR factor [6].

This paper will conduct proof-of-concept studies of a machine learning-based solution capable of CVR M&V for a selected feeder using only feeder characteristics. This is an entirely new yet untested approach in CVR M&V. The rest of the paper is organized as follows. Section 2 presents the proposed model. Section 3 discusses the required preprocessing step. Section 4 elaborates on some of the most suitable machine learning approaches for the problem at hand. Section 5 conducts numerical simulation on a practical test system and identifies the best-performing approach. This approach is validated and updated in Section 6. Section 7 concludes the paper.

2. PROPOSED MACHINE LEARNING-BASED CVR M&V MODEL

Figure 1 shows a high-level outline of the proposed framework. Feeder characteristics are considered as inputs to the machine learning engine. In addition to feeder characteristics, the CVR schedule (e.g., 4-days-ON and 4-days OFF) is considered an input. The engine includes a sophisticated model that is trained and tested based on an existing data pool and identifies the weighted impact of each input on its outputs. The engine's outputs are CVR results, including the energy baseline and energy savings.

This would include assessment (for feeders that have active CVR) and projection (for those that have not yet been deployed with CVR). This would eliminate the need to collect significant amount of power and voltage data and conduct detailed CVR M&V analysis.

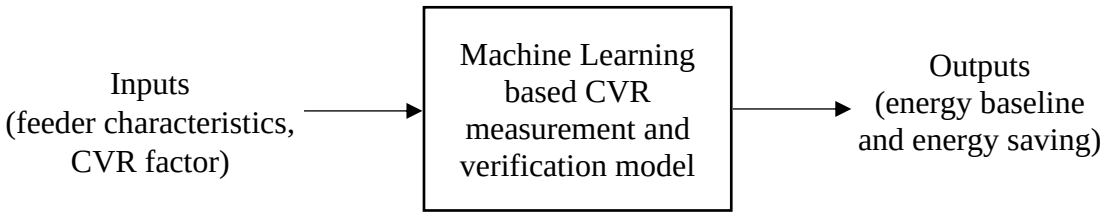

Figure 1: Overall outline of the proposed model.

The model inputs are listed in Table 1.

Table 1: Inputs of the Proposed Machine Learning based CVR M&V model

Input	Description
Average voltage (ON/OFF)	Average voltage ON and average voltage OFF are the average voltages over CVR-on time intervals and CVR-off time intervals, respectively. These two values contain useful information about the characteristics of the feeder.
Capacitor banks	Capacitor bank counts are the total number of capacitor banks on a feeder.
Customer count	Customer count represents the total number of customers in a feeder.
CVR factor	A fixed CVR factor of 0.8 is considered for all feeders.

DER capacity (kW)	DER capacity of a feeder shows the integrated Distributed Energy Resources (DER) size at that feeder.
Feeder length (Miles)	Feeder length shows the length of the feeder beginning from its associated substation and spanning through the branches ending to the customers. Feeder length could have a potential impact on a feeder's load and thus its energy savings.
Feeder nominal voltage (kV)	Feeder voltage in kV is the nominal voltage of the feeder. As the energy savings is related to the voltage reduction, nominal feeder voltage needs to be considered for its potential impact.
Geography/ZIP code	The geography of a feeder can be considered by its ZIP code. Weather-related characteristics of the area which the feeder locates in, as well as the socioeconomic characteristics of customers, are embedded in its ZIP code.
Load factor	The load factor is the average load divided by the peak load in a specified period. It represents the amount of energy that was used in a period versus the amount of energy that would have been used if the power had been left on during peak demand.
Pre-CVR peak load (amp)	Pre-CVR peak load is the highest occurrence of load before CVR implementation.
Power factor	The power factor is the ratio of active power to apparent power in a feeder. The power factor can impact the amount of active power and the CVR energy savings.
Rated load (amp)	Rated load is the maximum load that the feeder is designed for and can supply.
RCI mix (Percentage)	RCI mix represents the percentages of residential, commercial, and industrial customers in a feeder. A feeder's load and the impact of the feeder's voltage change on its load are directly influenced by the proportion of residential, commercial, and industrial customers.
Regulators	Regulator counts are the total number of regulators on a feeder.
Temperature	Temperature could potentially impact energy savings, however, an annualized number (e.g., average annual temperature) should be considered as an input. Also, the ZIP code may add weather characteristics to the model, so there should be considerations as not to duplicate this input.

A fixed CVR factor of 0.8 is considered for all feeders. In this paper, the training datasets were obtained based on the CVR factor of 0.8, following the approach set by the utility regulatory body [7]. The outputs are provided in Table 2.

Table 2: Outputs of the Proposed Machine Learning based CVR M&V model

Output	Description
Energy baseline (MWh)	Energy baseline is the total energy consumed by a feeder if CVR was not implemented.
Energy saving (MWh)	Energy saving is the energy saved by applying CVR to a feeder.

The inputs and outputs of the model will be engineered to extract useful information from raw data. For instance, for the ZIP code input, each feeder's socioeconomic and weather characteristics can be extracted from public resources. In addition, a set of normalization/standardization steps are needed to ensure the inputs are adequately readable by the machine learning model.

3. DATA PRE-PROCESSING

The preprocessing step includes standardization and feature selection. The standardization of datasets is a common requirement for many machine learning estimators. Input variables and features measured at different scales do not contribute equally to the model fitting and model function, which might create a bias in the model. If a feature has a variance with orders of magnitude larger than others, then that might dominate the objective function and make the regression unable to learn from other features correctly as needed. Therefore, standardization is used as the first step to deal with this potential problem. In this paper, standardization is performed feature-wise in an independent way. Standardized features are obtained by removing the mean (μ) and scaling to unit variance (σ), i.e., μ=0, σ=1. In other words, centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the data as in (1), where μ and σ are the mean and standard deviation of feature *x*, and *z* is the standardized feature.

$$z = \frac{x-\mu}{\sigma} \qquad (1)$$

The standardized data will consequently go through the feature selection process. There are various methods of feature selection [8], however, a feature selection based on the correlation metric is used in this paper. This is a univariate feature selection method that works by selecting the best features based on statistical tests and can be seen as a preprocessing step to an estimator. The correlation metric measures the linear relationship (Pearson's correlation) or monotonic relationship (Spearman's correlation) between two variables, X and Y.

The correlation coefficient is an essential measure of the relationship between two random variables. Once calculated, it describes the validity of a linear fit. For two random variables, X and Y, the correlation coefficient, ρ_{xy}, is calculated as follows by dividing the covariance of the two variables by the product of their standard deviations (2):

$$\rho_{xy} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \qquad (2)$$

Covariance serves to measure how much the two random variables vary together. The correlation coefficient would be a positive number if both variables consistently lie above the expected value and would be negative if one tends to lie above the anticipated value and the other to lie below.

The correlation coefficient will take on values from 1 to -1. Values 1 and -1 indicate perfect increasing and decreasing linear fits, respectively. As the population correlation coefficient approaches zero from either side, the strength of the linear fit diminishes. Approximate ranges representing the relative strength of the correlation are shown below. The ranges also apply for negative values between 0 and -1.

Feature selection using the correlation metric is based on an F-test that estimates the degree of linear dependency between two random variables. This is done in two steps: computing the cross-correlation between each regressor (X) and the target (Y), followed by a calculation of the F-score (or F-value). F-score is the result of a test in which the null hypothesis is that all of the regression coefficients are equal to zero. This means that the model has no predictive capability. The F-test compares the model with zero predictor variables (the intercept-only model) and decides whether the added coefficients improved the model. Significant results

denote that the newly added coefficients improved the model's fit. The F-score can be obtained as follows, where *n* is the number of observations.

$$F = \frac{corr^2 \times (n-1)}{(1-corr^2)} \quad (3)$$

4. MACHINE LEARNING MODEL DESIGN

The proposed problem is a system identification problem that aims to estimate energy baseline and energy savings for feeders with active CVR while predicting energy baseline and energy savings for feeders without CVR. Considering the type of the problem at hand, a regression method is used.

Regression is a supervised learning technique that helps in establishing a relationship between a dependent variable (output) and one or more independent variables (inputs) [9]. Traditional data-driven regression methods include but are not limited to Linear regression, Logistic regression, Ridge regression, Lasso regression, Polynomial regression, Quasi Poisson regression, and Stepwise regression. In addition to the mentioned methods, newer regression techniques are also available such as K-nearest neighbor (KNN), Artificial Neural Networks (ANN), Regression trees, Random Forests, and Support Vector Regression (SVR) which may lead to more accurate results. Authors in [10] present a machine learning based framework to forecast baseline loads through short-term electrical load forecasting using neural networks [11]. Authors in [11] present a methodology to quantify energy savings through SVR.

This paper uses three regression models: KNN, SVR, and ANN. The key purpose of using neural networks over linear regression is the fact that the linear regression learns the linear relationship between the features and the target and therefore sometimes cannot learn the complex nonlinear relationships between energy consumption and other factors i.e., temperature, geographical location, device status. ANN's layer based architecture are much more suitable to learn the complex relationship between the features. Following is a brief introduction to each of these three methods.

4.1.Artificial Neural Network

Neural networks, also known as ANNs or simulated neural networks, are a subset of machine learning tools. Their name and structure are inspired by the human brain, mimicking how biological neurons signal to one another. An ANN consists of simple input/output units or processing nodes that are interconnected, and each connection has a weight associated with it. The nets are organized into layers and data moves through these nodes. Figure 2 presents the basic architecture of an artificial neural network.

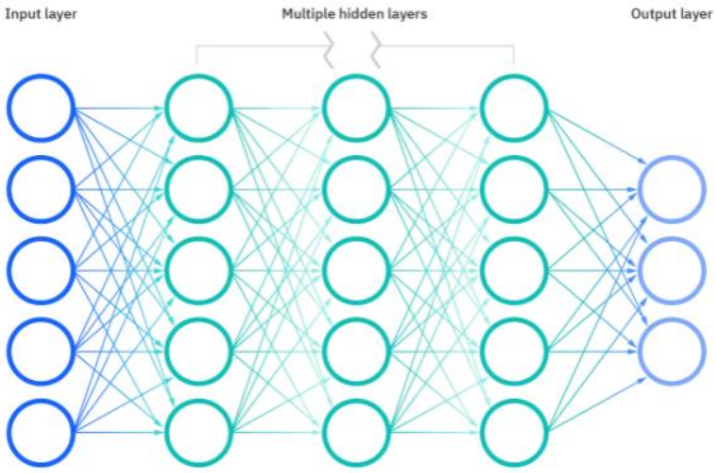

Figure 2: Architecture of an artificial neural network [12].

An ANN consists of the input, hidden, and output layers. Each layer consists of *n* neurons and an activation function associated with each neuron, which is responsible for introducing nonlinearity in the relationship. The data are supplied into the input layer, transformed in hidden layers, and scaled to the wanted outcome in the output layer. Each node receives the output from the previous node to which it is connected. The node is where computation occurs [13]. These are multiplied by weights, summed, and transformed with a function and passed on to the nodes of the next layer. Searching for the weights trains the network. The key functions of neural networks are input scoring, error calculating, and weight adjustment. The deeper the network, the more complex features the nodes recognize. This flexibility enables neural networks to handle large and high-dimensional datasets with numerous parameters and of nonlinear nature.

4.2.Support Vector Regression

Support Vector Machines (SVMs) are one of the most popular and widely used algorithms for classification problems in machine learning. The basic idea for classification is to maximize the margin between classes, yielding maximally robust classification. Based on SVM, a new regression technique can be defined. Support vector regression (SVR) is a supervised learning algorithm used to predict discrete values. SVR, using the same principle as the SVM, is the closest in form to traditional regression and brings concepts that make regression more robust and less prone to error. The basic idea behind SVR is to find the best-fitted line. In SVR, the best-fitted line is the hyperplane with the maximum number of points. SVR can be dealt with as an optimization problem based on finding a straight line called a hyperplane [14]. The hyperplane can be defined as in (4):

$$w.x + b = e \qquad (4)$$

where w and b are the SVM model parameters, and x is the input training data. SVR, unlike other regression models that try to minimize the error between the real and predicted values, attempts to fit the best line within a threshold value (i.e., the distance between the hyperplane and boundary line). Thus, the error term is instead handled in the constraints, where the absolute error is set to be less than or equal to a specified margin, called the maximum error. By tuning the maximum error, the model's accuracy can be improved. In other words, the SVR model tries to satisfy the condition in (5):

$$-a < y - (wx + b) < a \qquad (5)$$

This model is shown in Figure 3. The hyperplane is calculated using the model in (6).

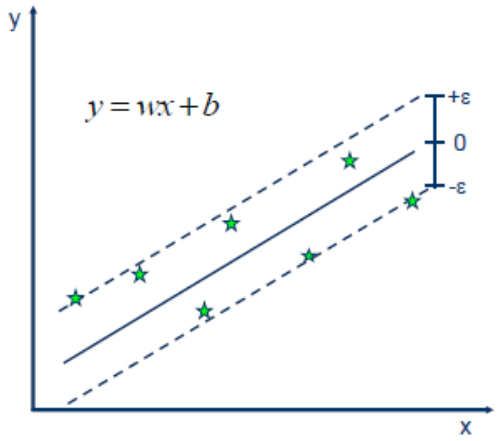

Figure 3: SVR model [15].

$$\min \quad \frac{1}{2}\|w\|^2 \qquad s.t. \quad |y_i - w_i x_i| \leq \varepsilon \qquad (6)$$

Although the original SVR algorithm is a linear regression, by applying a kernel function to the algorithm, nonlinear SVR can be achieved. A kernel function, which is a set of mathematical functions that takes data as input and transforms it into the required form, transforms the input vector into a higher-dimension space. There are several kernel functions used in SVM, including but not limited to linear polynomial, and radial basis function (RBF) [16].

4.3.K-Nearest Neighbors Regression

K-Nearest Neighbors (KNN) regression is a non-parametric method that intuitively approximates the association between independent variables and the continuous outcome. KNN regression predicts the numerical target based on a similarity measure, e.g., distance functions. A simple KNN regression can be achieve by averaging the numerical target of the K nearest neighbors. Another approach is to use an inverse distance weighted average of the K nearest neighbors. The size of neighborhoods, i.e., the optimal value for K, can be determined by data inspection or cross-validation technique. KNN regression uses the same distance functions as KNN classification. Various distance functions can be used for continuous variables such as Euclidean, Manhattan, and Minkowski [17].

The KNN model stores all the training data set. When a new test data point is given, the aim is to find K points from the training set closest to the test data point based on the selected distance formulation. It is then tried to find the points which minimize the distance metric.

5. SIMULATION RESULTS

Simulations are performed using 380 feeders from a utility in the US. The dataset is created from a regression-based approach using deemed CVR factor. 80% of this dataset is used for

training while the rest is reserved for testing. Four performance metrics are used to evaluate the developed model, including R^2, Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Table 3 summarizes these metrics for model evaluation:

Table 3: Selected metrics for model evaluation

Metric	Formula	Range	Interpretation
R^2	$R^2 = 1 - \frac{SSE}{SST}$	0 to 1	Value close to 1 indicates better performance.
MSE	$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$	0 to ∞	Small value indicates better performance.
MAE	$MAE = \frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}_i\rvert$	0 to ∞	Small value indicates better performance.
MAPE	$MAPE = \frac{100\%}{N}\sum_{i=1}^{N}\left\lvert\frac{y_i - \hat{y}_i}{y_i}\right\rvert$	0% to 100%	Small percentage value indicates better performance.

R^2, also known as the coefficient of determination, measures how much the model can explain variability in a dependent variable. R^2 is the squared of the correlation coefficient (R) and is calculated as shown in Table 3, where SSE stands for sum square of errors (the squared difference between the predicted value and the actual value), and SST is sum square of the difference between the actual value and the mean of the actual value.

While R^2 may be used as a relative measure of how well the model fits dependent variables, MSE, MAE and MAPE will be considered as measures of the goodness for the fit. Figures 4 and 5 show the F-value of input features with respect to energy baseline and energy saving, respectively.

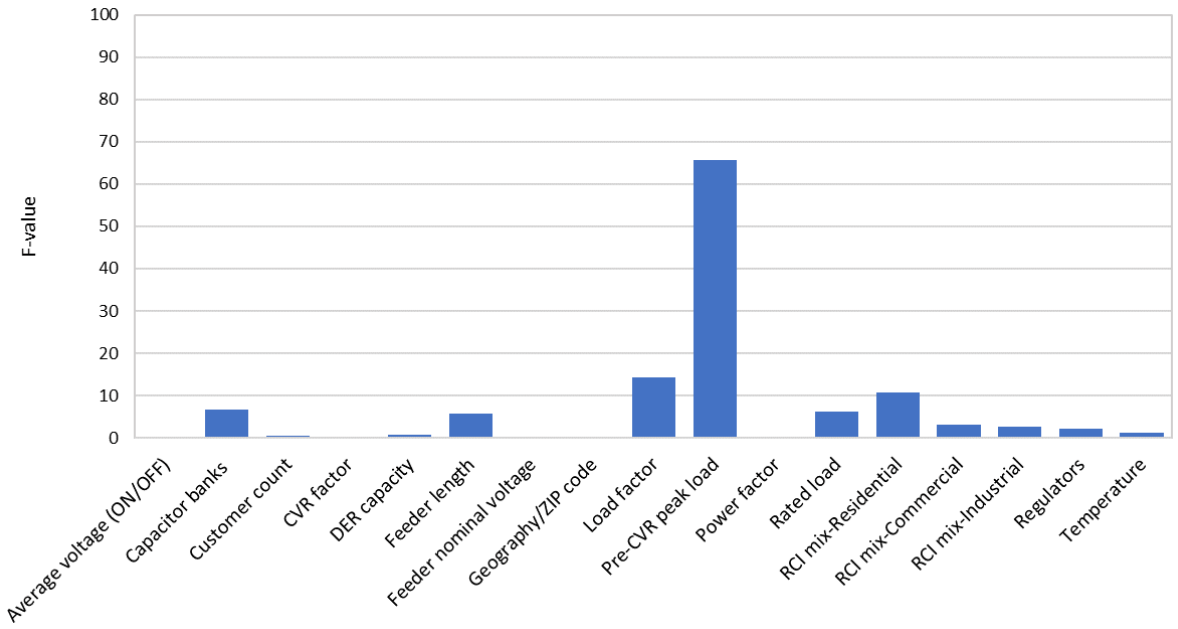

Figure 4: F-value of input features and energy baseline.

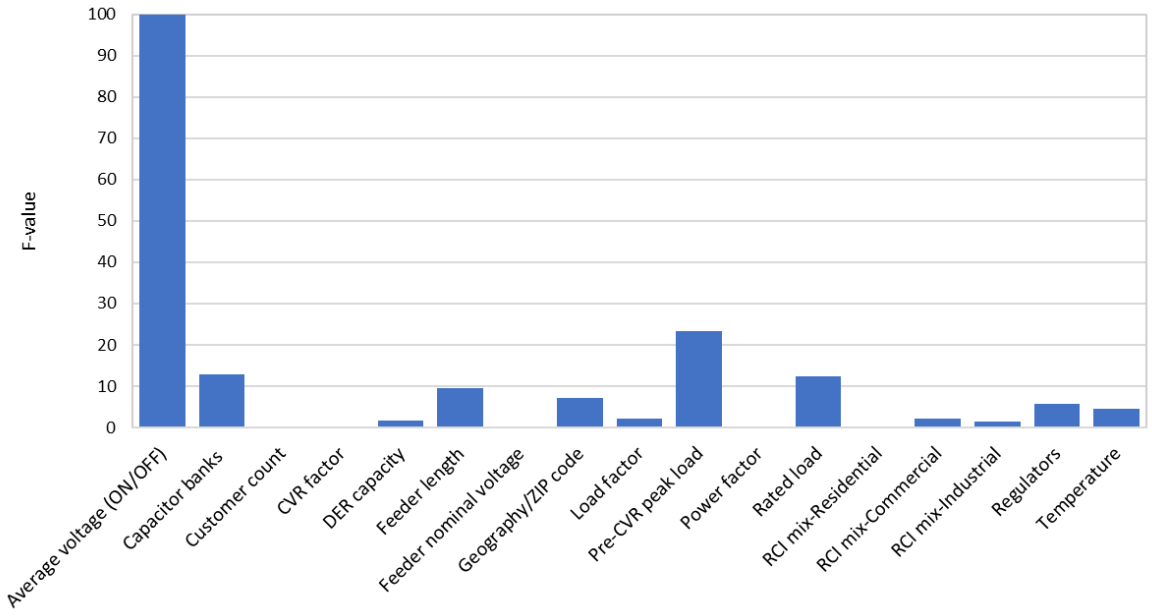

Figure 5: F-value of input features and energy savings.

As these two figures suggest, features which are constant or almost constant among all feeders show zero F-values. These features add no useful details about the input-output relationship, which means we can remove them from the input data. Based on these results, CVR factor, Power Factor, Feeder Nominal Voltage, and DER Capacity are the features which are eliminated from the input data. In addition, some inputs may have more impact on one of the outputs than the other. For example, the Average Voltage has a small F-value with respect to energy baseline, showing its low impact in energy baseline estimations, while presenting a large F-value for energy saving. Following data pre-processing, an initial configuration for the regression models, as shown in Table 4, is considered. The models are accordingly trained on the available dataset.

Table 4: Regression models' initial configuration.

Model	Configuration				
KNN	No. of neighbors = 4	Distance function = Euclidean			
SVR	Kernel = rbf	C = 100	Epsilon = 0.1		
ANN 1	No. of hidden layers = 5	No. of neurons in each layer = 50	Activation functions = relu & swish	Optimizer = Adam	Loss function = MSE
ANN 2	No. of hidden layers = 2	No. of neurons in each layer = 50	Activation functions = swish	Optimizer = Adam	Loss function = MAE

To evaluate the performance of the designed models, R^2, MSE, MAE, and MAPE metrics are utilized. Tables 5 and 6 show the results of these regressions by using all input features listed in Table 4 for energy baseline and energy savings, respectively.

Table 5: Model accuracy for energy baseline estimation.

	Energy Baseline			
	R^2	**MSE**	**MAE**	**MAPE (%)**

KNN	0.46	0.52	0.55	16.90
SVR	0.24	0.72	0.62	18.51
ANN 1	0.26	0.67	0.60	16.68
ANN 2	0.54	0.42	0.47	13.56

Table 6: Model accuracy for energy savings estimation.

	Energy Savings			
	R^2	**MSE**	**MAE**	**MAPE (%)**
KNN	0.61	0.37	0.49	26.32
SVR	0.50	0.45	0.50	25.00
ANN 1	0.58	0.38	0.42	18.53
ANN 2	0.73	0.24	0.36	16.71

The obtained results demonstrate that ANN can be a good candidate for the problem at hand, considering better performance for both energy baseline and energy savings calculations. However, the results for ANN1 and ANN2 suggest that the accuracy highly depends on the ANN design and its parameter tuning. The next section uses the ANN model for further analysis, validation, and potential upgrade.

6. IMPROVING MODEL PERFORMANCE

The ANN model is used in this section to ensure the standard performance metrics are satisfied. Additional parameter tuning for this model is performed to enhance the model performance. Hyperparameter tuning procedures are utilized using *k*-fold (*k*=6) to automatically find the optimal values for the hyperparameters that lead to the best model performance. It should be noted that in the following studies, all the input features are used. In addition, *k* is set to be 6 for hyperparameter tuning processes to ensure a thorough hyperparameter search for the models.

The regression model consists of two sets of parameters. The first set, called parameters, are the model components learned during the training process and cannot be manually set. A model starts the training process with random parameter values and adjusts them throughout the training procedure. The second set, known as hyperparameters, needs to be set before the model training. Hyperparameters define the model configuration. Adjusting the hyperparameters may improve the model's accuracy, thus in this section, hyperparameter tuning is presented for the ANN model.

To create the optimal ANN model, the first step in hyperparameter tuning is to find the optimal number of hidden layers in the network. In this regard, a set of sensitivity analyses are studied with respect to the number of hidden layers. In other words, the number of hidden layers in the neural network is increased from 1 to 10. MAPE and computation time of each simulation run is shown in Figure 6. The number of neurons in each layer is set at 500.

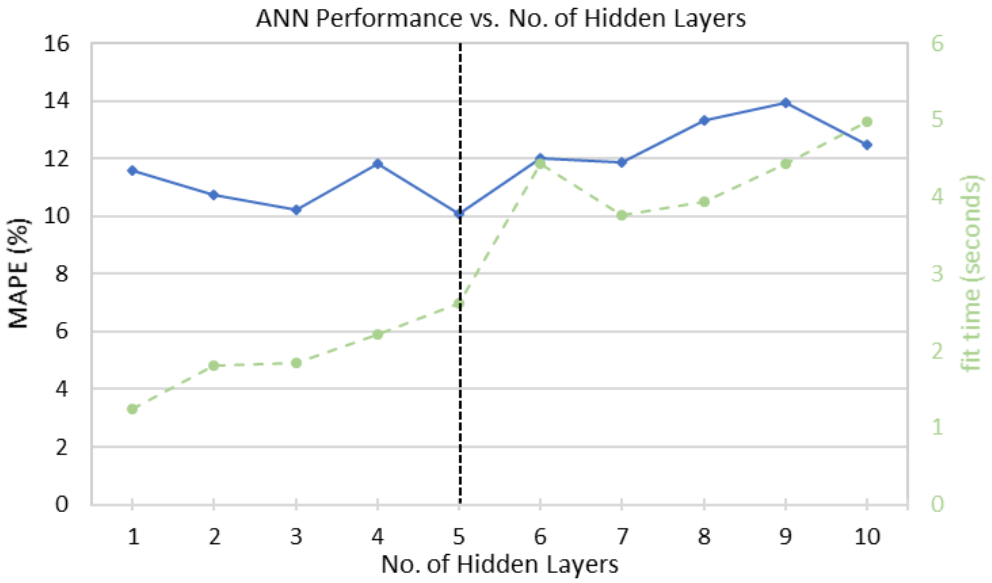

Figure 6: MAPE and computation time of energy baseline estimation using the ANN model with respect to the number of hidden layers in the network.

As Figure 6 suggests, the optimal number of hidden layers is 5, which has the lowest MAPE and an acceptable computation time. Now that the optimal number of hidden layers is determined, other hyperparameters of the ANN are tuned by comparing various possible values. In total, 10,000 candidates are defined, and six folds for each candidate is performed, meaning that 60,000 fits have been studied. Table 7 shows the optimal hyperparameter candidate for the ANN model.

Table 7: Optimal configuration for ANN model.

Hyperparameter	Optimal Configuration	
	Energy Baseline	**Energy Savings**
No. Neurons in Layer 1	500	500
No. Neurons in Layer 2	500	500
No. Neurons in Layer 3	500	500
No. Neurons in Layer 4	100	500
No. Neurons in Layer 5	100	100
Activation Function in Layer 1	swish	swish
Activation Function in Layer 2	swish	swish
Activation Function in Layer 3	swish	swish
Activation Function in Layer 4	swish	swish
Activation Function in Layer 5	swish	swish
Optimizer	Adam	Adam
Batch_size	200	200
epochs	20	20
Loss Function	MAE	MSE

Using the optimal hyperparameters, the performance of the ANN model is studied against the data pool. The training and testing are repeated six times for different selections of training and testing datasets, and the accuracy is calculated by averaging the accuracies of the six folds. The performances for estimating energy baseline and energy savings are tabulated in Tables 8 and 9, respectively.

Table 8: Accuracy of the updated ANN model in estimating energy baseline.

	Energy Baseline			
	R^2	**MSE**	**MAE**	**MAPE (%)**
ANN	0.81	0.19	0.31	10.76

Table 9: Accuracy of the updated ANN model in estimating energy savings.

	Energy Savings			
	R^2	**MSE**	**MAE**	**MAPE (%)**
ANN	0.88	0.12	0.24	13.92

Comparing these results with those in Tables 8 and 9 proves that hyperparameter tuning can help significantly improve the accuracy of the ANN model and achieve MAPE values of close to 10%.

7. CONCLUSIONS

This paper proposed and developed a machine learning-based methodology for CVR M&V. The proposed methodology is capable of finding CVR-enabled energy savings using only feeder characteristics. Historical CVR data for 380 feeders from a utility were used to train, test, and validate this methodology. Simulations showed that an artificial neural network model could outperform other approaches, however it needs careful network design and parameter tuning. The proposed methodology in this paper can be used by electric utilities to assess the impact of deployed CVR programs, and further develop projections for future CVR projects. This all can be done with limited reliance on data collected from the field., thus removing the concerns associated with data quality and availability.

BIBLIOGRAPHY

[1] M.S., Chen, R., Shoults, J., Fitzer, and H., Songster. "The effects of reduced voltages on the efficiency of electric loads" (IEEE Transactions on Power Apparatus and Systems, vol. 7, pp. 2158-2166, 1982).

[2] A., Faruqui, K., Arritt, S., and Sergici. "The impact of AMI-enabled conservation voltage reduction on energy consumption and peak demand" (The Electricity Journal, vol. 30, no. 2, pp. 60-65, 2017).

[3] Z. S. Hosseini, A. Khodaei, W. Fan, M. Hossan, H. Zheng, S. A. Fard, A. Paaso, and S. Bahramirad. "Conservation Voltage Reduction and Volt-VAR Optimization: Measurement and Verification Benchmarking" (IEEE Access, vol. 8, pp. 50755-50770, March 2020_.

[4] ANSI. "ANSI Standard C84.1-2016 Electric Power Systems and Equipment Voltage Ratings (60 Hz)," (2016).

[5] Z. S. Hosseini, M. Mahoor, A. Khodaei, M. Hossan, W. Fan, and P. Pabst. "Understanding the Impact of Data Anomalies on Conservation Voltage Reduction Measurement and Verification – A Practical Study" (IEEE/PES Transmission and Distribution Conference and Exposition (T&D), New Orleans, LA, April 2022).

[6] Z. S. Hosseini, M. Mahoor, A. Khodaei, M. Hossan, W. Fan, P. Pabst, and A. Paaso, "Sensitivity Analyses of CVR Measurement and Verification Methodologies to Data Availability and Quality" (IEEE Access, vol. 9, pp. 157203-157214, Nov. 2021).

[7] IL Statewide Technical Reference Manual Version 10.0. Available [online:] https://www.ilsag.info/technical-reference-manual/il-statewide-technical-reference-manual-version-10-0/

[8] A. Jović, K. Brkić, and N. Bogunović. "A review of feature selection methods with applications," (38th international convention on information and communication technology, electronics and microelectronics (MIPRO), pp. 1200-1205, 2015).

[9] T. P. Ryan. "Modern regression methods" (Vol. 655. John Wiley & Sons, 2008).

[10] A. Haque, S. Rahman. "Short-term electrical load forecasting through heuristic configuration of regularized deep neural network" (Applied Soft Computing,Volume 122,2022,108877,ISSN 1568-4946,https://doi.org/10.1016/j.asoc.2022.108877).

[11] A. Haque et al. "An SVR-based Building-level Load Forecasting Method Considering Impact of HVAC Set Points," (2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), 2019, pp. 1-5, doi: 10.1109/ISGT.2019.8791649).

[12] IBM. Available [online:] https://www.ibm.com/cloud/learn/neural-networks.

[13] S. Wang. "Artificial neural network" (Interdisciplinary computing in java programming, pp. 81-100. Springer, Boston, MA, 2003).

[14] A. J. Smola, and B. Schölkopf. "A tutorial on support vector regression" (Statistics and computing 14, no. 3, pp. 199-222, 2004).

[15] S. Sayad. "Support Vector Machine – Regression (SVR)," Available [online:] https://www.saedsayad.com/support_vector_machine_reg.htm.

[16] A. Patle and D. S. Chouhan. "SVM kernel functions for classification" (International Conference on Advances in Technology and Engineering (ICATE), pp. 1-9, IEEE, 2013).

[17] J. Walters-Williams and Y. Li. "Comparative study of distance functions for nearest neighbors," (Advanced techniques in computing sciences and software engineering, pp. 79-84. Springer, Dordrecht, 2010).