

# Automatic Reactive Power Classification Based on Selected Machine Learning Methods

Viktor Pristaš<sup>a</sup>, Ľubomír Antoni<sup>b</sup> and Gabriel Semanišin<sup>c</sup>

*Institute of Computer Science, Faculty of Science, Pavol Jozef Šafárik University, Jesenná 5, 041 80, Košice, Slovakia*

**Keywords:** Machine Learning,  $5 \times 2$  Cross-Validation Test, Reactive Power, Power Factor, Electricity System.

**Abstract:** Reactive power is an important part of the electric power systems in order to rotate machines or to transmit active power by transmission lines. However, an excess of reactive power in the electrical systems can increase the risk of failure of the transmission system. We present an automatic reactive power classification on multifamily residential dataset of electricity based on selected machine learning methods. We aim to predict an excess of reactive power in the apartments located in the Northeastern United States. Moreover, we explore the statistical significance of differences between mean performances of selected machine learning methods.

## 1 INTRODUCTION


Recently, the electricity system has undergone many important changes. As a result of technological development, the amount of distributed electricity is increasing, the nature of electrical appliances is changing, and cable lines are increasing (Alahmad et al., 2011; Maitre and Glon, 2015; Sarkar et al., 2018; Anaya and Pollitt, 2020).


If the balance of capacitive and inductive power in the distribution system is unbalanced, reactive electricity flows arise. An excess of reactive power causes an increase in the voltage in the electrical system. In periods of low load on individual lines, the voltage in the transmission system rises to values that exceed the permitted limit. Exceeding this limit increases the risk of failure of the transmission system and reduces its planned service life. Reactive power compensation technologies are thoroughly investigated to improve power quality (Dixon et al., 2005; Téllez et al., 2018; Vishnu and Kumar, 2020).


The identification of power stations with a possible future excess of reactive power can be formulated as the supervised machine learning classification. Several methods were proposed to tackle the classification tasks in various application domains, but none of them can claim being universally best (Alpaydin, 2010). However, several statistical tests for

deciding if one learning algorithm (supervised classification method) outperforms another on a particular learning task are generally described (Dietterich, 1998; Raschka, 2018). The  $5 \times 2$  cross-validation test is mostly recommended, since it is slightly more powerful and it measures variation due to the selection of training set.

In our paper, we applied several machine learning algorithms to classify the apartments from multifamily residential dataset of electricity (Meinrenken et al., 2020) due to their reactive power and power factor. Moreover, we applied  $5 \times 2$  cross-validation test to find if the difference in score measures between mean performance is probably real or not. In Section 2, we present the notions of reactive power, capacitive elements and induction elements. In Section 3, we describe the multifamily residential dataset of electricity, the extracted attributes and training set. We present our results of reactive power and power factor analysis in Section 4. We provide the review of the other studies with analysis of the multifamily residential dataset of electricity in discussion. The remarks and comments on our possible future work conclude the paper.

<sup>a</sup>  <https://orcid.org/0000-0001-9494-4122>

<sup>b</sup>  <https://orcid.org/0000-0002-7526-8146>

<sup>c</sup>  <https://orcid.org/0000-0002-5837-2566>

## 2 REACTIVE POWER, CAPACITIVE AND INDUCTION ELEMENTS

In general, power is defined as the amount of energy transferred or converted per unit time. In electrical circuits, power is defined as the product of instantaneous voltage and instantaneous current. In an alternating current (AC) circuit the voltage oscillates between positive and negative maximum values at the frequency of the network.

As a result, in a resistive circuit, the current also oscillates at the same frequency, because the resistive load current is directly proportional to the applied voltage. Figure 1 shows the correlation between voltage (red), current (blue) and power (green). Instantaneous power (since it is the product of the voltage and current) oscillates at twice the voltage frequency, but unlike the other two, in such a purely resistive current the power never drops to negative. This part of the power flow (or its averaged value over one cycle of the AC circuit), is known as real power or active power and it always flows from the direction of the source to the load while transferring the net energy (Zhou et al., 2018; Mukherjee et al., 2018; Mbinkar et al., 2022).

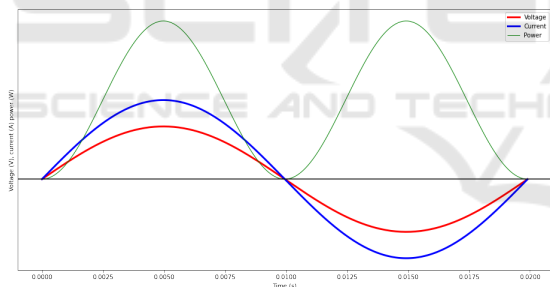


Figure 1: Voltage, current and power in a purely resistive AC circuit.

Distribution and transmission systems are proposed to transfer active power from power stations to energy consumers. The number of loads requires for operation not only the active power, but reactive power, as well. For example, AC motors require magnetic fields, which are generated by inductors that consume a so-called reactive power. Moreover, all components in inductive reactance distribution and transmission systems (cables, transformers) require reactive power.

Induction elements such as electric motors, transformers or induction cookers store energy in their magnetic field, which direction is opposite to the change in voltage. Thus, as the supply voltage rises, the net voltage on the inductor rises more slowly due

to the reverse voltage, which is induced by the inductor. This causes the current lagging behind the voltage since the current is proportional to the voltage in the circuit. In a purely inductive circuit, the current lags 90 degrees behind the voltage.

Capacitive elements e.g., capacitors, overhead power wires, electrical cables, or fluorescent lamps store energy in their electric field. The behavior of the capacitive circuit is best seen on the example of the capacitor. The pressure difference across the plates of a capacitor is the result of the accumulation of electrons on one plate and lack of electron on the other one. When the plates are uncharged, the increasing voltage from the supply sends the maximum current in the circuit to be stored on the plates until they become full of charge and no more charge is being added. At this point the voltage reaches its peak and the current is zero. In a pure capacitive circuit, the current leads the voltage by 90 degrees (Zhou et al., 2018; Mukherjee et al., 2018; Mbinkar et al., 2022).

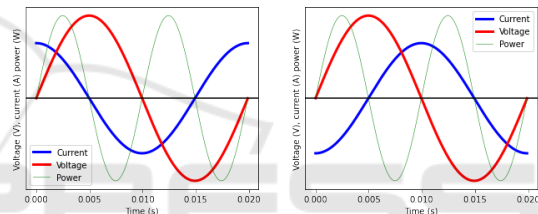


Figure 2: Voltage, current and power in an AC circuit where the current lags (left) and leads (right) the voltage.

Unlike in a resistive circuit, in a capacitive or inductive circuit the instantaneous power oscillates between a negative and a positive maximum, averaging zero during one network cycle, as it is seen in Figure 2. The seemingly negative part of the power is being returned to the source in each cycle and it is called reactive power. No net energy can be transmitted using reactive power.

The real networks consist of both resistive and reactive loads, which form together the complex (apparent) power expressed as volt-amperes (VA). The ratio of the active power to the apparent power defines the power factor of the circuit that describes the amount of the real power transmitted to the load along the transmission line to the total apparent power flowing in the line. The apparent power is the vector sum of active and reactive power, which is visualised in Figure 3 as the so-called power triangle (Maitre and Glon, 2015; Zhao et al., 2005).

The power factor might also be computed as the cosine of the angle  $\phi$ , called the phase angle, by which the current waveform leads or lags the voltage waveform. Both methods affirm, that the potential values

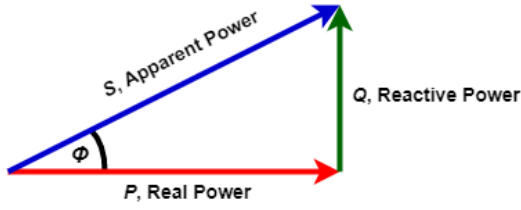


Figure 3: The power triangle of the real power, reactive power and apparent power.

of the power factor are within the interval (0,1): the closer the power factor is to 1, the more efficient the circuit, and vice versa.

### 3 METHODS

In this section, we describe our dataset, characteristics of attributes, training set and classification methods used for our analysis.

Our core focus is to advance research in the field of binary classification of electricity consumers based on their power factor. We used a period of three weeks as the observation period and one week as the resulting period. In particular, we explore if the power factor will be lower than 0.98 at least one day in the following week based on data about the power factor of the previous three weeks. Our aim is to find the most efficient algorithm and the most appropriate hyperparameter values for the chosen problem.

#### 3.1 Dataset

Multifamily residential dataset of electricity (Meinrenken et al., 2020) includes the annual electricity use of 390 apartments located in the Northeastern United States. These apartments are organized into 26 groups of 15 apartments each by their total electricity consumption in 2019. The apartments differ in the construction year (constructed before 1940, between 1940-1980, or constructed after 1980). Moreover, they use different cooling system or heating system (stem water, hot water, packaged terminal air conditioning units).

Time stamp, instantaneous real and reactive power, and cumulative electricity consumed are included for each of the 26 apartments groups. Data at 10-second or 15-minutes time resolution are available, which corresponds to 3,153,600 and 35,040 number of rows, respectively.

#### 3.2 Extracted Attributes and Training Set

For our analysis, we use the data at 15 minutes time resolution of instantaneous real power and reactive power.

Let  $O = \{o_1, o_2, \dots, o_{26}\}$  be the set of 26 groups of apartments and  $T = \{t_1, t_2, \dots, t_{35040}\}$  be the index of timestamp in 2019 at 15 minutes time resolution. Thus, the value  $P(o, t)$  corresponds to the instantaneous real power of the group  $o \in O$  in the time  $t \in T$ . Analogously, the value  $Q(o, t)$  corresponds to the instantaneous reactive power of the group  $o \in O$  in the time  $t \in T$ . The unit of active and reactive power is kW and kVar, respectively.

Since the phase angle and power factor are not included in the original multifamily residential dataset of electricity, we need to compute the phase angle as the binary relation

$$\phi(o, t) = \arctan \frac{P(o, t)}{Q(o, t)}$$

and power factor as the binary relation

$$F(o, t) = \cos(\phi(o, t))$$

for each  $o \in O$  and  $t \in T$ , i.e. at 15 minutes time resolution for each 26 groups of apartments.

Note that  $F(o, t) \in [-1; 1]$  for each  $o \in O$  and  $t \in T$ . If power factor is equal to 0 (i.e.,  $F(o, t) = 0$  for some  $o \in O$  and  $t \in T$ ), the energy flow is entirely reactive and the energy in the load returns to the source on each cycle. If the power factor is 1 (i.e.,  $F(o, t) = 1$  for some  $o \in O$  and  $t \in T$ ), all the energy supplied by the source is consumed by the load.

In order to construct the training set, we create a period of 3 weeks as the observation period and 1 week as the result period. In particular, our models are built by data from the observation period of 3 weeks to classify if the power factor will be lower than 0.98 at least one day in the following week. Thus, our target attribute has two categories.

Regarding the input attributes, we take 6 original attributes from the original multifamily residential dataset of electricity (number of bedrooms, number of all rooms, apartment area and their standard deviations). However, we constructed additional 28 attributes from the binary relation  $F(o, t)$  for each  $o \in O$  and  $t \in T$ . The extracted attributes are described in Table 1. Low excess of reactive power means that power factor  $F(o, t)$  will be lower than 0.98 at least in one day of the week. Medium excess means that the power factor  $F(o, t)$  will be lower than 0.98 at least in two days of the week. High excess means that the power factor  $F(o, t)$  will be lower than 0.98 at least in three days of the week.

Table 1: The extracted attributes and their types.

Characteristics	Period	Type
Month		nominal
Low excess	1.-3. week	binary
Low excess	1. week	binary
Low excess	2. week	binary
Low excess	3. week	binary
Medium excess	1. week	binary
Medium excess	2. week	binary
Medium excess	3. week	binary
High excess	1. week	binary
High excess	2. week	binary
High excess	3. week	binary
Proportion of excess	1. week	[0,1]
Proportion of excess	2. week	[0,1]
Proportion of excess	3. week	[0,1]
Average of $F$	1.-3. week	[0,1]
Standard deviation of $F$	1.-3. week	[0,1]
Minimum of $F$	1.-3. week	[0,1]
Maximum of $F$	1.-3. week	[0,1]
25th percentile of $F$	1.-3. week	[0,1]
75th percentile of $F$	1.-3. week	[0,1]
Median of $F$	1.-3. week	[0,1]
Average of $F$	3. week	[0,1]
Standard deviation of $F$	3. week	[0,1]
Minimum of $F$	3. week	[0,1]
Maximum of $F$	3. week	[0,1]
25th percentile of $F$	3. week	[0,1]
75th percentile of $F$	3. week	[0,1]
Median of $F$	3. week	[0,1]

### 3.3 Learning Algorithms

To solve our binary classification task, we use the following learning algorithms (methods):

- naive classification by using the high excess in last three weeks (which served as a baseline),
- Gaussian naive Bayes,
- k-nearest neighbors classifier,
- support vector classifier,
- deep feedforward neural network,
- logistic regression,
- decision trees,
- AdaBoost classifier
- GradientBoosting classifier,
- random forests,
- balanced random forests classifier,

The main motivation for using these algorithms is to compare the performance of various popular

machine learning methods on the selected data set. For hyper-parameter optimization, we used the grid search method, which is thoroughly described for example in (Mantovani et al., 2016; Feurer and Hutter, 2019; Wu et al., 2019; Yang and Shami, 2020).

As our **baseline method**, we use the high excess of power factor of the last 3 weeks of the observation period. The baseline method is stronger than random assignment, since a high proportion of power factor excess in the actual period can be a strong indicator of a high proportion of power factor excess in the future.

**Naive Bayes Method:** is based on Bayes' theorem with the "naive" assumption of conditional independence between each pair of attributes given the value of the target attribute (Rish, 2001; Chen et al., 2020). We apply the Gaussian algorithm, since the likelihood of the features is assumed to be Gaussian.

**Nearest Neighbor Methods:** are based on finding a particular number of training examples closest in distance to the new example, and classify the label from these. The number of examples can be given by a user by parameter or can vary on the local density of points (Jiang et al., 2007; Gou et al., 2019). We use parameters  $k = 6$  for number of neighbors, distance metric and  $p = 3$  for the Minkowski metric.

**Support Vector Classification:** methods use a subset of training points for the decision function (in the role of support vector) and they provide diverse options for building the decision function (Christiani and Shawe-Taylor, 2000). These methods were proposed at AT&T Bell in New Jersey around 1995 and they were used to recognize handwritten digits on postal items. We use the linear kernel with regularization parameter of value 1.

**Feedforward Neural Network:** is a non-linear function approximator for classification, which contains the input layer, the output layer, and there can be one or more non-linear layers, called hidden layers, between input and output layers (Glorot and Bengio, 2010). We use the multilayer perceptron with one hidden layer of 300 neurons, relu activation function, adam solver for weight optimization, and constant learning rate.

**Logistic Regression:** is a linear model for classification which describes the probabilities of the possible outputs of a single example. The probabilities are modeled by applying a logistic function (Kleinbaum et al., 2002). We use maximum number of 500 iterations taken for the solvers to converge, both L1 and L2 penalty terms and the elastic-net mixing parameter with L1 ratio of 0.7.

A **decision tree** method builds in the data a hierarchical structure, implementing divide-and-conquer strategy. It is an efficient method which can be used

both for classification (just splitting the data into some categories) and regression (predicting also the typical values in different categories (Breiman, 2001; Alpaydin, 2010)). Decision trees are designed for dependent variables that take a finite number of unordered values, with prediction error measured in terms of misclassification cost. Decision trees are constructed using a recursive partitioning algorithm. This algorithm builds a tree by recursively splitting the training sample into smaller subsets. The algorithm has three main components:

- a way to select a splitting rule,
- a rule to determine when a tree node is terminal (termination criterion),
- a rule for assigning a value to each terminal node.

For decision tree, we use gini index to measure the quality of a split, the maximum depth of the tree of 5, and the minimum 4 of samples required to be at a leaf node.

**AdaBoost:** (abbreviation from Adaptive Boosting) classification is an ensemble classification method which is based on an iterative approach to learn from the mistakes of weak classifiers. It starts by training a classifier on the original data. Thus, the additional copies of the classifier are constructed on the same dataset. However, the weights of incorrectly classified examples are modified (Freund and Schapire, 1997). We use decision tree classifier as the base estimator for AdaBoost classification. We use the maximum number of 1000 estimators at which boosting is terminated.

**Gradient Boosting:** classifier is an ensemble classification method which generalizes the other boosting methods by allowing optimization of an arbitrary differentiable loss function (Friedman, 2001). We use decision tree classifier as the base estimator for gradient boosting classification. We use the number of 1000 boosting stages to perform.

**Random Forests:** are a combination of tree classifiers such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest (Breiman, 2001). The principle is averaging classifications over a large number of trees learned on randomly chosen subsets of features (predictors). We use 2000 decision trees in the random forest, the gini index to measure the quality of a split, the maximum depth of 5, and 5 features to consider if looking for the best split.

**Balanced Random Forest:** classifier is the extension of random forests since the classification tasks are often imbalanced which means that at least one of the categories comprises only a small minority of the data. This issue can be solved by adding the

class weights to the random forest classifier. Thus, it penalizes misclassifying the minority category. On the other hand, down-sampling the majority class and growing each tree on a more balanced data set can be applied (Chen et al., 2004). We use 2000 decision trees in the random forest, the gini index to measure the quality of a split, the maximum depth 5 of the tree, and 5 features to consider if looking for the best split.

### 3.4 Comparing Classification Methods

For the evaluation of algorithms, we used an extension of 5-fold cross validation which returns stratified folds. It means that the folds are constructed by preserving the percentage of samples for each class. In our experiments, we computed the accuracy, F1 score, and F2 score for each of the described methods.

Dietterich described the principles of five statistical tests (McNemar's test, a test for the difference of two proportions, the resampled paired  $t$  test, the  $k$ -fold cross-validated paired  $t$  test, and the  $5 \times 2$  cv paired  $t$  test) for deciding if one learning algorithm (supervised classification method) outperforms another (Dietterich, 1998; Raschka, 2018). He argued that the  $5 \times 2$  cross-validation test is mostly recommended since it is slightly more powerful and it measures variation due to the selection of the training set. By this recommended test, we performed five duplicates of 2-fold cross-validation for each pair of our methods (learning algorithm). In each duplicate, data from our paper are randomly partitioned into two equal sized sets, and each method (learning algorithm) is trained on each set and tested on the other set.

## 4 RESULTS

In this subsection we present our results, including the statistical tests for comparing supervised classification learning algorithms.

The overall results are described in Table 2. We present the accuracy, F1 score, and F2 score for each of the described methods.

In Figure 4, we present the comparison of F1 scores of 10 learning algorithms on folds of 5-fold cross validation. The abbreviations of the algorithms in diagram corresponds to the titles of methods introduced in Table 2. We found that the smallest difference between F1 scores of folds was observed by gradient boosting classifier.

The results of the  $5 \times 2$  cv paired  $t$  test (described in Subsection 3.4) are shown in Table 3. We found that there is a statistically significant difference between mean F1 score of random forest algorithm

Table 2: The overall results on binary classification task.

Algorithm	Accu- racy	F1 score	F2 score
Baseline	0.71	0.668	0.569
Naive Bayes (NB)	0.76	0.750	0.673
Nearest neighbors (NN)	0.78	0.808	0.818
Support vector (SV)	0.85	0.869	<b>0.883</b>
Neural network (NN)	0.85	0.868	0.880
Logistic regression (LR)	0.83	0.851	0.849
Decision tree (DT)	0.79	0.824	0.835
AdaBoost (AB)	0.83	0.848	0.845
Gradient boost (GB)	0.83	0.852	0.852
Random forest (RF)	<b>0.86</b>	<b>0.877</b>	0.880
Balanced forest (BF)	<b>0.86</b>	0.873	0.858

and naive Bayes algorithm, or between random forest algorithm and nearest neighbors algorithm. There is also a statistically significant difference between mean F1 score of balanced forest algorithm and naive Bayes algorithm, or between balanced forest algorithm and nearest neighbors algorithm. Surprisingly, if the  $5 \times 2$  cv paired  $t$  test is applied between neural network algorithm and logistic regression, difference between mean performance is probably real.

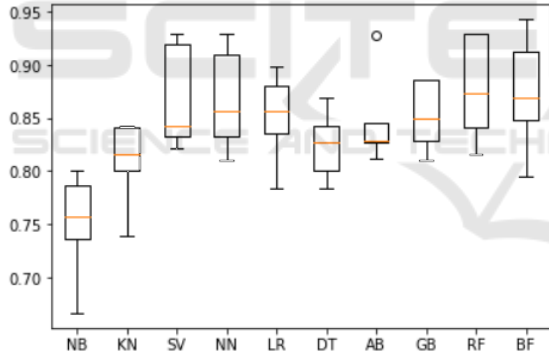


Figure 4: Comparison of F1 scores on folds.

## 5 DISCUSSION

Multifamily residential dataset of electricity (Meinrenken et al., 2020) provides the unique source of 12-month duration data of apartment electricity use in the multifamily area. This type of data is valuable since residential building sector responds for over 30% of the energy consumption worldwide (Meinrenken et al., 2020; Li et al., 2021a).

We have not found a similar study of power factor classification in the literature for this particular data set. However, we present the other interesting tasks solved on these data set in Discussion, which have

Table 3: Comparison of learning algorithms by  $5 \times 2$  cv paired  $t$  test.

	NB	KN	SV	NN	LR
NB	–	0.252	0.012*	0.042*	0.299
KN		–	0.036*	0.025*	0.055
SV			–	0.941	0.141
NN				–	0.021*

	DT	AB	GB	RF	BF
NB	0.370	0.038*	0.035*	0.011*	0.006*
KN	0.114	0.023*	0.024*	0.024*	0.027*
SV	0.648	0.908	0.713	0.268	0.178
NN	0.688	0.882	0.847	0.480	0.703
LR	0.885	0.229	0.254	0.056	0.111
DT	–	0.680	0.434	0.439	0.533
AB		–	0.650	0.364	0.234
GB			–	0.619	0.925
RF				–	0.687

\*statistically significant difference on level 0.05

been published by researchers recently. They mainly focus on the analysis of the active power and they do not consider the reactive power, as it is described in our paper. Thus, we hope that our paper can provide the added value in this interesting issue of electricity consumption.

Regarding the scientific papers related to this original dataset, researchers (Li et al., 2021b) explored the effects of outdoor temperature and COVID-19 related stay-at-home restrictions in residential electricity usage. They applied the set of regression methods to predict the average consumption over the specific time windows on weekdays. Moreover, they applied Monte Carlo methods to predict the effects of COVID-19 related stay-at-home restrictions. These types of results can help for managers to improve the balance of demand and supply in future. The results of data analysis can help citizens to load-shift part of their electricity usage, e.g. from day time to night.

In this way, the feedback to residents about the usage of electricity can help to immediately reduce demand, however an effect of boomerang can appear, as well (Meinrenken et al., 2021; Asensio and Delmas, 2015). The results of feedback effectiveness study have shown the average observed reduction of electricity usage of approximately 11% versus control group with no feedback. The text messages of feedback for residents were generated by the methods of natural language processing (Meinrenken et al., 2021).

Another study (Cen et al., 2022) analyzed multifamily residential dataset of electricity (Meinrenken et al., 2020) from the clustering point of view. The au-

thors presented four phases of their analysis – cleansing of data, extraction of features by multilevel discrete wavelet transformation, reduction of dimensionality by Pearson correlation coefficients and by principal component analysis, and the phase of clustering by  $k$ -means, fuzzy  $c$ -means, and hierarchical clustering. They found the representative electricity load patterns in three clusters – group of low load consumption, medium load consumption, or high load consumption including instability group. These patterns can be used to maintain the stability of power system or to design the optimal strategies for energy consumption.

We emphasize that other related datasets are available, as well. For example, a synthetic building operation dataset (Li et al., 2021a) contains electric loads, end-use energy consumptions, or historical weather data. It can be used to construct novel datasets with various conditions on weather, behavior of residents, or type of building.

Recently, a home energy management device was installed in the apartment house with 365 flats in Tokyo to measure the energy consumption data, and gas use (Yoshida et al., 2021). They argued that implementation of practises for energy saving of residents is becoming an important challenge and they proposed the optimal energy-saving behaviours.

Multifamily residential dataset of electricity (Meinrenken et al., 2020) contains both instantaneous active and reactive power. We would like to put emphasis that our approach provides the additional results to the mentioned researchers since we did not find the papers which analyze the reactive power for this dataset. However, the reactive power allows us to analyze the phase angle and the power factor of the apartments since it can help to reduce energy waste in homes and the built environment. Moreover, the optimization of reactive power and power factor can be fruitful in the process of building a smart environment, as well.

## 6 CONCLUSIONS

In this paper, we presented an automatic reactive power classification based on selected machine learning methods. We predicted an excess of reactive power in the particular buildings from multifamily residential electricity dataset. Moreover, we presented the  $5 \times 2$  cross validation test of the several machine learning methods. Finally, we discussed the added value of our paper in comparison with other related studies.

In our region, the energy company East Slovakia

Distribution distributes the electricity via its own distribution system to the end customer. Regarding our cooperation with this company, we have an access to real data of reactive power of 70,000 households, 30,000 companies and 5,000 transformer stations. In our future work, we plan to analyze these data of customers regarding the possibility to reduce the reactive power in the electricity system.

## ACKNOWLEDGEMENTS

This work was supported by the Scientific Grant Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under contract VEGA 1/0177/21 Descriptive and computational complexity of automata and algorithms (G. Semanišin, V. Pristaš) and under contract VEGA 1/0645/22 Proposal of novel methods in the field of Formal Concept Analysis and their application (L. Antoni). We would like to kindly thank the employees of the Data Collection and Management Department in energy company East Slovakia Distribution, Jozef Dudiak and Ján Pirigyí, for motivation and discussions on reactive power, as well as for the real data regarding our future work. We would like to thank Professor Peter Kollár, the head of Institute of Physics at Faculty of Science at Pavol Jozef Šafárik university, for his fruitful remarks and comments on reactive power principles.

## REFERENCES

- Alahmad, M., Hasna, H., and Sordiashie, E. (2011). Non-intrusive electrical load monitoring and profiling methods for applications in energy management systems. In *IEEE long island systems, applications and technology conference*, pages 1–6. IEEE.
- Alpaydin, E. (2010). *Introduction to Machine Learning*. MIT Press.
- Anaya, K. L. and Pollitt, M. G. (2020). Reactive power procurement: A review of current trends. *Applied Energy*, 270:114939.
- Asensio, O. and Delmas, M. (2015). Nonprice incentives and energy conservation. *Proceedings of the National Academy of Sciences*, 112(6):E510–515.
- Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.
- Cen, S., Yoo, J. H., and Lim, C. G. (2022). Electricity pattern analysis by clustering domestic load profiles using discrete wavelet transform. *Scientific data*, 15(4):1350.
- Chen, C., Liaw, A., and Breiman, L. (2004). *Using random forest to learn imbalanced data*. University of California, Berkeley.

- Chen, S., Webb, G. I., Liu, L., and Ma, X. (2020). A novel selective naïve bayes algorithm. *Knowledge-Based Systems*, 192:106361.
- Christiani, N. and Shawe-Taylor, J. (2000). *An introduction to support vector machines*. Cambridge University Press.
- Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.
- Dixon, J., Moran, L., Rodriguez, J., and Domke, R. (2005). Reactive power compensation technologies: State-of-the-art review. *Proceedings of the IEEE*, 93(12):2144–2164.
- Feurer, M. and Hutter, F. (2019). Hyperparameter optimization. In *Automated machine learning*, pages 3–33. Springer, Cham.
- Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139.
- Friedman, J. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5).
- Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*.
- Gou, J., Ma, H., Ou, W., Zeng, S., Rao, Y., and Yang, H. (2019). A generalized mean distance-based k-nearest neighbor classifier. *Expert Systems with Applications*, 115:356–372.
- Jiang, L., Cai, Z., Wang, D., and Jiang, S. (2007). Survey of improving k-nearest-neighbor for classification. In *Fourth international conference on fuzzy systems and knowledge discovery (FSKD 2007)*, pages 679–683.
- Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., and Klein, M. (2002). *Logistic regression*. New York: Springer-Verlag.
- Li, H., Wang, Z., and Hong, T. (2021a). A synthetic building operation dataset. *Scientific data*, 8(1):1–13.
- Li, L., Meinrenken, C. J., Modi, V., and Culligan, P. J. (2021b). Impacts of covid-19 related stay-at-home restrictions on residential electricity use and implications for future grid stability. *Energy and Buildings*, 251:111330.
- Maitre, J. and Glon, G. (2015). Efficient appliances recognition in smart homes based on active and reactive power, fast fourier transform and decision trees. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*. SCITEPRESS.
- Mantovani, R. G., Horváth, T., Cerri, R., Vanschoren, J., and de Carvalho, A. C. (2016). Hyper-parameter tuning of a decision tree induction algorithm. In *2016 5th Brazilian Conference on Intelligent Systems (BRACIS)*, pages 37–42. IEEE.
- Mbinkar, E. N., Asoh, D. A., and Kujabi, S. (2022). Microcontroller control of reactive power compensation for growing industrial loads. *Energy and Power Engineering*, 14(9):460–476.
- Meinrenken, C. J., Abrol, S., Gite, G. B., Hidey, C., McKeown, K., Mehmani, A., Modi, V., Turcan, E., Xie, W., and Culligan, P. J. (2021). Residential electricity conservation in response to auto-generated, multi-featured, personalized eco-feedback designed for large scale applications with utilities. *Energy and Buildings*, 232:110652.
- Meinrenken, C. J., Rauschkolb, N., Abrol, S., Chakrabarty, T., Decalf, V. C., Hidey, C., McKeown, K., Mehmani, A., Modi, V., and Culligan, P. J. (2020). Mfred, 10 second interval real and reactive power for groups of 390 us apartments of varying size and vintage. *Scientific Data*, 7:375.
- Mukherjee, S., Ganguly, A., Paul, A. K., and Datta, A. K. (2018). Load flow analysis and reactive power compensation. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, pages 207–211. IEEE.
- Raschka, S. (2018). Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack. *The Journal of Open Source Software*, 3(24).
- Rish, I. (2001). An empirical study of the naive bayes classifier. *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 3(22):41–46.
- Sarkar, M. N. I., Meegahapola, L. G., and Datta, M. (2018). Reactive power management in renewable rich power grids: A review of grid-codes, renewable generators, support devices, control strategies and optimization algorithms. *IEEE Access*, 6:41458–41489.
- Téllez, A. A., López, G., Isaac, I., and González, J. W. (2018). Optimal reactive power compensation in electrical distribution systems with distributed resources. review. *Heliyon*, 4(8):e00746.
- Vishnu, M. and Kumar, S. (2020). An improved solution for reactive power dispatch problem using diversity-enhanced particle swarm optimization. *Energies*, 13(11):2862.
- Wu, J., Chen, X. Y., Zhang, H., Xiong, L. D., Lei, H., and Deng, S. H. (2019). Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17(1):26–40.
- Yang, L. and Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415:295–316.
- Yoshida, K., Rijal, H. B., Bogaki, K., Mikami, A., and Abe, H. (2021). Field study on energy-saving behaviour and patterns of air-conditioning use in a condominium. *Energies*, 14(24):8572.
- Zhao, Y., Irving, M. R., and Song, Y. (2005). A cost allocation and pricing method for reactive power service in the new deregulated electricity market environment. In *2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific*, pages 1–6. IEEE.
- Zhou, X., Wei, K., Ma, Y., and Gao, Z. (2018). A review of reactive power compensation devices. In *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 2020–2024. IEEE.