

## Article

# Forecasting of 10-Second Power Demand of Highly Variable Loads for Microgrid Operation Control

Miroslaw Parol <sup>1</sup>, Pawel Piotrowski <sup>1,\*</sup> , Piotr Kapler <sup>1</sup> and Mariusz Piotrowski <sup>2</sup>

<sup>1</sup> Faculty of Electrical Engineering, Warsaw University of Technology, Koszykowa 75 Street, 00-662 Warsaw, Poland; miroslaw.parol@ee.pw.edu.pl (M.P.); piotr.kapler@ee.pw.edu.pl (P.K.)

<sup>2</sup> MP System Mariusz Piotrowski, Przejeczna 6/25 Street, 16-001 Ignatki-Osiedle, Poland; mpmpsystem@gmail.com

\* Correspondence: pawel.piotrowski@ee.pw.edu.pl; Tel.: +48-22-234-7314

**Abstract:** This paper addresses very short-term (10 s) forecasting of power demand of highly variable loads. The main purpose of this study is to develop methods useful for this type of forecast. We have completed a comprehensive study using two different time series, which are very difficult to access in practice, of 10 s power demand characterized by big dynamics of load changes. This is an emerging and promising forecasting research topic, yet to be more widely recognized in the forecasting research community. This problem is particularly important in microgrids, i.e., small energy micro-systems. Power demand forecasting, like forecasting of renewable power generation, is of key importance, especially in island mode operation of microgrids. This is due to the necessity of ensuring reliable power supplies to consumers. Inaccurate very short-term forecasts can cause improper operation of microgrids or increase costs/decrease profits in the electricity market. This paper presents a detailed statistical analysis of data for two sample low voltage loads characterized by large variability, which are located in a sewage treatment plant. The experience of the authors of this paper is that very short-term forecasting is very difficult for such loads. Special attention has been paid to different forecasting methods, which can be applied to this type of forecast, and to the selection of explanatory variables in these methods. Some of the ensemble models (eight selected models belonging to the following classes of methods: random forest regression, gradient boosted trees, weighted averaging ensemble, machine learning) proposed in the scope of choice of methods sets constituting the models set are unique models developed by the authors of this study. The obtained forecasts are presented and analyzed in detail. Moreover, qualitative analysis of the forecasts obtained has been carried out. We analyze various measures of forecasts quality. We think that some of the presented forecasting methods are promising for practical applications, i.e., for microgrid operation control, because of their accuracy and stability. The analysis of usefulness of various forecasting methods for two independent time series is an essential, very valuable element of the study carried out. Thanks to this, reliability of conclusions concerning the preferred methods has considerably increased.



**Citation:** Parol, M.; Piotrowski, P.; Kapler, P.; Piotrowski, M. Forecasting of 10-Second Power Demand of Highly Variable Loads for Microgrid Operation Control. *Energies* **2021**, *14*, 1290. <https://doi.org/10.3390/en14051290>

Academic Editor: José Matas

Received: 18 January 2021

Accepted: 21 February 2021

Published: 26 February 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Keywords:** microgrids; operation control; big dynamics loads; power demand; very short-term forecasting; machine learning; interval type 2 fuzzy logic system



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Microgrids, i.e., small energy micro-systems, are autonomous systems which can operate in two different modes: in synchronous (parallel) mode with grids of distribution system operators or in the island mode. In both modes, one of the most important challenges is how to control the microgrid. In particular, this problem is essential in the island mode which needs for its operation proper very short-term forecasts of power demand and power generated from renewable energy sources. Especially, it is a key issue in industrial microgrids or in some public utility microgrids with big dynamics of power

demand. In this way, the issue of very short-term power demand forecasting is gaining prominence, in particular due to the necessity of ensuring reliable electricity supplies to consumers in such microgrids. It is also worth mentioning that inaccurate very short-term forecasts can cause improper operation of microgrids or increase costs/decrease profits in the electricity market.

### 1.1. Related Works

The first parts of the publications related to this paper are the ones which specifically address microgrids.

The subject of microgrids has been described thoroughly in the literature. A formal definition of microgrids as well as detailed characteristics of different kinds of microgrids are presented in [1,2]. Many other publications describe the idea of microgrids, e.g., [3,4]. A lot of publications concern the topic of microgrids operation control [3–14]. These papers address different aspects of the control with the use of active components of microgrids. A very comprehensive overview of publications concerning optimal control in microgrids is presented in [5,13,14]. These contributions describe different ways of controlling microgrids, in particular, they discriminate between two main control strategies: centralized control logic and decentralized (distributed) control logic. The centralized control logic is presented, e.g., in [6,13]. Papers [7–9,13,14] describe the distributed control logic in microgrids. In [13,14], the problem of optimal control operation in rural low voltage microgrids with the use of the centralized control logic and the Particle Swarm Optimization (PSO) algorithm as well as the decentralized (distributed cooperative) control logic and a modified Monte Carlo method is addressed. The subject-matter of implementation of optimal operation control algorithms in microgrids is presented in [10]. The model of predictive control in microgrids is described in [6]. Meanwhile, other papers [6,9,11] address the problem of microgrid operational control in the island mode. Paper [12] presents the use of fuzzy logic to build a coordinated control system of energy storage and dispatchable generation for microgrids.

The second part of the literature sources related to this paper are the ones which address short-term forecasting. Within this area, we distinguish between power generation forecasts and load demand forecasts.

Many different methods (models) can be used to prepare short-term or very short-term forecasts of wind power generation. Some examples are given as follows: the model of wavelet decomposition and weighted random forest optimized by the niche immune lion algorithm utilized for ultra-short-term wind power forecasts [15]; hybrid empirical mode decomposition and ensemble empirical mode decomposition models (for wind power forecasts) [16]; a neuro fuzzy system with grid partition, a neuro fuzzy system with subtractive clustering, a least square support vector regression, and a regression tree (in the forecasting of hourly wind power) [17]; different approaches for minute-scale forecasts of wind power [18]; the Takagi–Sugeno fuzzy model utilized for ultra-short-term forecasts of wind power [19]; the integration of clustering, two-stage decomposition, parameter optimization, and optimal combination of multiple machine learning approaches (for compound wind power forecasts) [20]; hybrid model using modified long short-term memory for short-term wind power prediction [21]; extreme gradient boosting model with weather similarity analysis and feature engineering (for short-term wind power forecasts) [22]; a model for wind power forecasts utilizing a deep learning approach [23]; hybrid model using data preprocessing strategy and improved extreme learning machine with kernel (for wind power forecasts) [24]; the use of dual-Doppler radar observations of wind speed and direction for five-minute forecasts of wind power [25]; the multi-stage intelligent algorithm which combines the Beveridge–Nelson decomposition approach, the least square support vector machine, and intelligent optimization approach called the Grasshopper Optimization Algorithm (for short-term wind power forecasts) [26]; hybrid model, which combines extreme-point symmetric mode decomposition, extreme learning machine, and PSO algorithm (for short-term wind forecasts) [27]; chicken swarm algorithm

optimization support vector machine model for short-term forecasts of wind power [28]; hybrid neural network based on gated recurrent unit with uncertain factors (for ultra-short-term forecasts of wind power) [29]; discrete time Markov chain models for very short-term wind power forecasts [30].

Similarly, different methods (models) can be used for very short-term or short-term solar (photovoltaic) power prediction. The examples are: using smart persistence and random forests for forecasts of photovoltaic energy production [31]; an ensemble model for short-term photovoltaic power forecasts [32]; hybrid method based on the variational mode decomposition technique, the deep belief network and the auto-regressive moving average model (for short-term solar power forecasts) [33]; a model which combines the wavelet transform, adaptive neuro-fuzzy inference system, and hybrid firefly and particle swarm optimization algorithm (for solar power forecasts) [34]; a physical hybrid artificial neural network for the 24 h ahead photovoltaic power forecast in microgrids [35]; a hybrid solar and wind energy forecasting system on short time scales [36].

Another aspect to be considered is short-term load demand forecasting. Selected methods used for this purpose are presented here: a random forest model utilized for short-term electricity load forecasts [37]; a technique based on Kohonen neural network and wavelet transform (for short-term load forecasts of power systems) [38]; univariate models based on linear regression and patterns of daily cycles of load time series (for short-term load forecasts) [39].

Several publications, e.g., [40–42], present the problem of very short-term forecasting of power demand. A very comprehensive overview of ultra-short-term and short-term load forecasting methods is offered in [43]. This overview addresses topics such as forecast time horizons and different areas and location types to which forecasts can be dedicated (small city, microgrid, smart building). Mainly, different kinds of neural networks are used in the papers for ultra-short-term and short-term power demand forecasts.

In the analyzed publications, loads associated with big dynamics of power demand are very rarely addressed by very short-term forecasting exercises. This is the clue and the key aspect of this paper. Very short-term forecasts refer to microgrid components characterized by significant dynamics in the level of received power. Two of the main technological components of a sewage treatment plant play the role of loads characterized by considerable power demand variability. Some other active and passive components are also part of the microgrid.

Very short-term forecasting of power demand of loads with big dynamics is presented in paper [44]. The current paper very significantly builds on [44]. In comparison with the study described in [44], the scope of studies presented in this paper has been significantly expanded. Two time series from a December day instead of the one from the same month described in [44] are analyzed. Thanks to this, the credibility of conclusions has considerably increased. Fifteen forecasting methods were subjected to extensive testing, including advanced ensemble methods, which utilize machine learning, in comparison with six forecasting methods described in [44]. Many more nonlinear forecasting methods have been tested. The number of evaluation (quality) criteria of forecasts has been increased from 4 to 7. Thanks to this, the analysis of obtained results has significantly deepened. The selection of explanatory variables to the models as well as the choice of predictors to ensemble methods is also a novelty in comparison with the study described in [44].

## 1.2. Objective and Contribution

The main objectives of this paper are described as follows:

- Conduct statistical analysis of two distinct time series of values of 10 s power demand together with the initial choice of input data with the use of six different selection methods;
- Verify the efficiency of 10 s horizon power demand forecasting with the use of 15 forecasting methods, including ensemble methods; more than 150 various models with different values of parameters/hyperparameters have been tested for this purpose;

- Verify a newly developed strategy of choosing the best four predictors to be included in ensemble methods; the strategy, apart from the criterion of the least RMSE error of forecasts as well as dissimilarity of predictors action, utilizes an additional criterion, with the biggest differences in rests from the forecasts obtained with the use of methods being candidates for inclusion into the ensemble;
- Check whether there are any effective methods for very short-term forecasting of 10 s power demand (characterized by big volatility of loads) which can be used in practice;
- Indicate forecasting methods which are most efficient for this type of time series based on our analysis of forecasting results obtained for the two time series studied here.

Below are described selected contributions of this paper:

- This study has been done using two distinct time series, very difficult in practice to access, of 10 s power demand of highly variable loads, which is a significant novelty in the area of forecasting. It is a new, very promising topic yet to be widely recognized by forecasting researchers;
- Very comprehensive analyses concerning usefulness of different forecasting methods have been conducted for two distinct time series—starting from methods utilizing only time series itself, to the newest machine learning techniques, and finally to ensemble methods using additional explanatory variables;
- The authors of the study have developed some of the ensemble models proposed in the range of choice of methods sets constituting the models set;
- Very significant, detailed suggestions concerning the most effective methods for very short-term forecasts of power demand of big dynamics of load changes have been developed; similarly, a list of the least useful methods for this type of time series has also been presented;
- The analysis of usefulness of various forecasting methods for two independent time series of very big dynamics of load changes is an essential, very valuable element of our study. Credibility of conclusions concerning the preferred methods has increased considerably thanks to the mentioned analysis.

Our studies allow us to conclude that there exist effective very short-term forecasting methods for 10 s power demand (characterized by big dynamics of load changes), which are suitable to practical use, although such a forecasting task is very complicated.

The remainder of this paper is organized as follows: Section 2.1 presents statistical analysis of the two time series of power demand data considered in this paper. The analysis leading to the selection of proper explanatory variables for forecasting methods is presented in Section 2.2. Section 3 describes the methods employed in this paper. The criteria used to evaluate the quality of individual forecasting models are discussed in Section 4. Section 5 presents an extensive comparative analysis of very short-term forecasts obtained with the use of different forecasting methods. Section 6 discusses the results and presents the findings of this paper. Main conclusions resulting from our study are included in Section 7. The paper ends with a list of references.

## 2. Data

### 2.1. Statistical Analysis of the Two Time Series of Power Demand Data

Power demand data in particular circuits of the sewage treatment plant were acquired using the WAGO PFC200 PLC controller, a part of a modern microprocessor system whose modular design allowed us to create a set matching the needs of our measurements. Measurements of network parameters on individual circuits were carried out using a special three-phase power measurement module. The module provides all measurement values, including: active/reactive/apparent power, energy consumption, power factor, phase angle, without involving large computing power of the control system. These numerous measurements allowed us to conduct comprehensive analysis of the power supply via a network located inside the facility considered here. The switching functions were realized by two-state input/output modules. Through the built-in router with a 3G modem, the acquired data were sent in an event-based manner directly to the server with

the SCADA system for further analysis. The mobile sim card worked in a dedicated private network within a private APN. This solution ensures that the communication system is isolated from the public communication network at the level of the mobile operator's network. Additional encryption was implemented at the level of the communication protocol based on IEC 61850-7 standard.

Statistical analysis is based on two time series from one December day. Each of the time series is the active power demand in different technological parts of the sewage treatment plant being components of the microgrid. The first time series represents a circuit which supplies power to the main process of the sewage treatment plant. The second time series represents a circuit supplying power to another important process of the sewage treatment plant. The daily time series includes 8640 periods of 10 s. In the first time series, 169 data points are missing. In the second time series, 144 data points are missing. Missing data from the December day were supplemented with information about the values neighboring the missing ones. Missing data occur in rather random places of the time series. In several cases, a longer fragment of the time series (several minutes) has been missing. The data from both time series were normalized separately for anonymization to relative units (1 relative unit is equal to an average value from the time series). The real power demand in both time series ranges from about 20 W to about 40 kW. The relationship between both time series is very small—the value of Pearson linear correlation coefficient is 0.100 (statistically significant at the 5% significance level). Table 1 shows selected statistical measures of the two mentioned time series of power demand.

**Table 1.** Statistical measures of the two time series of the power demand in the technological parts of the sewage treatment plant.

Statistical Measures	Time Series No. 1	Time Series No. 2
Max power [p.u.]	5.7789	4.6195
Min power [p.u.]	0.0032	0.0024
The ratio of maximum power to minimum power	1806	1925
The biggest power ratio between two consecutive 10 s periods [%]	848	1401
Average power [p.u.]	1	1
Median [p.u.]	0.8580	0.9572
Standard deviation [p.u.]	0.7150	0.4867
Variance [-]	0.5112	0.2369
Coefficient of variation [%]	71.4952	48.6662
Skewness [-]	0.2363	0.3138
Kurtosis [-]	−0.3084	1.1299

The first time series has characteristics significantly different from the second time series. The first time series has (more than two times) greater variance than the second one. Forecasting errors will be probably higher for the first time series, assuming that the random component is dominant. The Kolmogorov–Smirnov and Lilliefors tests show that both time series of power demand do not have normal distribution. Figure 1 shows the percentage distribution of cases in both time series depending on power demand. Power demand below 1 p.u. is definitely dominant in both time series. As much as 22% of power demand from the first time series are very small values, below 0.05 p.u. (which is at least 20 times less than the average value). In contrast, the second time series has only 5% of very small values (below 0.05 p.u.).

Figure 2 shows the two daily time series of power demand for the December day considered here. Dynamic changes (rapid jumps) in power demand are clearly visible. For both time series, periods of different length with very similar power demand are characteristic. These periods are usually alternated by much shorter periods (or isolated periods) of much bigger power demand. Meanwhile, Figure 3 shows a specific part of the December day for both time series (from 8:00 to 8:30 a.m.).

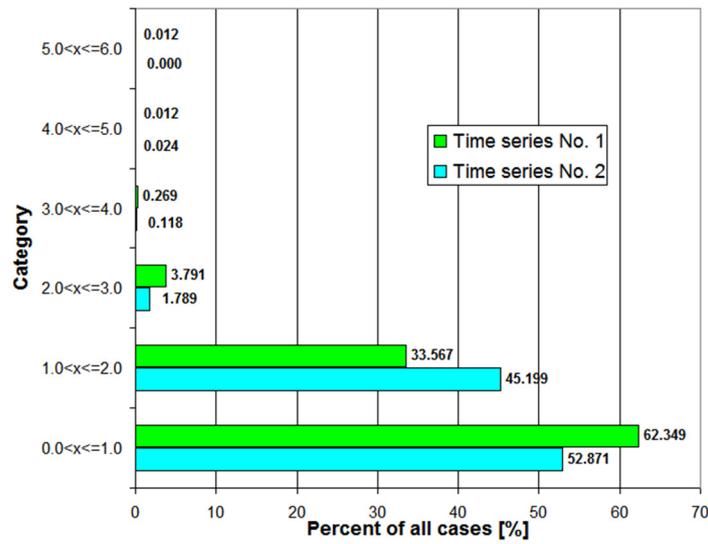


Figure 1. Percentage distribution of cases in both time series depending on the power demand value.

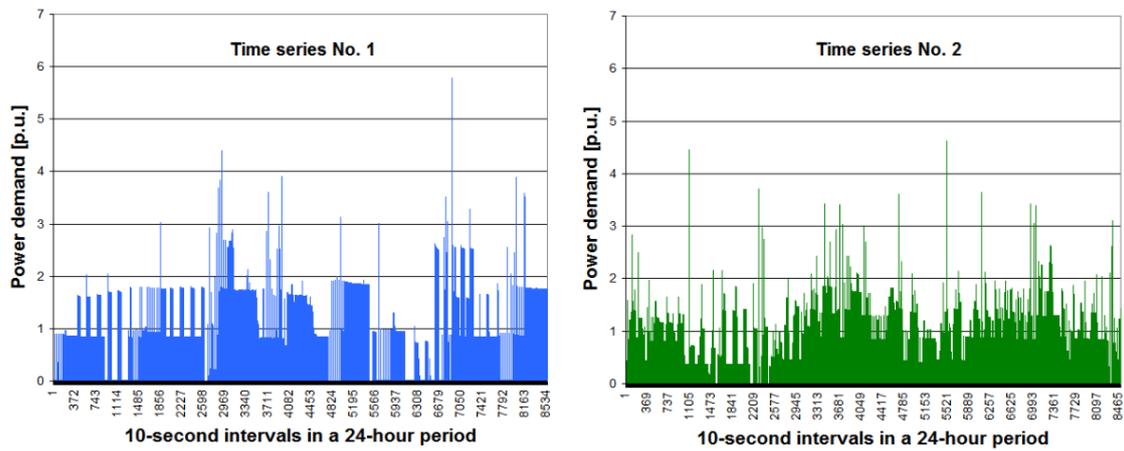


Figure 2. The two daily time series of power demand for the analyzed December day (figure of time series No. 1 is similar to the one from [44]).

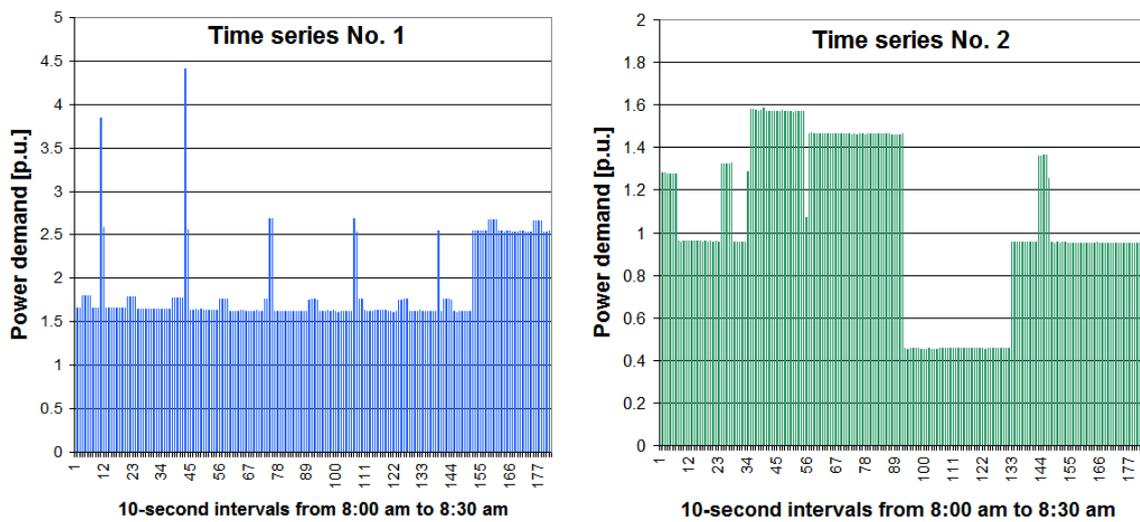


Figure 3. Two selected 30 min periods of power demand in the considered December day.

The seasonality problem of the time series of power demand was not analyzed due to unavailability of data. For both time series, it was examined whether there were daily variations in power demand in 1 h periods (see Figure 4). For each hour, average 10 s power demand was calculated. For the first time series,  $R^2$  equal to 0.3588 was obtained for the function of polynomial degree 8. For the second time series,  $R^2$  equal to 0.7464 was obtained for the function of polynomial degree 8. The value of the determination coefficient is low for the first time series and high for the second one. The function of polynomial degree 8 explains about 75% of the variability of the explanatory variable for the second time series. The value of Pearson linear correlation coefficients between the first time series and the second time series of average power demand in 1 h periods calculated based on the functions of polynomial degree 8 is 0.2956 (statistically significant at the 5% significance level).

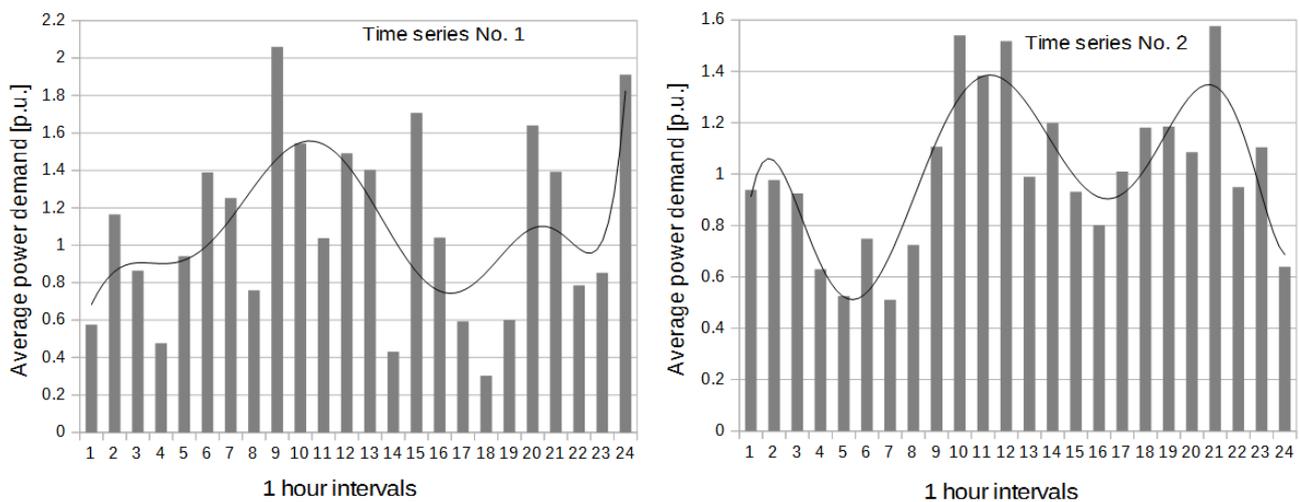


Figure 4. Daily variation of power demand in 1 h periods for both time series.

For the first time series, the autocorrelation coefficient dynamically decreases from 0.946 (1 period back) to 0.300 (90 periods back, i.e., 15 min) (see Figure 5). The autocorrelation coefficient temporarily increases (a big peak) up to 0.439 (150 periods back, i.e., 25 min). After 240 periods back (40 min), the autocorrelation coefficient falls to almost zero.

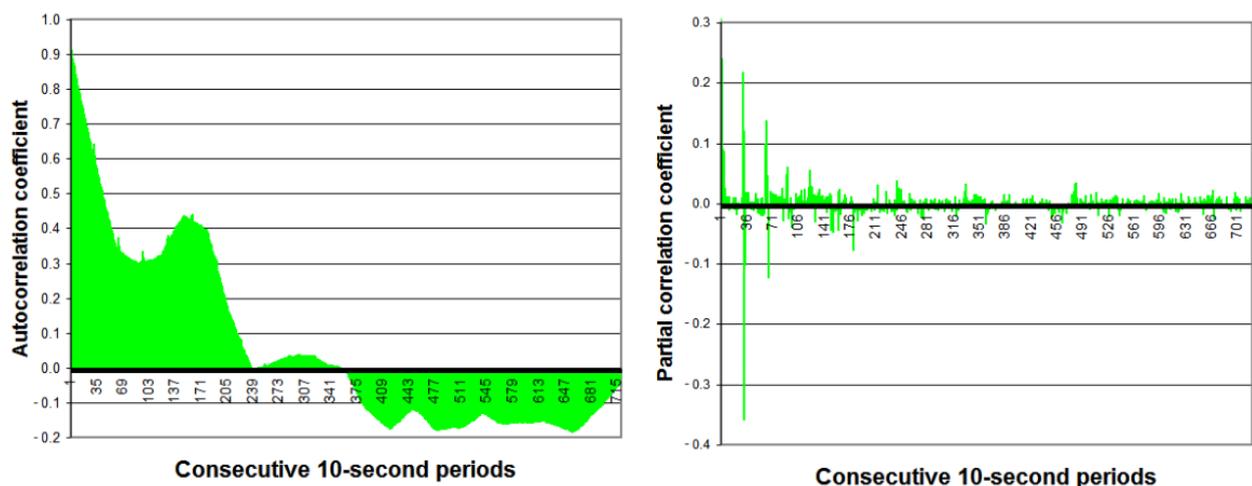
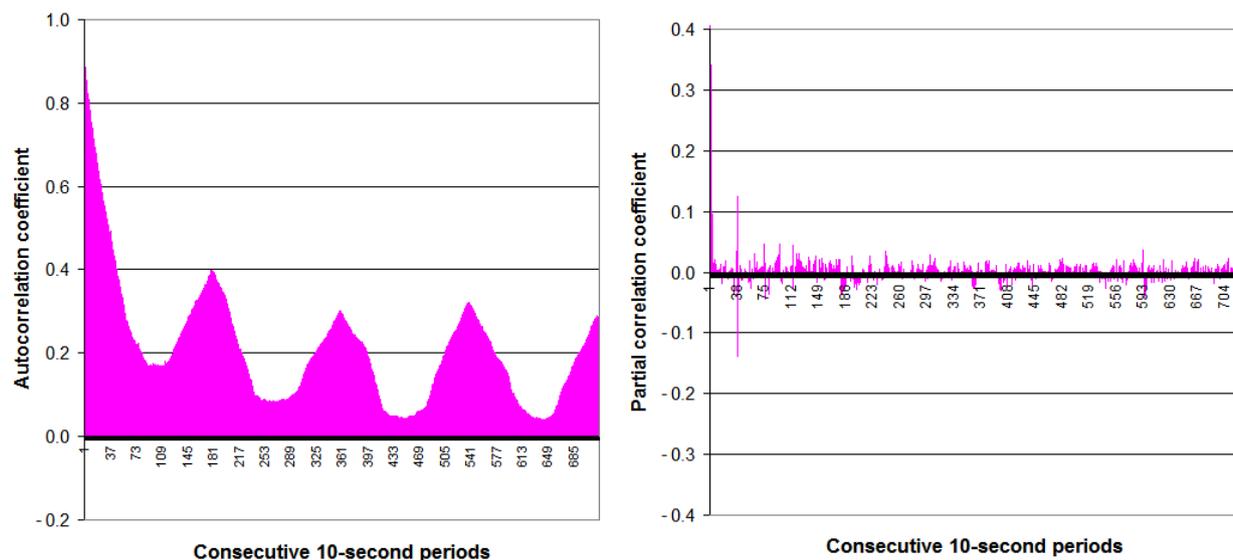


Figure 5. Autocorrelation function (ACF) and partial autocorrelation function (PACF) of the first time series in the analyzed December day (they are similar to figures of the ACF and PACF from [44]).

For the second time series, the autocorrelation coefficient dynamically decreases from 0.910 (1 period back) to 0.169 (90 periods back, i.e., 15 min) (see Figure 6). Unlike in the first time series, there are also temporary, repeated increases in the autocorrelation coefficient (big peaks) every 180 periods (i.e., 30 min) for the second time series. Autocorrelation coefficient for the first peak is 0.400. The next peaks gradually decrease. It means that the process element of the sewage treatment plant (second time series) probably has a 30 min periodicity.



**Figure 6.** Autocorrelation function (ACF) and partial autocorrelation function (PACF) of the second time series in the analyzed December day.

For both time series, the partial autocorrelation coefficients are statistically significant (5% significance level) only for the first several values and several single values distant from each other (see Figures 5 and 6).

The Table 2 shows, for both time series, Pearson linear correlation coefficients between 10 s power demand and the potential explanatory variables considered. All correlation coefficients are statistically significant (5% level of significance). For the variable coded AVEW (weighted average), optimal weights were calculated by optimization process using the Newton optimization algorithm which maximized the value of the correlation coefficient between the values of the AVEW and the time series. The sum of weights is equal to 1 (the limit for weights).

**Table 2.** Values of Pearson linear correlation coefficients between 10-second power demand of the first time series and the second time series and the explanatory variables considered (partially based on [44]).

Code of Variable	Potential Explanatory Variables Considered	Pearson Linear Correlation Coefficient	
		Time Series No. 1	Time Series No. 2
POW1	10 s power demand in period $t - 1$	0.9464	0.9100
POW2	10 s power demand in period $t - 2$	0.9114	0.8868
POW3	10 s power demand in period $t - 3$	0.8999	0.8707
POW4	10 s power demand in period $t - 4$	0.8882	0.8538
POW5	10 s power demand in period $t - 5$	0.8773	0.8412
POW6	10 s power demand in period $t - 6$	0.8657	0.8234
POW7	10 s power demand in period $t - 7$	0.8539	0.8092

Table 2. Cont.

Code of Variable	Potential Explanatory Variables Considered	Pearson Linear Correlation Coefficient	
		Time Series No. 1	Time Series No. 2
PEAK1	10 s power demand in period $t$ - $first\_peak$ , where $first\_peak$ is 150 (the first time series), and $first\_peak$ is 180 (for the second time series)	0.439	0.400
FUNC	24 h profile of the variation of power demand—the value of the polynomial function of degree 8 of power demand in a given hour	0.374	0.510
HOUR	The hour (number from the range 1–24) of power demand	0.0070	0.0061
AVE3	Average value of power demand from periods $t - 1$ , $t - 2$ , and $t - 3$	0.9402	0.9199
AVE2	Average value of power demand from periods $t - 1$ , and $t - 2$	0.9418	0.9195
AVEW	Sum of power demand values for period $t - 1$ with the weight $w_1$ , and period $t - 2$ with weight $w_2$ , where $w_1$ is 0.841, $w_2$ is 0.159 (for the first time series), and $w_1$ is 0.636, $w_2$ is 0.364 (for the second time series).	0.9478	0.9211

It is noteworthy that the sum of the power demand values from period  $t - 1$  with weight  $w_1$  and period  $t - 2$  with weight  $w_2$  (AVEW) has the maximum correlation coefficient of all potential explanatory variables for both time series. Moreover, the 24-h profile of the variation of power demand (FUNC) seems to be definitely a much better explanatory variable than the hour of power demand (HOUR).

## 2.2. Analysis of Potential Input Data for Forecasting Methods

Only time-lagged variables are used for the explained variable for methods based on the time series. The selection of explanatory variables is very often related to the process of testing the quality of models.

In the case of other methods using also exogenous variables, the process of selection of adequate explanatory variables can be supported by various methods of selection. Theoretically, for methods other than those based on time series, the forecasting model may contain 1 to 13 explanatory variables (see Table 2). Finding the optimal set of input variables for each model involves checking 8192 ( $2^{13}$ ) different combinations of variables (*brute force* method). Six methods of selection were used to select proper explanatory variables (see Figure 7).

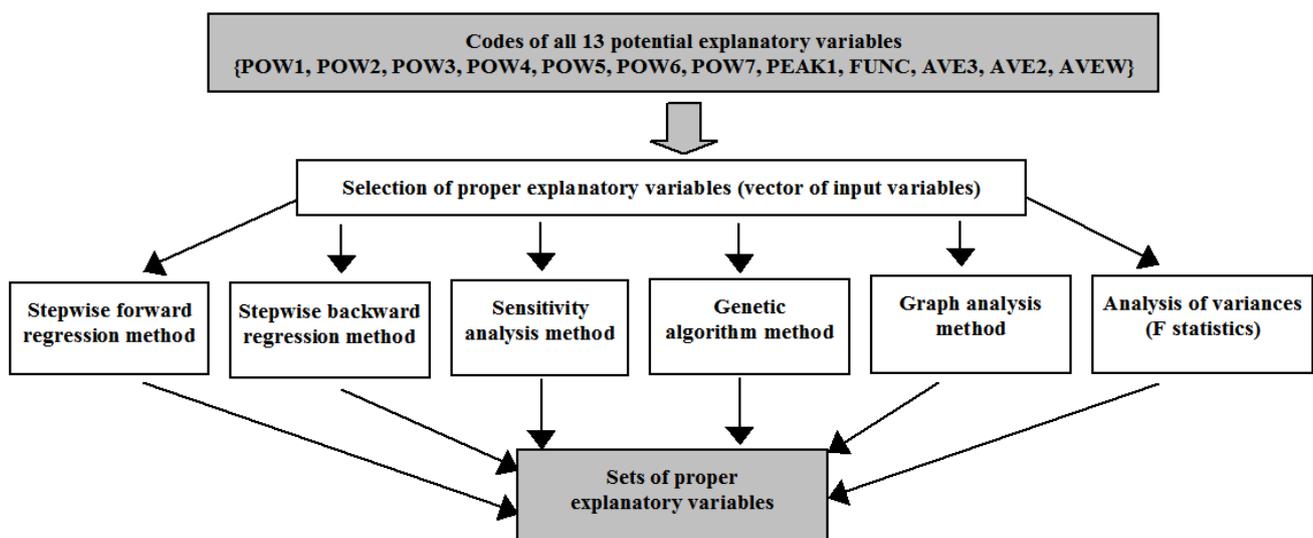


Figure 7. Scheme of selecting proper sets of explanatory variables.

The purpose of using these methods was to obtain a reduced set of explanatory variables without losing important information for the model and to eliminate non-essential explanatory variables from the model. The results of the selection of explanatory variables are only suggestions for the expert (see Tables 3 and 4). The expert makes the final decision on the selection of proper explanatory variables by testing forecasting models with various sets of explanatory variables.

**Table 3.** Results of initial selection of explanatory variables for all 6 methods of selection—selection for time series No. 1.

Method of Selection	Codes of All 13 Potential Explanatory Variables												
	PEAK1	FUNC	HOUR	AVEW	AVE3	AVE2	POW1	POW2	POW3	POW4	POW5	POW6	POW7
1-Stepwise forward regression	+	+	−	+	+	+	+	+	+	+	+	+	+
2-Stepwise backward regression	+	+	−	+	+	+	+	+	+	+	+	+	+
3-Sensitivity analysis	+	+	−	+	+	+	+	+	+	+	+	+	+
4-Genetic algorithm	+	−	−	+	+	+	+	+	+	+	−	+	+
5a-Graph analysis (strict selection)	+	+	−	+	−	−	−	−	−	−	−	−	−
5b-Graph analysis (inaccurate selection)	+	+	−	+	+	+	+	−	−	−	−	−	−
6-Analysis of variances	−	−	−	+	+	+	+	+	+	+	+	+	+
Summary—the number of indications	<b>6</b>	<b>5</b>	<b>0</b>	<b>7</b>	<b>6</b>	<b>6</b>	<b>6</b>	<b>5</b>	<b>5</b>	<b>5</b>	<b>4</b>	<b>5</b>	<b>5</b>

Remark: five best explanatory variables are bolded in the summary.

**Table 4.** Results of the initial selection of explanatory variables for all 6 methods of selection—selection for time series No. 2.

Method of Selection	Codes of All 13 Potential Explanatory Variables												
	PEAK1	FUNC	HOUR	AVEW	AVE3	AVE2	POW1	POW2	POW3	POW4	POW5	POW6	POW7
1-Stepwise forward regression	−	+	−	+	+	+	+	+	+	+	+	+	+
2-Stepwise backward regression	+	+	+	+	+	+	+	+	+	+	+	+	+
3-Sensitivity analysis	−	−	−	+	+	+	+	+	+	+	+	−	−
4-Genetic algorithm	+	+	+	+	+	−	+	+	+	+	−	−	+
5a-Graph analysis (strict selection)	+	+	−	+	−	−	−	−	−	−	−	−	−
5b-Graph analysis (inaccurate selection)	+	+	−	+	+	+	+	−	−	−	−	−	−
6-Analysis of variances	−	−	−	+	+	+	+	+	+	+	+	+	+
Summary—the number of indications	<b>4</b>	<b>5</b>	<b>2</b>	<b>7</b>	<b>6</b>	<b>6</b>	<b>6</b>	<b>5</b>	<b>5</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>3</b>

Remark: five best explanatory variables are bolded in the summary.

The analysis of the selection of input variables has shown that for both time series (No. 1 and No. 2) the input variable coded HOUR is the input variable most often selected for elimination. Therefore, the input variable coded HOUR has been eliminated from the forecasting process. The most often indicated input variables are coded AVEW (seven indications), AVE3, AVE2 and POW1 (six indications). For time series No. 1, the input variable coded PEAK1 also yielded six indications.

### 3. Forecasting Methods

This section describes the methods employed in this paper. Forecasts are made using single methods based only on the time series, the other single methods using also addi-

tional exogenous variables, including machine learning methods and the most advanced (complex) ensemble methods.

**Autoregressive Integrated Moving Average.** The autoregressive (AR) part of Autoregressive Integrated Moving Average (ARIMA) indicates that the evolving variable of interest is regressed on its own lagged values. The moving average (MA) part indicates that the regression error is actually a linear combination of error terms whose values occurred contemporaneously and at various times in the past. The term I (integrated) indicates that the data values have been replaced with the difference between their values and the previous values. The model has three parameters for tuning:  $p$  is the order (number of time lags) of the autoregressive model,  $d$  is the degree of differencing (the number of times the data have had past values subtracted) if the time series is not stationary, and  $q$  is the order of the moving-average model. The model is fitted using the least squares approach [45,46].

**Autoregressive model.** This is a simplified model of ARIMA. There is only one parameter for tuning— $p$  which is the number of time lags. The model with  $p = 1$  is only used for research, assuming that this model is a quality reference model for other methods. The model is fitted using the least squares approach [47,48].

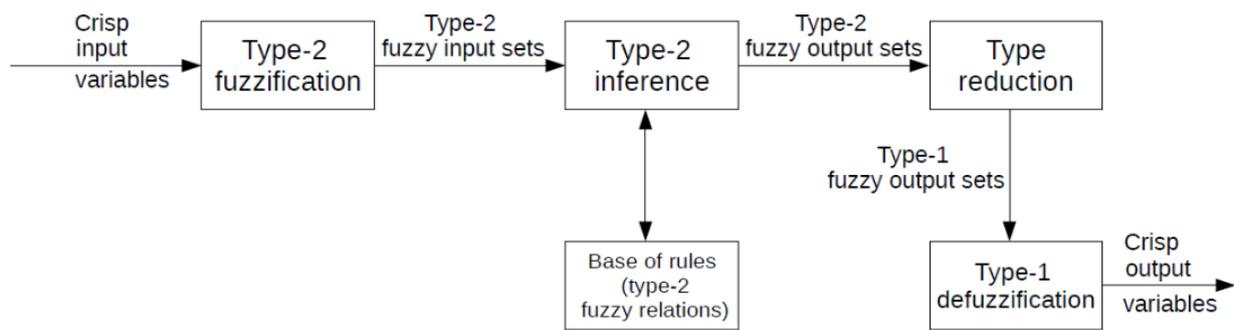
**Multiple Linear Regression model.** This is a linear model that assumes a linear relationship between input variables (more than one) and the single output variable. The input data are chosen values of lags of the forecasted output variable and other input explanatory variables correlated to the output variable. The model is fitted using the least squares approach [49].

**K-Nearest Neighbors.** The  $k$ -nearest neighbors algorithm is a non-parametric method used for classification and regression [49]. The input consists of  $k$  closest training examples in the feature space. In  $k$ -nearest neighbors (KNN) regression, the property value for the object is the output. This value is the average of the values of  $k$  nearest neighbors. The main hyperparameter for tuning is the number of nearest neighbors. Distance metric is the second hyperparameter.

**Support Vector Regression.** Support vector machine (SVM) for regression of the Gaussian kernel (non-linear regression) transforms the classification task into regression by defining a certain tolerance region with the width of  $\varepsilon$  around the target [49,50]. The learning task is reduced to the quadratic optimization problem and depends on several hyperparameters: regularization constant  $C$ , width parameter  $\sigma$  of the Gaussian kernel, and tolerance  $\varepsilon$ .

**Type MLP artificial neural network.** This is a class of feedforward artificial neural networks (ANNs). MLP is a popular and effective non-linear or linear (depending on the type of activation function in hidden layer/layers and output layer) global approximator [51,52]. It consists of one input layer, typically has one or two hidden layers, one output layer and often uses the backpropagation algorithm for supervised learning. The main hyperparameter for tuning is the number of neurons in hidden layer(s). The Broyden–Fletcher–Goldfarb–Shanno (BFGS) method used for solving unconstrained non-linear optimization problems has been chosen as a learning algorithm of the neural network.

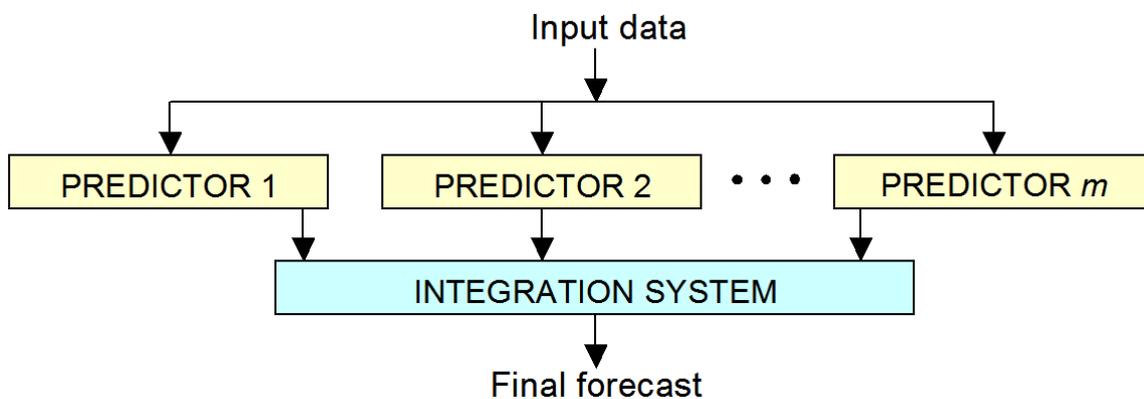
**Interval Type-2 Fuzzy Logic System.** Type-2 fuzzy logic systems (T2 FLSs) use type-2 fuzzy sets (T2 FS). Type-2 fuzzy sets were defined by L. Zadeh in 1975 as an extension of type-1 fuzzy sets (T1 FS). Further works on the subject were carried out by Karnik, Mendel and Liang [53–55]. T2 FSs are characterized by 3-dimensional membership functions (MF), which include footprint of uncertainty (FOU) [56]. The FOU and the third dimension of MF deliver additional degrees of freedom, which enables them to deal with uncertainties. The structure of type-2 fuzzy logic inference system is presented in Figure 8. The following components of T2 FLSs can be distinguished: fuzzification block, fuzzy inference block, base of fuzzy rules, type reduction block, and defuzzification block. The conversion of T2 FS to T1 FS takes place in the type reduction block. The Karnik–Mendel (KM) algorithm is usually applied for type reduction [54].



**Figure 8.** The structure of type-2 fuzzy logic inference system; elaborated based on [56].

Taking into account computational complexity of T2 FLSs, interval type-2 fuzzy logic systems (IT2FLSs) (see, e.g., [56]) are often used in practice [57]. The latter systems are characterized by a smaller computational effort of inference process in comparison with T2 FLSs. The IT2FLS with Takagi–Sugeno–Kang inference model, i.e., the IT2 TSK FLS [56], or the IT2FLS with Sugeno inference model, i.e., the IT2 S FLS, can be distinguished among the different IT2FLSs. The latter systems require a lower number of model parameters and in this way, they seem to be easier to use. In the training process of the system being designed to determine values of its particular parameters, genetic algorithm (GA) or PSO algorithm can be used.

Ensemble methods, using more than one individual predictor and supported by a simple or more complex integration system of individual forecasts, are addressed next. The general scheme of establishing an ensemble of predictors is presented in Figure 9. The simplest integration system is weighted averaging of individual predictors. The most advanced integrator systems are based on machine learning methods.



**Figure 9.** General structure of ensemble methods.

**Random Forest Regression.** Random forest is an ensemble method based on many single decision trees. Random forest is a versatile machine learning algorithm that can perform both classification and regression tasks. In the regression task, the prediction in a single decision tree is the average target value of all instances associated with the single leaf node. The final prediction is the average value of all  $m$  single decision trees. The regularization hyperparameters depend on the algorithm used, but generally restricted are: the maximum depth of the single decision tree, maximum number of levels in each decision tree, minimum number of data points placed in a node before the node is split, minimum number of data points allowed in a leaf node and maximum number of nodes. The number of predictors for each of the  $m$  single decision trees is made by the random choice of  $k$  predictors from all available  $n$  predictors [49,58,59].

**Gradient Boosted Trees for Regression.** Gradient boosting refers to an ensemble method that can combine several weak learners into a strong learner. It works by se-

quentially adding predictors (single decision tree) to the ensemble, each one correcting its predecessor. The method tries to fit the new predictor to the residual errors made by the previous predictor. The final prediction is the average value from all  $m$  single decision trees. This machine learning method is very effective and is recommended in many papers for forecasting tasks [22,60,61]. In comparison with random forest, this method has one additional hyperparameter—learning rate which scales the contribution of each tree [49,59,62,63].

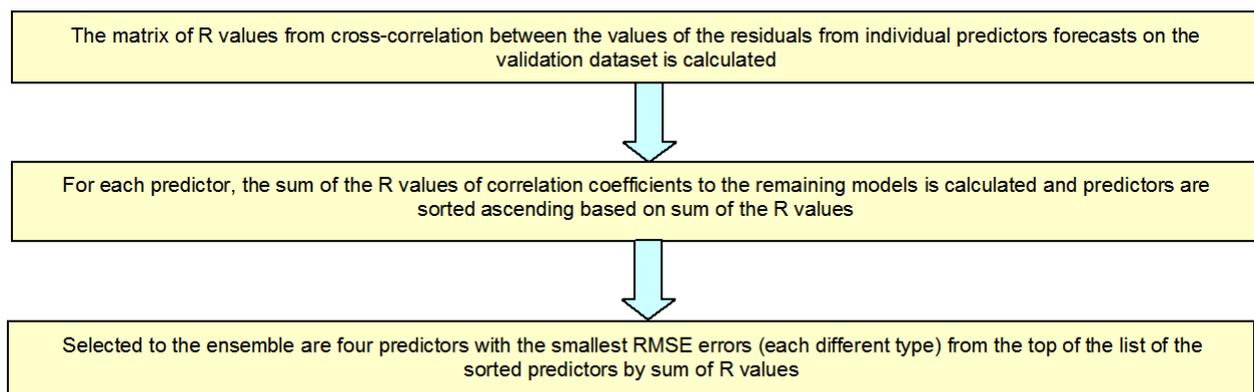
**Weighted Averaging Ensemble.** This is an integration of the results of selected predictors into final verdict of the ensemble. The final forecast is defined as the average of the results produced by all  $m$  predictors organized in an ensemble [50]. The final prediction result is calculated by Formula (1). This formula makes use of the stochastic distribution of the predictive errors. The process of averaging reduces the final error of forecasting.

$$\hat{y}_i = \frac{1}{m} \sum_{j=1}^m \hat{y}_i^j \quad (1)$$

where,  $i$  is the prediction point,  $\hat{y}_i$  is the final predicted value,  $\hat{y}_i^j$  is predicted value by predictor number  $j$ , and  $m$  is the number of predictors in ensemble. Note: all weights are equal in this case.

The averaging of the forecast results is an established method of reducing the variance of forecast errors. An important condition for including the predictor into the ensemble is independent operation from the others and also similar level of prediction error [50]. Two main strategies of predictor choice were tested:

- The choice of predictors based on RMSE (code WAE\_RMSE). Three or four predictors based on the smallest RMSE errors on validation subset and only predictors of different types are selected to the ensemble;
- The choice of predictors based on RMSE and R (code WAE\_R/RMSE). The time series of the residues from forecasts should be most different from each other and RMSE error should be small. The choice algorithm of predictors based on RMSE and R is presented in Figure 10.



**Figure 10.** Algorithm of selection of predictors to the ensemble based on RMSE and R.

**Machine Learning (ML) as an Integrator of Ensemble.** This is an integration of the results of the selected predictors into final verdict of the ensemble using ML model. Four predictors based on the smallest RMSE errors on validation subset are included in the ensemble, and these predictors must be of different types. Four ML methods are adopted as the integrator of the ensemble: MLP, KNN, random forest (RF) and linear regression (LR). Each ML integrator uses forecasts of power demand from four individual predictors as input data and the real values of power demand as the output. A training dataset is used for training ML integrator, and a validation dataset is used for RMSE control and tuning

of hyperparameters. Finally, the evaluation criteria are verified against the test dataset. The general structure of four machine learning methods as an integrator of ensemble is presented in Figure 11 [49,64].

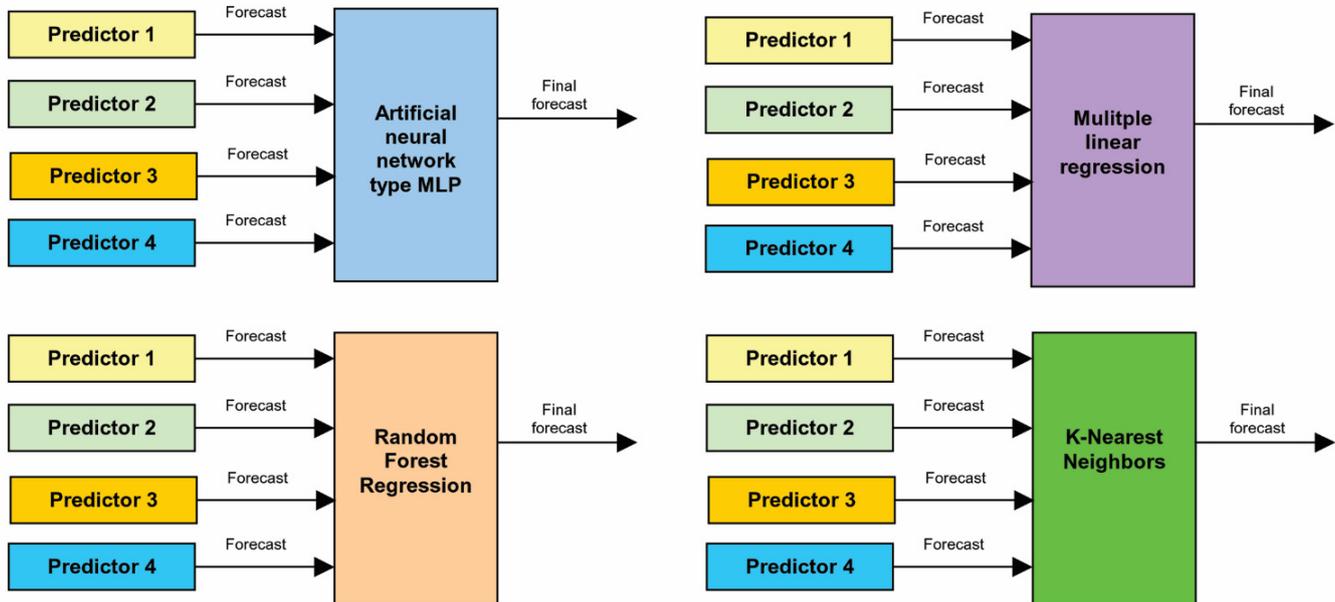


Figure 11. The general structure of four machine learning methods as an integrator of ensemble.

Summary description of fifteen tested forecasting methods is shown in Table 5. There are two methods based on time series and thirteen methods based on time series and exogenous input data. The eight methods are ensemble methods, and the seven ones are single methods. Ten methods all are machine learning methods.

Table 5. Summary description of fifteen tested forecasting methods.

Name of Method	Method Code	Category	Complexity (the Number of Predictors)	Type of Input Data
Autoregressive model	AR(1)	Linear/parametric	Single	Time series
Autoregressive Integrated Moving Average	ARIMA(p,d,q)	Linear/parametric	Single	Time series
Multiple Linear Regression model	LR	Linear/parametric	Single	Time series and exogenous variables
Type MLP artificial neural network	MLP	Non-linear or linear/parametric	Single	Time series and exogenous variables
Interval Type-2 Fuzzy Logic System	IT2FLS	Non-linear/parametric	Single	Time series and exogenous variables
K-Nearest Neighbors	KNN	Non-linear/non-parametric	Single	Time series and exogenous variables
Support Vector Regression	SVR	Non-linear/non-parametric	Single	Time series and exogenous variables
Random Forest Regression	RF	Non-linear/non-parametric	Ensemble	Time series and exogenous variables
Gradient Boosted Trees for regression	GBT	Non-linear/non-parametric	Ensemble	Time series and exogenous variables

Table 5. Cont.

Name of Method	Method Code	Category	Complexity (the Number of Predictors)	Type of Input Data
MLP as an Integrator of Ensemble	MLP [p1 *, ... ,p4]	Non-linear or linear/parametric	Ensemble	Forecasts from 4 predictors
RF as an Integrator of Ensemble	RF [p1,...,p4]	Non-linear/non-parametric	Ensemble	Forecasts from 4 predictors
KNN as an Integrator of Ensemble	KNN [p1,...,p4]	Non-linear or linear/non-parametric	Ensemble	Forecasts from 4 predictors
LR as an Integrator of Ensemble	LR [p1, ... ,p4]	linear/parametric	Ensemble	Forecasts from 4 predictors
Weighted Averaging Ensemble based on RMSE	WAE_RMSE [p1, ... ,pm]	Non-linear/parametric	Ensemble	Forecasts from <i>m</i> predictors
Weighted Averaging Ensemble based on R and RMSE	WAE_R/RMSE [p1,...,p4]	Non-linear/parametric	Ensemble	Forecasts from 4 predictors

Remark: \* denotes first predictor in ensemble of four predictors.

#### 4. Evaluation Criteria

In order to have a broader view of the quality of individual forecasting models, seven evaluation criteria are used. The main evaluation criterion is Root Mean Square Error (RMSE) due to the specificity of the forecast time series (more important for predictions of values which are greater than values close to zero or very small). Other major evaluation criteria are Maximum Absolute Error (MAE), Symmetric Mean Absolute Percentage Error (sMAPE) and Mean Absolute Percentage Error (MAPE) errors. Three auxiliary evaluation criteria are R (Pearson linear correlation coefficient), Mean Bias Error (MBE) and Maximum Absolute Error (MaxAE).

Root Mean Square Error is calculated by Formula (2). RMSE is sensitive to large errors and is more useful when large errors are particularly undesirable.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2)$$

where,  $\hat{y}_i$  is the predicted value,  $y_i$  is the actual value, and  $n$  is the number of prediction points.

Mean Absolute Error is calculated by Formula (3)

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (3)$$

In the process of forecasting of electric power demand, changes in RMSE and MAE have the same trend, and the smaller the two error values, the more accurate the prediction results.

Maximum Absolute Error is calculated by Formula (4)

$$MaxAE = \max_{i=1, \dots, n} |y_i - \hat{y}_i| \quad (4)$$

Mean Absolute Percentage Error is calculated by Formula (5). MAPE is asymmetric and it puts a heavier penalty on negative errors (when forecasts are higher than actuals) than on positive errors. This is due to the fact that percentage error cannot exceed 100% for forecasts that are too low. Meanwhile, there is no upper limit for forecasts which

are too high. As a result, MAPE will favor models that yield under-forecast rather than over-forecast.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \cdot 100\% \quad (5)$$

Symmetric Mean Absolute Percentage Error is calculated by Formula (6). sMAPE error overcomes the asymmetry of MAPE error—the boundlessness of the forecasts that are higher than the actuals and it has both the lower (0%) and the upper (200%) bounds.

$$sMAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i + \hat{y}_i} \right| \cdot 200\% \quad (6)$$

Mean Bias Error captures the average bias in the prediction and is calculated by Formula (7)

$$MBE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i) \quad (7)$$

Pearson linear correlation coefficient of the observed and predicted data is calculated by Formula (8)

$$R = \frac{C_{y\hat{y}}}{std(y) \cdot std(\hat{y})} \quad (8)$$

where,  $C_{y\hat{y}}$  is the covariance value between the actually observed and predicted data, and  $std$  denotes standard deviation of the appropriate variable.

The bigger the error  $R$  value (range from  $-1$  to  $1$ ), the more accurate the prediction results.

## 5. Results

This section presents an extensive comparative analysis of very short-term forecasting methods for active power demand in two different technological elements of the sewage treatment plant.

Both time series, including 8640 periods of 10 s, were divided into three subsets—training and validation subsets were composed of the initial 85% of time series (division into training and validation subsets which are different depending on the forecasting method) and the test subset comprised the last 15% of the time series. The training subset is used for estimation of model parameters. The validation subset is used for tuning hyperparameters of parts of methods. The test subset is applied to the final results of errors of the applied methods. For most of the methods, a random strategy choice of data for training and validation subsets was applied to get the right representation of data in each subset. Some methods, e.g., gradient boosted trees (GBT), used bootstrap technique for the choice of the training and validation subsets from the initial 85% data of the time series. Some other methods, e.g., AR, used only the training subset (the initial 85% of data in the time series) without validation subset.

Table 6 shows performance measures of proposed single methods (on test subset) for time series No. 1. Detailed description of the results of hyperparameters tuning for the single methods tested is provided in Table A1 of Appendix A.

**Table 6.** Performance measures of proposed single methods (on test subset) for time series No. 1.

Method Code	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
MLP(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	<b>0.2196</b>	<b>0.0641</b>	3.0142	<b>0.9454</b>	−0.0019	<b>123.71</b>	<b>25.25</b>
LR(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	<b>0.2197</b>	<b>0.0638</b>	3.0195	<b>0.9454</b>	<b>−0.0014</b>	135.51	27.32
LR(6_inputs)	PEAK1, POW1, POW2, POW3, POW5, POW7 (the choice based on backward stepwise regression from all input data)	<b>0.2198</b>	0.0663	3.0201	0.9453	<b>0.0009</b>	134.20	27.31

Table 6. Cont.

Method Code	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
ARIMA(5,1,1)	POW1—POW5	0.2200	<b>0.0640</b>	<b>3.0020</b>	<b>0.9455</b>	0.0030	129.44	26.46
MLP(5_inputs)	PEAK1, AVEW, AVE3, AVE2, POW1	0.2204	0.0733	<b>2.9908</b>	0.9450	<b>0.0018</b>	<b>116.39</b>	25.97
IT2FLS(5_inputs)	PEAK1, AVEW, AVE3, AVE2, POW1	0.2321	0.0715	2.9976	0.9360	−0.0035	204.83	33.29
AR(1)	POW1	0.2332	0.0661	3.0214	0.9396	0.0199	143.23	29.40
KNN(5_inputs)	PEAK1, AVEW, AVE3, AVE2, POW1	0.2347	0.0653	<b>3.0036</b>	0.9377	0.0160	<b>94.15</b>	<b>14.94</b>
KNN(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	0.2372	0.0908	3.0361	0.9376	0.0249	140.24	30.14
SVR(5_inputs)	PEAK1, AVEW, AVE3, AVE2, POW1 (the choice based on initial selection from Table 3)	0.2441	0.1278	<b>3.0397</b>	0.9425	−0.0186	447.35	39.30
SVR(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	0.2510	<b>0.1484</b>	3.0087	0.9401	<b>−0.0914</b>	<b>462.63</b>	<b>41.21</b>
IT2FLS(4_inputs)	POW1-POW4	<b>0.2936</b>	0.0924	3.0033	<b>0.9012</b>	0.0191	167.17	<b>22.03</b>

Remarks: The forecasts based on IT2FLS models were created using Matlab environment with Fuzzy Logic Toolbox and Optimization Toolbox. Three best fitting results for each fitting measure are printed in bold in blue. The worst fitting result is printed in red.

Table 7 presents performance measures of proposed ensemble methods (on test subset) for time series No. 1. Detailed description of the results of hyperparameters tuning for the ensemble methods tested is provided in Table A2 of Appendix A.

Table 7. Performance measures of proposed ensemble methods (on test subset) for time series No. 1.

METHOD CODE	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
GBT(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	<b>0.2102</b>	0.0591	<b>2.9823</b>	<b>0.9501</b>	<b>0.0016</b>	121.35	25.55
LR [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble *	<b>0.2126</b>	<b>0.0537</b>	<b>2.9968</b>	<b>0.9490</b>	0.0034	108.63	24.50
GBT (5_inputs)	PEAK1, AVEW, AVE3, AVE2, POW1	<b>0.2134</b>	<b>0.0536</b>	<b>2.9967</b>	<b>0.9487</b>	0.0047	97.75	22.81
MLP [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2140	<b>0.0526</b>	3.0011	0.9483	0.0047	94.22	20.45
WAE_RMSE [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2147	0.0587	<b>3.0073</b>	0.9479	<b>0.0013</b>	119.43	25.66
WAE_RMSE [LR(11_inputs), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2147	<b>0.0608</b>	3.0054	0.9479	<b>−0.0006</b>	<b>126.71</b>	<b>26.09</b>
WAE_R/RMSE [KNN(5_inputs), RF(10/12), AR(1), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2181	0.0583	3.0004	0.9462	0.0066	111.85	25.29

Table 7. Cont.

METHOD CODE	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
RF [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2185	0.0569	2.9998	0.9460	0.0047	<b>79.20</b>	<b>18.41</b>
KNN [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.2191	0.0583	3.0002	0.9458	0.0039	<b>67.12</b>	<b>10.23</b>
RF(10/12)	For each tree random choice of 10 inputs from 12: PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW7	0.2261	0.0582	3.0170	0.9421	0.0051	<b>89.83</b>	<b>22.58</b>
RF(6/12)	For each tree random choice of 6 inputs from 12: PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW7	<b>0.2261</b>	0.0567	3.0131	<b>0.9421</b>	<b>0.0070</b>	97.87	23.53

Remarks: \* Input datasets for each predictor in ensemble are described in Tables 6 and 7. Three best fitting results for each fitting measure are printed in bold in blue. The worst fitting result is printed in red.

Table 8 shows performance measures of proposed single methods (on test subset) for time series No. 2. Detailed description of the results of hyperparameters tuning for the single methods tested is provided in Table A3 of Appendix A.

Table 8. Performance measures of proposed single methods (on test subset range) for time series No. 2.

Method Code	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
LR(6_inputs)	FUNC, AVE2, POW1, POW3, POW4, POW5 (the choice based on backward stepwise regression from all input data)	<b>0.1827</b>	<b>0.0652</b>	<b>2.5893</b>	<b>0.9318</b>	<b>0.0008</b>	<b>94.83</b>	<b>14.04</b>
LR(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	<b>0.1832</b>	<b>0.0658</b>	<b>2.6106</b>	<b>0.9314</b>	−0.0011	98.19	14.22
ARIMA(3,1,1)	POW1—POW3	<b>0.1841</b>	<b>0.0640</b>	2.6843	<b>0.9330</b>	<b>−0.0001</b>	96.08	<b>13.96</b>
MLP(5_inputs)	FUNC, AVEW, AVE3, AVE2, POW1	0.1859	0.0686	2.7409	0.9293	−0.0026	129.88	14.89
MLP(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	0.1861	0.0691	2.7101	0.9291	−0.0010	<b>87.81</b>	<b>14.18</b>
SVR(5_inputs)	FUNC, AVEW, AVE3, AVE2, POW1 (the choice based on initial selection from Table 3)	0.1893	0.0960	2.7513	0.9269	−0.0118	<b>61.78</b>	16.38
SVR(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	0.1992	<b>0.1178</b>	2.6727	0.9293	−0.0416	<b>286.28</b>	<b>20.36</b>
KNN(5_inputs)	FUNC, AVEW, AVE3, AVE2, POW1	0.2017	0.0998	2.7910	0.9179	0.0045	282.84	18.16
IT2FLS(5_inputs)	FUNC, AVEW, AVE3, AVE2, POW1	0.2035	0.0778	2.8036	0.9048	0.0033	218.13	14.86
AR(1)	POW1	0.2048	0.0700	2.7143	0.9173	<b>0.0007</b>	129.19	14.69
KNN(11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1—POW6	0.2080	0.1064	<b>2.6502</b>	0.9119	0.0180	231.56	18.21
IT2FLS(4_inputs)	POW1-POW4	<b>0.2410</b>	0.1033	<b>2.8139</b>	<b>0.8799</b>	0.0048	197.56	17.34

Remarks: Three best fitting results for each fitting measure are printed in bold in blue. The worst fitting result is printed in red.

Table 9 presents performance measures of proposed ensemble methods (on test subset) for time series No. 2. Detailed description of the results of hyperparameters tuning for the ensemble methods tested is provided in Table A4 of Appendix A.

**Table 9.** Performance measures of proposed ensemble methods (on test subset) for time series No. 2.

Method Code	Input Data Codes	RMSE	MAE	MaxAE	R	MBE	MAPE [%]	sMAPE [%]
MLP [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Different, depends on predictor in ensemble *	<b>0.1721</b>	0.0693	2.5275	<b>0.9398</b>	−0.0043	96.95	13.93
GBT (11_inputs)	PEAK1, FUNC, AVEW, AVE3, AVE2, POW1–POW6	<b>0.1723</b>	0.0660	2.5283	<b>0.9397</b>	<b>0.0028</b>	92.90	<b>13.35</b>
LR [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	<b>0.1726</b>	0.0645	2.5364	<b>0.9394</b>	0.0032	89.47	<b>13.11</b>
WAE_RMSE [LR(11_inputs), MLP(5_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.1777	<b>0.0635</b>	2.5455	0.9356	<b>0.0003</b>	104.84	14.15
WAE_RMSE [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.1790	<b>0.0632</b>	2.5520	0.9346	<b>0.0002</b>	102.60	14.07
GBT (5_inputs)	FUNC, AVEW, AVE3, AVE2, POW1	0.1808	0.0647	2.6952	0.9332	0.0045	<b>125.21</b>	<b>14.31</b>
WAE_R/RMSE [MLP(5_inputs), RF(10/12), AR(1), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.1815	<b>0.0635</b>	2.5485	0.9330	0.0034	108.46	13.96
KNN [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.1848	0.0689	2.5961	0.9309	0.0135	<b>75.77</b>	<b>12.79</b>
RF [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Different, it depends on predictor in ensemble	0.1929	<b>0.0762</b>	2.5664	0.9257	<b>0.0202</b>	<b>87.51</b>	14.64
RF (10/12)	For each tree random choice of 10 inputs from 12: PEAK1, FUNC, AVEW, AVE3, AVE2, POW1–POW7	0.2008	0.0746	2.7032	0.9182	0.0127	<b>87.79</b>	14.26
RF (6/12)	For each tree random choice of 6 inputs from 12: PEAK1, FUNC, AVEW, AVE3, AVE2, POW1–POW7	<b>0.2024</b>	0.0741	<b>2.7194</b>	<b>0.9177</b>	0.0175	93.98	14.19

Remarks: \* Input dataset for each predictor in ensemble are described in Tables 8 and 9. Three best fitting results for each fitting measure are printed in bold in blue. The worst fitting result is printed in red.

## 6. Discussion

This section discusses the forecasts obtained for active power demand in two different technological parts of a sewage treatment plant.

Based on the results from Table 6, the following preliminary conclusions can be drawn regarding the proposed single methods for time series No. 1:

- The best methods in terms of RMSE value are, respectively: MLP, LR, ARIMA; for MLP and LR, a more favorable solution is to utilize all 11 input data instead of 5 input data selected on the basis of the analysis of variables selection;
- An advantage of the LR method in comparison with MLP is very short time of calculations, low computational complexity and no necessity of proper selection of hyperparameter values which are required by MLP (only selection of the most advantageous input data may be needed);
- Qualitative differences between the best three methods are very small;
- The worst methods in terms of RMSE value, in comparison with the reference method AR(1), are: SVR, KNN and IT2FLS; for IT2FLS method the use of five input data instead of four last time-lagged values of variables of time series being forecast allowed to decrease RMSE error;
- A disadvantage of IT2FLS is very long time of calculations (the time depends on CPU and RAM memory of computer being used) and the necessity of proper selection of many hyperparameter values; for the KNN method with five input data, the smallest MAPE and sMAPE errors have been obtained from all tested methods, although the RMSE error for the method is significant in comparison with the best methods.

In turn, based on the results from Table 7, it is possible to formulate the following preliminary conclusions concerning the proposed ensemble methods for time series No. 1:

- The best ensemble methods are, respectively: GBT and LR as an integrator of ensemble; for GBT, it is more useful to utilize all 11 input data;
- GBT ensemble method seems to be more useful than LR as an integrator of ensemble, considering greater simplicity of the method (no necessity to integrate four different method types), in which the ensemble is an internal component of the algorithm (single decision trees);
- All ensemble methods have smaller RMSE errors than the reference AR(1) method;
- Among ensemble methods, only RF as an integrator of ensemble has greater RMSE error than the best single method, i.e., MLP; it clearly shows that using ensemble methods (except RF) is justified in this forecasting exercise and allows us to decrease RMSE error in comparison with single methods;
- Similarly, as in the KNN single method, using KNN as an integrator of ensemble allows us to get the smallest MAPE and sMAPE errors of all tested ensemble methods, at the same time the RMSE error for the method is significant in comparison with the best ensemble methods;
- WAE\_R/RMSE ensemble method is less advantageous than WAE\_RMSE ensemble method;
- RF is the worst of all ensemble methods.

Based on the results from Table 8, the following preliminary conclusions can be drawn which concern the proposed single methods for time series No. 2:

- The best methods in terms of RMSE value are, respectively: LR, ARIMA and MLP; for MLP and for LR, it is more useful to utilize 5 input data, selected on the basis of the analysis of variables selection, instead of all 11 input data;
- Qualitative differences among the best three methods are very small;
- The worst methods in terms of RMSE value, in comparison with the reference method AR(1), are: KNN utilizing 11 input data and IT2FLS; for IT2FLS, the use of five input data instead of four last time-lagged values of variables of time series being forecast allowed us to decrease RMSE error.

Lastly, based on the results from Table 9, the following partial conclusions can be formulated concerning proposed ensemble methods for time series No. 2:

- The best ensemble methods are: MLP, GBT and LR as an integrator of ensemble; for GBT, it is more useful to utilize all 11 input data;
- GBT ensemble method seems to be more useful than MLP as an integrator of ensemble because of the greater simplicity of the method (no necessity to integrate four different

method types), in which the ensemble is an internal algorithm component (single decision trees);

- All ensemble methods have smaller RMSE errors than the reference method AR(1);
- Among ensemble methods, only the last four ones in the ranking (as integrators of ensemble) have a greater RMSE error than the best single method, i.e., LR(6); it suggests that using an ensemble method (except for the last four ones) is justified in this forecasting exercise and allows us to decrease RMSE error in comparison with single methods;
- As for the KNN single method, using KNN as an integrator of ensemble resulted in the smallest MAPE and sMAPE errors of all tested ensemble methods, at the same time the RMSE error for the method is significant in comparison with the best ensemble methods;
- WAE\_RMSE ensemble method is more useful than WAE\_R/RMSE ensemble method;
- RF is the worst of all ensemble methods.

Table 10 shows the best eight methods chosen from single and ensembles methods for both time series sorted in descending order by RMSE and the basic (reference) method.

**Table 10.** Summary comparison—the best eight methods chosen from single and ensembles methods for both time series sorted in descending order by RMSE and the basic (reference) method.

Results for Time Series No. 1			Results for Time Series No. 2		
Method Code	RMSE	Difference [%]	Method Code	RMSE	Difference [%]
<b>GBT</b> (11_inputs)	0.2102	-	<b>MLP</b> [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	0.1721	-
<b>LR</b> [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	0.2126	0.24	<b>GBT</b> (11_inputs)	0.1723	0.02
<b>GBT</b> (5_inputs)	0.2134	0.32	<b>LR</b> [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	0.1726	0.05
<b>MLP</b> [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	0.2140	0.38	<b>WAE_RMSE</b> [LR(11_inputs), MLP(5_inputs), GBT(11_inputs)]	0.1777	0.56
<b>WAE_RMSE</b> [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	0.2147	0.45	<b>WAE_RMSE</b> [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	0.1790	0.69
<b>WAE_RMSE</b> [LR(11_inputs), MLP(11_inputs), GBT(11_inputs)]	0.2147	0.45	<b>GBT</b> (5_inputs)	0.1808	0.87
<b>WAE_R/RMSE</b> [KNN(5_inputs), RF(10/12), AR(1), GBT(11_inputs)]	0.2181	0.79	<b>WAE_R/RMSE</b> [MLP(5_inputs), RF(10/12), AR(1), GBT(11_inputs)]	0.1815	0.94
<b>RF</b> [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	0.2185	0.83	<b>LR</b> (6_inputs)	0.1827	1.06
AR(1)—basic (reference) method	0.2332	10.94	AR(1)—basic (reference) method	0.2048	19.00

Actual values from test range (subset) of time series and the forecasted values by the best method are presented in Figure 12 for time series No. 1 and in Figure 13 for time series No. 2.

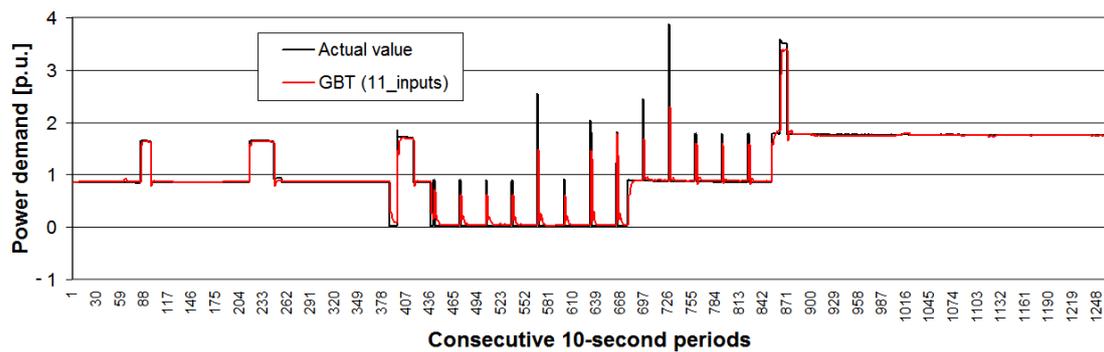


Figure 12. Actual values from the test range (subset) of time series No. 1 and the forecast values by the best method.

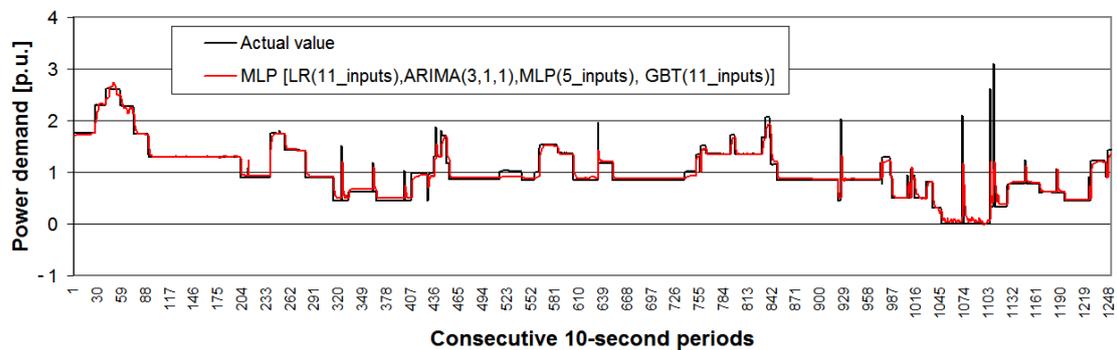


Figure 13. Actual values from the test range (subset) of time series No. 2 and the forecast values by the best method.

## 7. Conclusions

Very dynamic random rises of power demand of only 10 s duration (basic time interval) have been a major problem, and are impossible to forecast very accurately in practice. It is difficult to identify the best of the single methods. However, for both time series, the following three methods have always performed best: MLP, LR and ARIMA. Finally, from the point of view of forecasting efficiency (RMSE error value) and complexity level of the method, the most beneficial method is the GBT ensemble method, which utilizes all 11 input data. Comparative analysis of the results obtained for both time series shows that single methods—KNN, IT2FLS (with the use of four last time-lagged values of variables of time series being forecasted) and SVR—belong to the category of less recommended methods for this type of forecasting exercise.

Other observations and conclusions of the study can be summarized as follows:

- (1) The fact that the two time series being forecast are untypical ones means that achieving significantly smaller RMSE errors than in the case of the reference method AR(1) is difficult, but the studies carried out show that it is possible. For time series No. 1, improvement in comparison with the AR(1) method is equal to 10.94%, whereas for time series No. 2, improvement reaches 19.00% for the best methods being tested.
- (2) For both time series, the validity of using a vast majority of ensemble methods in comparison with single methods has been demonstrated. It is necessary to pay attention to the great complexity of such methods, their time demand as well as very small differences in RMSE error value for the best eight ensemble methods for both forecast time series.
- (3) RF is the worst performing ensemble method in both time series, thus this technique of machine learning should not be preferred as an ensemble method for this forecasting exercise.
- (4) Very small MAPE and sMAPE errors for the KNN method, used as a single method or applied as an integrator of ensemble, together with quite large RMSE error at the same time, is a characteristic and recurring phenomenon.

- (5) The impact of proper selection of input data varies from case to case; some methods achieved smaller RMSE errors with the use of 5 input data, whereas other methods performed better with the use of 11 input data, depending on time series being forecast, the exception being the GBT method, for which it was more useful to utilize all 11 input data.
- (6) For both time series, it was better to use the criterion of the least RMSE errors for selection of the four most beneficial predictors used for constructing the ensemble methods, as well as the rule of dissimilarity of predictor action, rather than to use an additional criterion of the biggest differences in rests from forecasts obtained with the use of methods being candidates to the ensemble.

The studies conducted allow us to conclude that there are effective methods of forecasting very short-term (10 s) power demand values (characterized by big dynamics of load changes) which are suitable for practical use, although it is a very complicated forecasting exercise. This is due to very dynamic random rises of the received power of 10 s duration. In further studies on methods of very short-term forecasting of power demand, we intend to conduct, among others, research on the development of forecasts for time intervals from  $t + 1$  to  $t + 6$ , that is, with the time horizon up to 1 min.

**Author Contributions:** Conceptualization, P.P. and M.P. (Mirosław Parol); formal analysis, P.P.; methodology, P.P., P.K. and M.P. (Mirosław Parol); investigation, P.P., P.K. and M.P. (Mariusz Piotrowski); supervision, M.P. (Mirosław Parol); validation, P.P.; writing, P.P., M.P. (Mirosław Parol), P.K. and M.P. (Mariusz Piotrowski); visualization P.P.; project administration, P.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Acknowledgments:** The dataset used for calculations in this paper has been collected under the RIGRID research project being part of the ERA-Net Smart Grid Plus initiative, having a support from the European Union's Horizon 2020 research and innovation program. The data acquisition system used for the mentioned data measurements has been also developed under the RIGRID project.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

ACF	Autocorrelation Function
APN	Access Point Name
AE	Absolute Error
EIASC	Enhanced iterative algorithm with stop condition
ANN	Artificial Neural Network
AR	Autoregressive model
ARIMA	Autoregressive Integrated Moving Average
BFGS	Broyden–Fletcher–Goldfarb–Shanno algorithm
EKM	Enhanced Karnik–Mendel
FLS	Fuzzy Logic System
GA	Genetic Algorithm
GBT	Gradient Boosted Trees
IASC	Iterative algorithm with stop condition
IT2FLS	Interval Type-2 Fuzzy Logic System
KM	Karnik–Mendel
KNN	K-Nearest Neighbors
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MBE	Mean Bias Error
ML	Machine Learning
MLP	Multi-Layer Perceptron

MaxAE	Maximum Absolute Error
LR	Linear Regression
PACF	Partial Autocorrelation Function
PSO	Particle Swarm Optimization
R	Pearson linear correlation coefficient
R2	Determination coefficient
RBF	Radial Base Function
RF	Random Forest
RMSE	Root Mean Square Error
sMAPE	Symmetric Mean Absolute Percentage Error
SVM	Support Vector Machine
SVR	Support Vector Regression

## Appendix A

Table A1 shows the results of hyperparameter tuning for proposed single methods for time series No. 1.

**Table A1.** Results of hyperparameter tuning for proposed single methods for time series No. 1.

Method Code	Description of Method, Name and Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameter Tuning
ARIMA(5,1,1)	The number of time lags $p$ , range: 1–8, selected: 5, degree of differencing $d$ , range: 1–3, selected: 1, order of the moving-average $q$ , range 1–3, selected: 1.	Manual, based on ACF and PACF results
SVR(11_inputs)	Regression SVM: Type-1, Type 2, selected: Type-1, kernel type: Gaussian (RBF), width parameter $\sigma$ : 0.077, regularization constant $C$ , range: 1–20 (step 1), selected: 2, tolerance $\epsilon$ , range: 0.05–2 (step 0.05), selected: 0.1.	Grid search using cross-validation
SVR(5_inputs)	Regression SVM: Type-1, Type 2, selected: Type-1, kernel type: Gaussian (RBF), width parameter $\sigma$ : 0.200, regularization constant $C$ , range: 1–20 (step 1), selected: 3, tolerance $\epsilon$ , range: 0.05–2 (step 0.05), selected: 0.1.	Grid search using cross-validation
KNN(11_inputs)	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 9.	Grid search using cross-validation
KNN(5_inputs)	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 12.	Grid search using cross-validation
MLP(11_inputs)	The number of neurons in hidden layer: 5–20, selected: 13, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: hyperbolic tangent, activation function in output layer: linear.	Manual, supported by cross-validation
MLP(5_inputs)	The number of neurons in hidden layer: 5–20, selected: 4, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: hyperbolic tangent, activation function in output layer: linear.	Manual, supported by cross-validation
IT2FLS(4_inputs)	Interval Type-2 Sugeno FLS, learning and tuning algorithm: GA, PSO, selected: PSO, initial swarm span: 1500–3000, selected: 1500, minimum neighborhood size: 0.20–0.30, selected: 0.20, inertia range: from [0.10–1.10] to [0.20–2.20], selected: [0.15–1.50], number of iterations in the learning and tuning process: 5–20, selected: 12, type of the membership functions: triangular, Gauss, selected: triangular, the number of output membership functions: 3–81, selected: 81, window size for computing average validation cost: 4–7, selected: 5, maximum allowable increase in validation cost: 0.0–1.0, selected: 0.1, the type-reduction methods: KM, EKM, IASC, EIASC, selected: KM.	Manual, supported by cross-validation
IT2FLS(5_inputs)	Interval Type-2 Sugeno FLS, learning and tuning algorithm: GA, PSO, selected: PSO, initial swarm span: 1500–3000, selected: 1500, minimum neighborhood size: 0.20–0.30, selected: 0.20, inertia range: from [0.10–1.10] to [0.20–2.20], selected: [0.15–1.50], number of iterations in learning and tuning process: 5–20, selected: 12, type of the membership functions: triangular, Gauss, selected: triangular, number of output membership functions: 3–243, selected: 243, window size for computing average validation cost: 4–7, selected: 5, maximum allowable increase in cost: 0.0–1.0, selected: 0.1, the type-reduction methods: KM, EKM, IASC, EIASC, selected: KM.	Manual, supported by cross-validation

Table A2 presents the results of hyperparameter tuning for the proposed ensemble methods for time series No. 1.

**Table A2.** Results of hyperparameter tuning for the proposed ensemble methods for time series No. 1.

Method Code	Description of Method, Name and Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameter Tuning
RF(6/12)	The number of predictors chosen at random: 6, number of decision trees: 1–500, selected: 352. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation
RF(10/12)	The number of predictors chosen at random: 10, number of decision trees: 1–500, selected: 421. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation
GBT(11_inputs)	The learning rate: 0.05, 0.1, 0.5, selected: 0.1, number of decision trees: 1–500, selected: 461. Stop parameters: maximum number of levels in each decision tree: 5, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 3, maximum number of nodes: 3.	Grid search using cross-validation, early stopping
GBT(5_inputs)	The learning rate: 0.05, 0.1, 0.5, selected: 0.1, number of decision trees: 1–500, selected: 440. Stop parameters: maximum number of levels in each decision tree: 5, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 3, maximum number of nodes: 3.	Grid search using cross-validation, early stopping
MLP [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	The number of neurons in hidden layer: 3–10, selected: 7, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: linear, activation function in output layer: linear.	Manual, supported by cross-validation
KNN [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 12.	Grid search using cross-validation
RF [LR(11_inputs), RF(6/12), MLP(11_inputs), GBT(11_inputs)]	The number of predictors by random choice: 3 from 4, number of decision trees: 1–400, selected: 241. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation

Table A3 shows the results of hyperparameter tuning for proposed single methods for time series No. 2.

**Table A3.** Results of hyperparameters tuning for proposed single methods for time series No. 2.

Method Code	Description of Method, the Name and the Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameters Tuning
ARIMA(3,1,1)	The number of time lags $p$ , range: 1–8, selected: 3, degree of differencing $d$ , range: 1–3, selected: 1, order of the moving-average $q$ , range 1–3, selected: 1.	Manual, based on ACF and PACF results
SVR(11_inputs)	Regression SVM: Type-1, Type 2, selected: Type-1, kernel type: Gaussian (RBF), width parameter $\sigma$ : 0.833, regularization constant $C$ , range: 1–20 (step 1), selected: 7, tolerance $\epsilon$ , range: 0.05–2 (step 0.05), selected: 0,1.	Grid search using cross-validation
SVR(5_inputs)	Regression SVM: Type-1, Type 2, selected: Type-1, kernel type: Gaussian (RBF), width parameter $\sigma$ : 0.200, regularization constant $C$ , range: 1–20 (step 1), selected: 8, tolerance $\epsilon$ , range: 0.05–2 (step 0.05), selected: 0,1.	Grid search using cross-validation

Table A3. Cont.

Method Code	Description of Method, the Name and the Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameters Tuning
KNN(11_inputs)	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 29.	Grid search using cross-validation
KNN(5_inputs)	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 14.	Grid search using cross-validation
MLP(11_inputs)	The number of neurons in hidden layer: 5–20, selected: 13, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: hyperbolic tangent, activation function in output layer: linear.	Manual, supported by cross-validation
MLP(5_inputs)	The number of neurons in hidden layer: 5–20, selected: 4, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: hyperbolic tangent, activation function in output layer: linear.	Manual, supported by cross-validation
IT2FLS(4_inputs)	Interval Type-2 Sugeno FLS, learning and tuning algorithm: GA, PSO, selected: PSO, initial swarm span: 1500–3000, selected: 1500, minimum neighborhood size: 0.20–0.30, selected: 0.20, inertia range: from [0.10–1.10] to [0.20–2.20], selected: [0.15–1.50], number of iterations in learning and tuning process: 5–20, selected: 12, type of the membership functions: triangular, Gauss, selected: triangular, number of output membership functions: 3–81, selected: 81, window size for computing average validation cost: 4–7, selected: 5, maximum allowable increase in validation cost: 0.0–1.0, selected: 0.1, type-reduction methods: KM, EKM, IASC, EIASC, selected: KM.	Manual, supported by cross-validation
IT2FLS(5_inputs)	Interval Type-2 Sugeno FLS, learning and tuning algorithm: GA, PSO, selected: PSO, initial swarm span: 1500–3000, selected: 1500, number of iterations in learning and tuning process: 5–20, selected: 12, minimum neighborhood size: 0.20–0.30, selected: 0.20, inertia range: from [0.10–1.10] to [0.20–2.20], selected: [0.15–1.50], type of the membership functions: triangular, Gauss, selected: triangular, number of the output membership functions: 3–243, selected: 243, window size for computing average validation cost: 4–7, selected: 5, maximum allowable increase in validation cost: 0.0–1.0, selected: 0.1, type-reduction methods: KM, EKM, IASC, EIASC, selected: KM.	Manual, supported by cross-validation

Table A4 presents the results of hyperparameter tuning for the proposed ensemble methods for time series No. 2.

Table A4. Results of hyperparameter tuning for the proposed ensemble methods for time series No. 2.

Method Code	Description of Method, the Name and the Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameters Tuning
RF (6/12)	The number of predictors chosen at random: 6, number of decision trees: 1–500, selected: 245. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation
RF (10/12)	The number of predictors chosen at random: 10, number of decision trees: 1–500, selected: 281. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation
GBT (11_inputs)	The learning rate: 0.05, 0.1, 0.5, selected: 0.1, number of decision trees: 1–500, selected: 392. Stop parameters: maximum number of levels in each decision tree: 5, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 3, maximum number of nodes: 3.	Grid search using cross-validation, early stopping

Table A4. Cont.

Method Code	Description of Method, the Name and the Range of Values of Hyperparameters Tuning and Selected Values	Method for Hyperparameters Tuning
GBT (5_inputs)	The learning rate: 0.05, 0.1, 0.5, selected: 0.1, number of decision trees: 1–500, selected: 448. Stop parameters: maximum number of levels in each decision tree: 5, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 3, maximum number of nodes: 3.	Grid search using cross-validation, early stopping
MLP [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	The number of neurons in hidden layer: 3–10, selected: 8, learning algorithm: BFGS, activation function in hidden layer: linear, hyperbolic tangent, selected: hyperbolic tangent, activation function in output layer: linear.	Manual, supported by cross-validation
KNN [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	Distance metrics: Euclidean, Manhattan, Minkowski, selected: Euclidean, number of nearest neighbors $k$ , range: 1–50, selected: 11.	Grid search using cross-validation
RF [LR(11_inputs), ARIMA(3,1,1), MLP(5_inputs), GBT(11_inputs)]	The number of predictors chosen at random: 3 from 4, number of decision trees: 1–400, selected: 243. Stop parameters: maximum number of levels in each decision tree: 10, minimum number of data points placed in a node before the node is split: 200, min number of data points allowed in a leaf node: 5, maximum number of nodes: 100.	Grid search using cross-validation

## References

1. *Microgrids 1: Engineering, Economics, & Experience—Capabilities, Benefits, Business Opportunities, and Examples—Microgrids Evolution Roadmap*; Study Committee: C6, CIGRÉ Working Group C6.22, Technical Brochure 635; CIGRE: Paris, France, 2015.
2. Marnay, C.; Chatzivasileiadis, S.; Abbey, C.; Iravani, R.; Joos, G.; Lombardi, P.; Mancarella, P.; von Appen, J. Microgrid evolution roadmap. Engineering, economics, and experience. In Proceedings of the 2015 Int. Symp. on Smart Electric Distribution Systems and Technologies (EDST15); CIGRE SC C6 Colloquium, Vienna, Austria, 8–11 September 2015.
3. Hatziargyriou, N.D. (Ed.) *Microgrids: Architectures and Control*; Wiley-IEEE Press: Hoboken, NJ, USA, 2014.
4. *Low Voltage Microgrids. Joint Publication Edited by Mirosław Parol*; Publishing House of the Warsaw University of Technology: Warsaw, Poland, 2013. (In Polish)
5. Li, Y.; Nejabatkhah, F. Overview of control, integration and energy management of microgrids. *J. Mod. Power Syst. Clean Energy* **2014**, *2*, 212–222. [[CrossRef](#)]
6. Olivares, D.E.; Canizares, C.A.; Kazerani, M. A Centralized Energy Management System for Isolated Microgrids. *IEEE Trans. Smart Grid* **2014**, *5*, 1864–1875. [[CrossRef](#)]
7. Han, Y.; Zhang, K.; Li, H.; Coelho, E.A.A.; Guerrero, J.M. MAS-Based Distributed Coordinated Control and Optimization in Microgrid and Microgrid Clusters: A Comprehensive Overview. *IEEE Trans. Power Electron.* **2018**, *33*, 6488–6508. [[CrossRef](#)]
8. Morstyn, T.; Hredzak, B.; Agelidis, V.G. Control strategies for microgrids with distributed energy storage systems: An overview. *IEEE Trans. Smart Grid* **2018**, *9*, 3652–3666. [[CrossRef](#)]
9. Li, Q.; Peng, C.; Chen, M.; Chen, F.; Kang, W.; Guerrero, J.M.; Abbott, D. Networked and Distributed Control Method With Optimal Power Dispatch for Islanded Microgrids. *IEEE Trans. Ind. Electron.* **2016**, *64*, 493–504. [[CrossRef](#)]
10. Parol, R.; Parol, M.; Rokicki, Ł. Implementation issues concerning optimal operation control algorithms in low voltage microgrids. In Proceedings of the 5th Int. Symp. on Electrical and Electronics Engineering (ISEEE-2017), Galati, Romania, 20–22 October 2017; pp. 1–7.
11. Lopes, J.A.P.; Moreira, C.L.; Madureira, A.G. Defining Control Strategies for MicroGrids Islanded Operation. *IEEE Trans. Power Syst.* **2006**, *21*, 916–924. [[CrossRef](#)]
12. Zhao, H.; Wu, Q.; Wang, C.; Cheng, L.; Rasmussen, C.N. Fuzzy logic based coordinated control of battery energy storage system and dispatchable distributed generation for microgrid. *J. Mod. Power Syst. Clean Energy* **2015**, *3*, 422–428. [[CrossRef](#)]
13. Parol, M.; Rokicki, Ł.; Parol, R. Towards optimal operation control in rural low voltage microgrids. *Bull. Pol. Ac. Tech.* **2019**, *67*, 799–812.
14. Parol, M.; Kapler, P.; Marzecki, J.; Parol, R.; Polecki, M.; Rokicki, Ł. Effective approach to distributed optimal operation control in rural low voltage microgrids. *Bull. Pol. Ac. Tech.* **2020**, *68*, 661–678.
15. Niu, D.; Pu, D.; Dai, S. Ultra-Short-Term Wind-Power Forecasting Based on the Weighted Random Forest Optimized by the Niche Immune Lion Algorithm. *Energies* **2018**, *11*, 1098. [[CrossRef](#)]
16. Bokde, N.; Feijóo, A.; Villanueva, D.; Kulat, K. A Review on Hybrid Empirical Mode Decomposition Models for Wind Speed and Wind Power Prediction. *Energies* **2019**, *12*, 254. [[CrossRef](#)]

17. Adnan, R.M.; Liang, Z.; Yuan, X.; Kisi, O.; Akhlaq, M.; Li, B. Comparison of LSSVR, M5RT, NF-GP, and NF-SC Models for Predictions of Hourly Wind Speed and Wind Power Based on Cross-Validation. *Energies* **2019**, *12*, 329. [[CrossRef](#)]
18. Würth, I.; Valdecabres, L.; Simon, E.; Möhrlein, C.; Uzunoğlu, B.; Gilbert, C.; Giebel, G.; Schlipf, D.; Kaifel, A. Minute-Scale Forecasting of Wind Power—Results from the Collaborative Workshop of IEA Wind Task 32 and 36. *Energies* **2019**, *12*, 712. [[CrossRef](#)]
19. Liu, F.; Li, R.; Dreglea, A. Wind Speed and Power Ultra Short-Term Robust Forecasting Based on Takagi–Sugeno Fuzzy Model. *Energies* **2019**, *12*, 3551. [[CrossRef](#)]
20. Sun, S.; Fu, J.; Li, A. A Compound Wind Power Forecasting Strategy Based on Clustering, Two-Stage Decomposition, Parameter Optimization, and Optimal Combination of Multiple Machine Learning Approaches. *Energies* **2019**, *12*, 3586. [[CrossRef](#)]
21. Son, N.; Yang, S.; Na, J. Hybrid Forecasting Model for Short-Term Wind Power Prediction Using Modified Long Short-Term Memory. *Energies* **2019**, *12*, 3901. [[CrossRef](#)]
22. Zheng, H.; Wu, Y. A XGBoost Model with Weather Similarity Analysis and Feature Engineering for Short-Term Wind Power Forecasting. *Appl. Sci.* **2019**, *9*, 3019. [[CrossRef](#)]
23. Mujeeb, S.; Alghamdi, T.A.; Ullah, S.; Fatima, A.; Javaid, N.; Saba, T. Exploiting Deep Learning for Wind Power Forecasting Based on Big Data Analytics. *Appl. Sci.* **2019**, *9*, 4417. [[CrossRef](#)]
24. Wang, K.; Niu, D.; Sun, L.; Zhen, H.; Liu, J.; De, G.; Xu, X. Wind Power Short-Term Forecasting Hybrid Model Based on CEEMD-SE Method. *Process.* **2019**, *7*, 843. [[CrossRef](#)]
25. Valdecabres, L.; Nygaard, N.G.; Vera-Tudela, L.; von Bremen, L.; Kühn, M. On the Use of Dual-Doppler Radar Measurements for Very Short-Term Wind Power Forecasts. *Remote Sens.* **2018**, *10*, 1701. [[CrossRef](#)]
26. Zhao, H.; Zhao, H.; Guo, S. Short-Term Wind Electric Power Forecasting Using a Novel Multi-Stage Intelligent Algorithm. *Sustainability* **2018**, *10*, 881. [[CrossRef](#)]
27. Zhou, J.; Yu, X.; Jin, B. Short-Term Wind Power Forecasting: A New Hybrid Model Combined Extreme-Point Symmetric Mode Decomposition, Extreme Learning Machine and Particle Swarm Optimization. *Sustainability* **2018**, *10*, 3202. [[CrossRef](#)]
28. Fu, C.; Li, G.Q.; Lin, K.P.; Zhang, H.J. Short-Term Wind Power Prediction Based on Improved Chicken Algorithm Optimization Support Vector Machine. *Sustainability* **2019**, *11*, 512. [[CrossRef](#)]
29. Meng, X.; Wang, R.; Zhang, X.; Wang, M.; Ma, H.; Wang, Z. Hybrid Neural Network Based on GRU with Uncertain Factors for Forecasting Ultra-short-term Wind Power. In Proceedings of the 2020 2nd International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China, 23–25 October 2020.
30. Carpinone, A.; Giorgio, M.; Langella, R.; Testa, A. Markov chain modeling for very-short-term wind power forecasting. *Electr. Power Syst. Res.* **2015**, *122*, 152–158. [[CrossRef](#)]
31. Tato, J.H.; Brito, M.C. Using Smart Persistence and Random Forests to Predict Photovoltaic Energy Production. *Energies* **2018**, *12*, 100. [[CrossRef](#)]
32. Zhu, R.; Guo, W.; Gong, X. Short-Term Photovoltaic Power Output Prediction Based on k-Fold Cross-Validation and an Ensemble Model. *Energies* **2019**, *12*, 1220.
33. Xie, T.; Zhang, G.; Liu, H.; Liu, F.; Du, P. A Hybrid Forecasting Method for Solar Output Power Based on Variational Mode Decomposition, Deep Belief Networks and Auto-Regressive Moving Average. *Appl. Sci.* **2018**, *8*, 1901. [[CrossRef](#)]
34. Abdullah, N.A.; Rahim, N.A.; Gan, C.K.; Adzman, N.N. Forecasting Solar Power Using Hybrid Firefly and Particle Swarm Optimization (HFPSO) for Optimizing the Parameters in a Wavelet Transform-Adaptive Neuro Fuzzy Inference System (WT-ANFIS). *Appl. Sci.* **2019**, *9*, 3214. [[CrossRef](#)]
35. Nespoli, A.; Mussetta, M.; Ogliari, E.; Leva, S.; Fernández-Ramírez, L.; García-Triviño, P. Robust 24 Hours ahead Forecast in a Microgrid: A Real Case Study. *Electronics* **2019**, *8*, 1434. [[CrossRef](#)]
36. Huang, J.; Boland, J. Performance Analysis for One-Step-Ahead Forecasting of Hybrid Solar and Wind Energy on Short Time Scales. *Energies* **2018**, *11*, 1119. [[CrossRef](#)]
37. Dudek, G. Short-Term Load Forecasting Using Random Forests. In *Intelligent Systems'2014. Advances in Intelligent Systems and Computing*; Filev, D., Ed.; Springer: Berlin/Heidelberg, Germany, 2015; Volume 323, pp. 821–828.
38. Kim, C.; Yu, I.; Song, Y.H. Kohonen neural network and wavelet transform based approach to short-term load forecasting. *Electr. Power Syst. Res.* **2002**, *63*, 169–176. [[CrossRef](#)]
39. Dudek, G. Pattern-based local linear regression models for short-term load forecasting. *Electr. Power Syst. Res.* **2016**, *130*, 139–147. [[CrossRef](#)]
40. Shamsollahi, P.; Cheung, K.W.; Chen, Q.; Germain, E.H. A neural network based very short term load forecaster for the interim ISO New England electricity market system. In Proceedings of the 2001 Power Industry Computer Applications Conf., Sydney, Australia, 20–24 May 2001; pp. 217–222.
41. Parol, M.; Piotrowski, P. Very short-term load forecasting for optimum control in microgrids. In Proceedings of the 2nd International Youth Conference on Energetics (IYCE 2009), Budapest, Hungary, 4–6 June 2009; pp. 1–6.
42. Parol, M.; Piotrowski, P. Electrical energy demand forecasting for 15 minutes forward for needs of control in low voltage electrical networks with installed sources of distributed generation. *Przegląd Elektrotechniczny (Electr. Rev.)* **2010**, *86*, 303–309. (In Polish)
43. Hernandez, L.; Baladron, C.; Aguiar, J.M.; Carro, B.; Sanchez-Esguevillas, A.J.; Lloret, J.; Massana, J. A Survey on Electric Power Demand Forecasting: Future Trends in Smart Grids, Microgrids and Smart Buildings. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1460–1495. [[CrossRef](#)]

44. Parol, M.; Piotrowski, P.; Piotrowski, M. Very short-term forecasting of power demand of big dynamics objects. *E3S Web Conf.* **2019**, *84*, 01007. [[CrossRef](#)]
45. Ziegel, E.R.; Box, G.; Jenkins, G.; Reinsel, G. Time Series Analysis, Forecasting, and Control. *Technometrics* **1995**, *37*, 238. [[CrossRef](#)]
46. Brockwell, P.J.; Davis, R.A. *Introduction to Time Series and Forecasting*; Springer: Berlin/Heidelberg, Germany, 2016.
47. Shumway, R.H.; Stoffer, D.S. *Time Series Analysis and Its Applications: With R Examples*, 4th ed.; Springer: Berlin/Heidelberg, Germany, 2017.
48. Rojas, I.; Pomares, H.; Valenzuela, O. (Eds.) *Time Series Analysis and Forecasting. Selected Contributions from ITISE 2017*; Springer: Berlin/Heidelberg, Germany, 2018.
49. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2019.
50. Osowski, S.; Siwek, K. Local dynamic integration of ensemble in prediction of time series. *Bull. Pol. Acad. Tech.* **2019**, *67*, 517–525.
51. Piotrowski, P.; Baczyński, D.; Kopyt, M.; Szafranek, K.; Helt, P.; Gulczyński, T. Analysis of forecasted meteorological data (NWP) for efficient spatial forecasting of wind power generation. *Electr. Power Syst. Res.* **2019**, *175*, 105891. [[CrossRef](#)]
52. Dudek, G. Multilayer perceptron for short-term load forecasting: From global to local approach. *Neural Comput. Appl.* **2019**, *32*, 3695–3707. [[CrossRef](#)]
53. Karnik, N.; Mendel, J.; Liang, Q. Type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **1999**, *7*, 643–658. [[CrossRef](#)]
54. Karnik, N.N.; Mendel, J.M. Centroid of a type-2 fuzzy set. *Inf. Sci.* **2001**, *132*, 195–220. [[CrossRef](#)]
55. Mendel, J.; John, R. Type-2 fuzzy sets made simple. *IEEE Trans. Fuzzy Syst.* **2002**, *10*, 117–127. [[CrossRef](#)]
56. Khosravi, A.; Nahavandi, S.; Creighton, D.; Srinivasan, D. Interval Type-2 Fuzzy Logic Systems for Load Forecasting: A Comparative Study. *IEEE Trans. Power Syst.* **2012**, *27*, 1274–1282. [[CrossRef](#)]
57. Liang, Q.; Mendel, J.M. Interval type-2 fuzzy logic systems: Theory and design. *IEEE Trans. Fuzzy Syst.* **2000**, *8*, 535–550. [[CrossRef](#)]
58. Kam, H.T. The Random Subspace Method for Constructing Decision Forests. *IEEE Trans. Pattern Anal.* **1998**, *20*, 832–844.
59. Berk, R.A. *Statistical Learning from a Regression Perspective*, 3rd ed.; Springer: Berlin/Heidelberg, Germany, 2020.
60. Abbasi, R.A.; Javaid, N.; Ghuman, M.N.J.; Khan, Z.A.; Ur Rehman, S. Amanullah Short Term Load Forecasting Using XGBoost. In *Web, Artificial Intelligence and Network Applications. WAINA 2019. Advances in Intelligent Systems and Computing*; Barolli, L., Takizawa, M., Xhafa, F., Enokido, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2019; Volume 927.
61. Wang, W.; Shi, Y.; Lyu, G.; Deng, W. Electricity Consumption Prediction Using XGBoost Based on Discrete Wavelet Transform. In *Proceedings of the 2017 2nd International Conference on Artificial Intelligence and Engineering Applications (AIEA 2017)*, Guilin, China, 23–24 September 2017; pp. 716–729.
62. Friedman, J.H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [[CrossRef](#)]
63. Hastie, T.; Tibshirani, R.; Friedman, J.H. *The elements of Statistical Learning. Data Mining, Inference, and Prediction*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2009.
64. Alpaydin, E. *Introduction to Machine Learning*, 4th ed.; MIT Press: Cambridge, MA, USA, 2020.