https://www.sworldjournal.com/index.php/swj/article/view/swj35-03-015
DOI: 10.30888/2663-5712.2026-35-03-015

UDC 004.9

COMPARATIVE ANALYSIS OF MODERN TIME SERIES FORECASTING METHODS FOR SOLVING PRACTICAL PROBLEMS

Sumets S.I.
student of the Artificial Department,
ORCID: 0009-0005-4348-8848
Kharkiv National University of Radio Electronics, Kharkiv, Nauky Ave. 14, 61166
Udovenko S.G.
d.t.s., prof.
ORCID: 0000-0001-5945-8647
Simon Kuznets Kharkiv National University of Economics, Kharkiv, Nauky Ave. 9-A, 61165
Shergin V.L.
c.t.s., as.prof.
ORCID: 0000-0002-4388-8180
Kharkiv National University of Radio Electronics, Kharkiv, Nauky Ave. 14, 61166
Chala L.E.
c.t.s., as.prof.
ORCID:https://orcid.org/0000-0002-9890-4790
Kharkiv National University of Radio Electronics, Kharkiv, Nauky Ave. 14, 61166

Abstract. *The goal of this work is to develop a modified technology for error detection in text documents using a multilayer perceptron neural network.*

Each of the considered forecasting models has its own advantages and limitations when using them for the selected dataset. The most effective models according to the experimental results were SARIMA, SARIMAX, and Prophet, which proved their effectiveness in forecasting time series. Given the success of these three methods individually, it should be expected that their combination could lead to improved accuracy and reliability of forecasts.

Keywords: *machine learning, model, neural networks, forecasting, time series, ARIMA, ARIMAX, DEEPAR, PROPHET, SARIMA, SARIMAX*

Introduction.

With the advent of new technologies and the development of the information society, the problem of improving the quality of forecasting future events in situations of ambiguity is gaining particular importance. The forecasting procedure involves the analysis of previously collected data using various methods (statistical models, machine learning, etc.) and can be used in such areas as medicine, industrial technologies, economics. In this case, it is worth considering potential dangers and difficulties, as well as using modern technologies to increase the accuracy and efficiency of forecasts.

Let us consider some of the features of choosing time series forecasting methods. First of all, the regularity and frequency of data measurements should be taken into account in the process of forming forecasts. Some methods, such as ARIMA, are effective for series with a fixed measurement interval, while others, such as methods based on neural networks, can adapt to different frequencies. If the data contain trends or seasonal variations, then methods that can take these features into account should be considered. SARIMA models or LSTM-type neural networks can be effective for modeling such structures. Autocorrelation analysis (ACF) and autocorrelation of the displacement (PACF) can be useful for choosing the most appropriate forecasting method. ARIMA models, for example, work well with autocorrelated data, while exponential smoothing is effective for reducing the effect of seasonality [1].

The size of the available dataset for forecasting also matters. Some methods may be more effective for large amounts of data, while others may produce acceptable results even with limited data. In particular, deep neural networks (such as recurrent neural networks (RNNs) or long-term short-term memory (LSTMs)) can be successfully used for current big data processing. These models can effectively account for complex relationships in time series due to their ability to automatically detect patterns and trends. If the data contains anomalies or unexpected events, then the forecasting models must be able to adapt to this. The speed of data processing also plays a significant role when choosing a method for time series forecasting. This is especially important if you need to obtain real-time forecasts or respond quickly to changes in the data [2].

It should be noted that deep neural networks, although powerful in modeling complex relationships in data, require significant computational resources and time for training and forecasting. The purpose of this work is a comparative analysis of modern time series forecasting methods for solving practical problems (in particular, in the field of power system management).

Main text.

Data selection for the research. For further analysis, we will use a dataset containing data on electricity consumption in Portugal. The dataset consists of measurements of total consumption every 15 minutes during the day (in kilowatt-hours, kW). The dataset does not contain missing values, which allows us to avoid additional data processing and focus directly on modeling and forecasting. Timestamps correspond to Portuguese time. Each column of the dataset corresponds to an electricity consumer. A common characteristic is that the dataset contains detailed data on electricity consumption at different enterprises or installations. This allows for analysis and forecasting of consumption for different scenarios and types of customers. A fragment of the dataset is shown in Fig. 1.

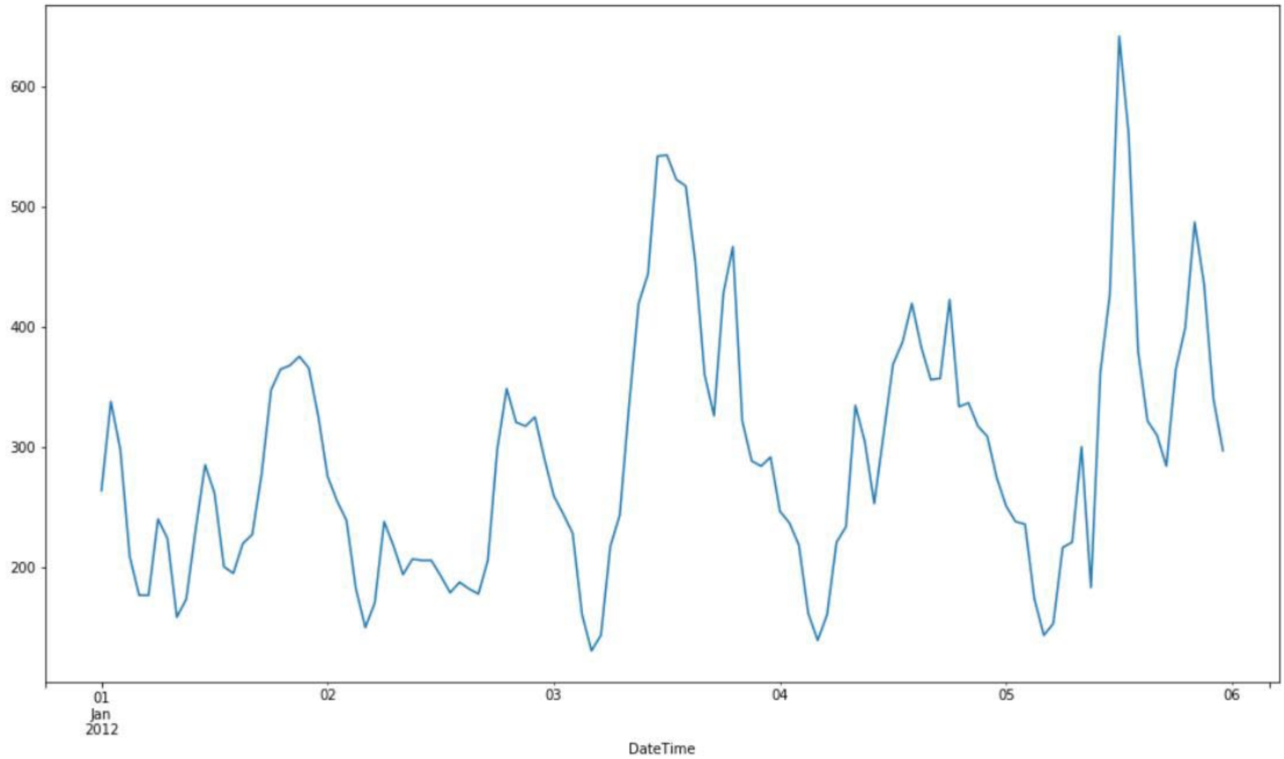

Figure 1 – Data fragment for research

Let us analyze the presence of such components in the time series under consideration as trend, seasonality, and residual component.

The trend shows the general tendency of the time series change over time. Seasonality reflects recurring patterns in the data that repeat after a fixed period of time. For example, seasonality can show peak consumption times during each day. The residual component reflects the deviation of the data from the trend and seasonality. It includes any random or non-periodic fluctuations that cannot be explained by trend and seasonality. The distribution of the analyzed series into components is shown in Fig. 2.

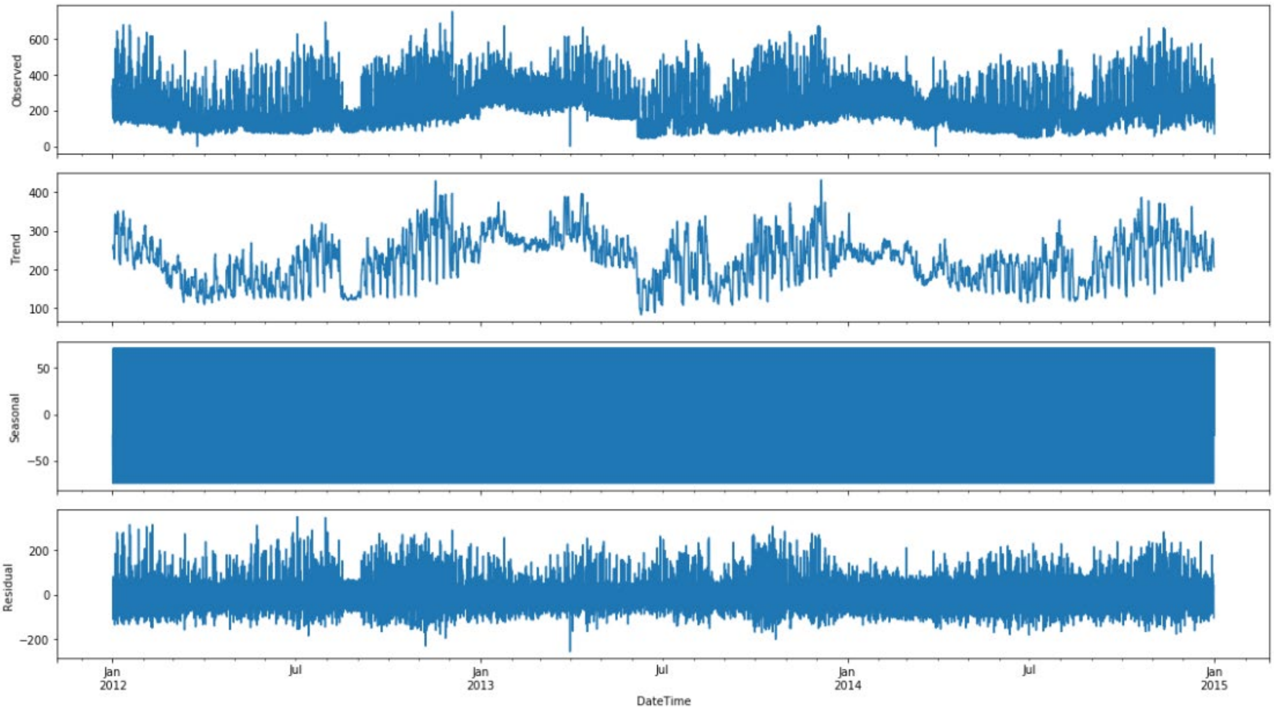

Figure 2 – Components of a time series

We will check the stationarity of the analyzed time series using the Dickey-Fuller statistical test (Fig. 3). The stationarity of a time series means that its statistical properties (in particular, the mean and variance) remain constant over time. According to the results obtained, the test statistic has a negative value, which indicates the stationarity of the time series and the absence of obvious trends or seasonality that require modeling.

```
Test Statistic                -1.202749e+01
p-value                        2.925873e-22
#Lags Used                     4.900000e+01
Number of Observations Used    2.625500e+04
Critical Value (10%)          -2.566829e+00
Critical Value (1%)           -3.430599e+00
Critical Value (5%)           -2.861650e+00
dtype: float64
```

Figure 3 – Dickey-Fuller test results

To identify correlations between the values of the time series, ACF and PACF graphs were constructed (Fig. 4), which allowed us to identify the lags of influence on the current values of the series for further modeling.

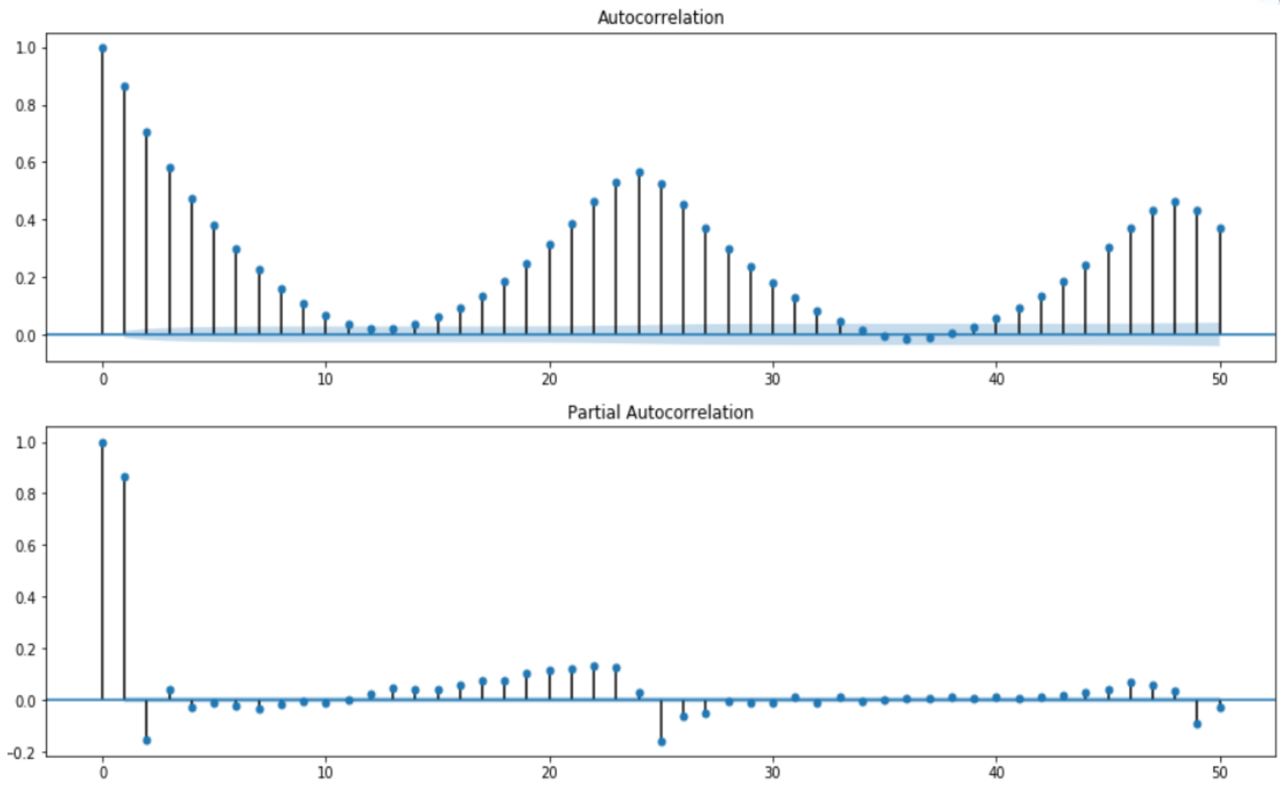

Figure 4 – ACF and PACF graphs

Model ARIMA. ARIMA (Autoregressive Integrated Moving Average) is a statistical model used for time series analysis and forecasting. It is based on three main components: autoregression (AR), integration (I), and moving averages (MA). The ARIMA model has three parameters (p, d, q) that define the model structure. By choosing the optimal values of these parameters, a more accurate time series forecast can be achieved. Parameter optimization avoids problems such as overfitting or underfitting the model. To build the model, the data was divided into training and test sets, and then different combinations of ARIMA parameters were searched. For each combination of parameters, the model was built on the training data and tested on the test set.

The forecast results were evaluated using the symmetric mean absolute percentage error (sMAPE) metric, which allows finding the model with the best predictive ability [3]:

$$sMAPE = n * 100\% \sum t = 1n(|At| + |Ft|)/2|Ft - At|, \quad (1)$$

where A_t is the actual value at time point t; F_t is the predicted value at time point t; n is the number of points in the test set.

The obtained optimal parameters are used to build the final ARIMA model, which is trained on the entire data set (training and test) and used to generate forecasts for the test period.

According to the optimization results, the best ARIMA model for a given time series is the model with parameters (p, d, q) = (3, 0, 2). Fig. 5 shows a graph showing the actual and predicted values of the time series.

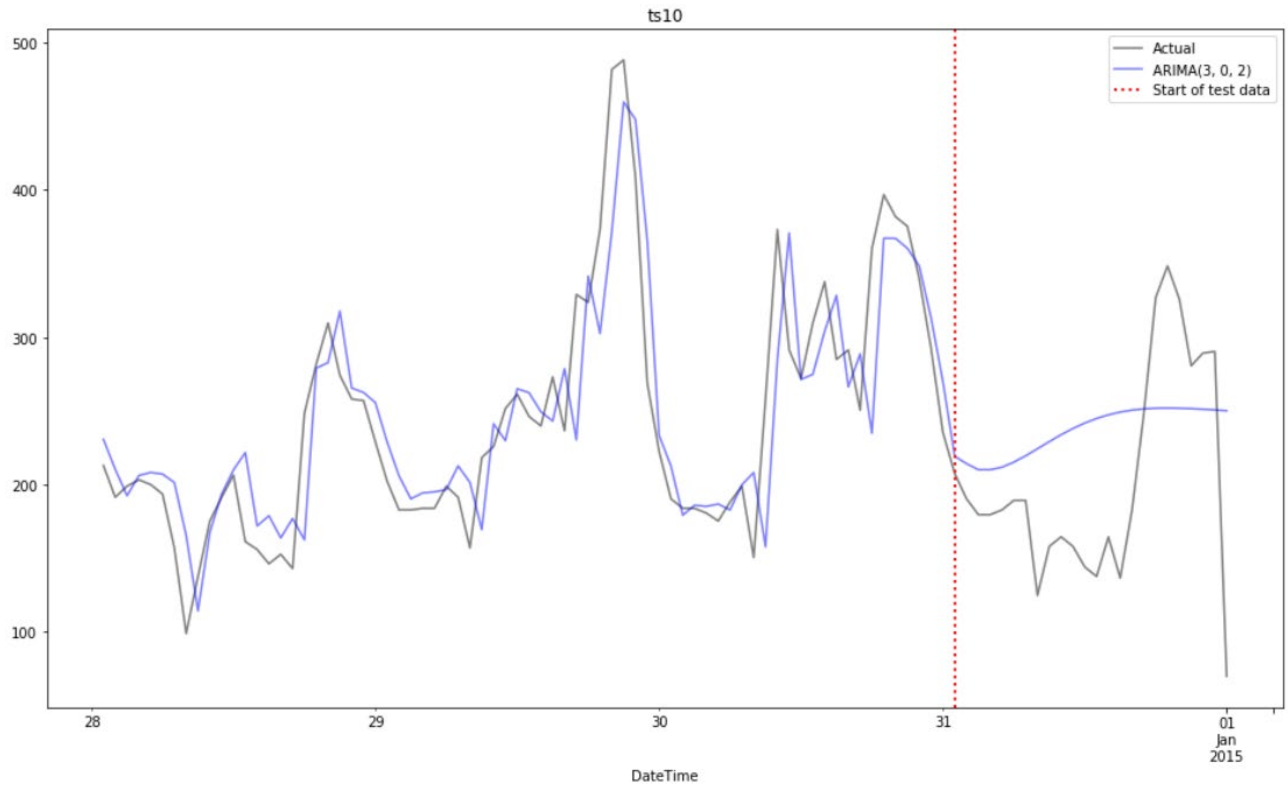

Figure 6 – Forecasting changes in a time series using the ARIMA model

Fig. 7 shows histograms of the distribution of sMASE metric values for a number of time series when using the constructed ARIMA statistical forecasting model.

The location of the peak point of the histogram indicates the average value of sMASE for the set of time series.

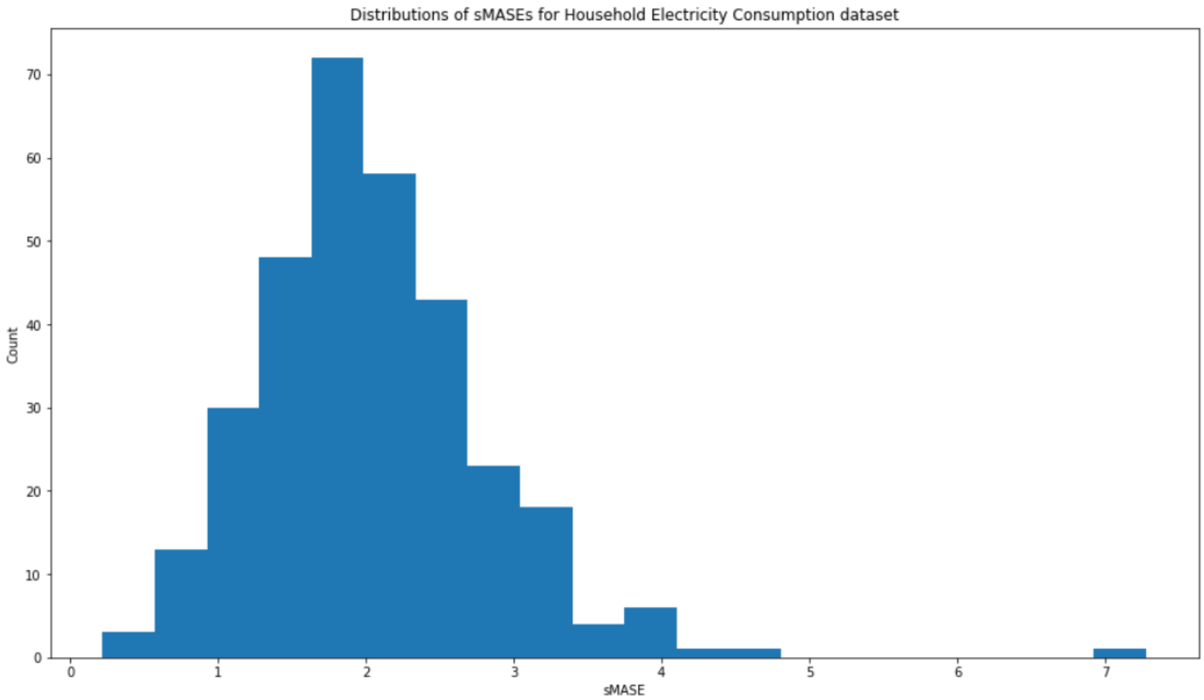

Figure 7 – Histograms of the distribution of sMASE metric values

ARIMAX, SARIMA, and SARIMAX models. ARIMAX is an extended version of the ARIMA model that allows the inclusion of exogenous variables and modeling of complex relationships between data series. This model has the following general form:

$$Y_t = c + \phi_1 Y_t - 1 + \cdots + \phi_p Y_t - p + \theta_1 \epsilon_t - 1 + \cdots + \theta_q \epsilon_t - q + \beta_1 X_{t1} + \cdots + \beta_k X_{tk} + \epsilon_t,$$

where Y_t is the value of the time series at time t; c is the bias; $\phi_1, \ldots, \phi_p$ are the parameters of autoregression (AR); $\theta_1, \ldots, \theta_q$ are the values of the time series at the previous time point; ϵ_t is white noise with zero mean and unit variance (N(0,1)); $X_{t1}, \ldots X_{tk}$ are exogenous variables; $\beta_1, \ldots, \beta_k$ are the parameters of exogenous variables; p is the order of autoregression (AR); q is the order of the moving average (MA).

SARIMA (Seasonal Autoregressive Integrated Moving Average), is an extension of the ARIMA model for analyzing and forecasting time series with seasonal components. For the SARIMA model, a fixed-parameter approach was used, namely SARIMA(1,1,1)(1,1,1)24. The main idea is that these parameters often provide reasonable accuracy for many time series. Optimizing the parameters can be time-consuming, and although this can improve the forecasts, it will also significantly increase the training time of the model.

After training and testing the SARIMA(1,1,1)(1,1,1)24 model, the AR, MA, and seasonal coefficients were estimated (Fig. 8).

```
                                     Statespace Model Results
==========================================================================================
Dep. Variable:                                 ts10   No. Observations:                672
Model:             SARIMAX(1, 1, 1)x(1, 1, 1, 24)   Log Likelihood             -3162.462
Date:                              Sun, 14 Oct 2018   AIC                         6334.924
Time:                                      10:42:02   BIC                         6357.080
Sample:                                  12-03-2014   HQIC                        6343.535
                                       - 12-31-2014
Covariance Type:                                opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.8293      0.019     44.082      0.000       0.792       0.866
ma.L1         -1.0000     25.031     -0.040      0.968     -50.059      48.059
ar.S.L24      -0.0522      0.020     -2.619      0.009      -0.091      -0.013
ma.S.L24      -0.9019      0.027    -33.982      0.000      -0.954      -0.850
sigma2      1458.2306   3.65e+04      0.040      0.968   -7.01e+04     7.3e+04
===================================================================================
Ljung-Box (Q):                       38.37   Jarque-Bera (JB):               329.17
Prob(Q):                              0.54   Prob(JB):                         0.00
Heteroskedasticity (H):               0.52   Skew:                             0.47
Prob(H) (two-sided):                  0.00   Kurtosis:                         6.44
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
Wall time: 33.3 s
```

Figure 8 – Quality assessments of SARIMA model forecasts

SARIMAX – The SARIMAX model (Seasonal Autoregressive Integrated Moving Average with eXogenous factors) is an extension of the SARIMA model, allowing for modeling and forecasting time series taking into account both endogenous (internal) variables and exogenous (external) factors. The main idea of the SARIMAX method is to take into account the influence of external variables, such as economic indicators, weather conditions, etc., that may affect the time series, along with the internal dynamic relationships already modeled by SARIMA.

The graph in Fig. 9 shows the actual data along with the predicted values for the SARIMAX(1,1,1)(1,1,1)24 model. The red vertical line represents the beginning of the test period.

Comparing this graph with similar graphs for the ARIMA, ARIMAX, and SARIMA models, we can identify similarities and differences in the forecasts. For example, the blue line representing the SARIMAX forecast may be close to the actual data if the model fits the time series well.

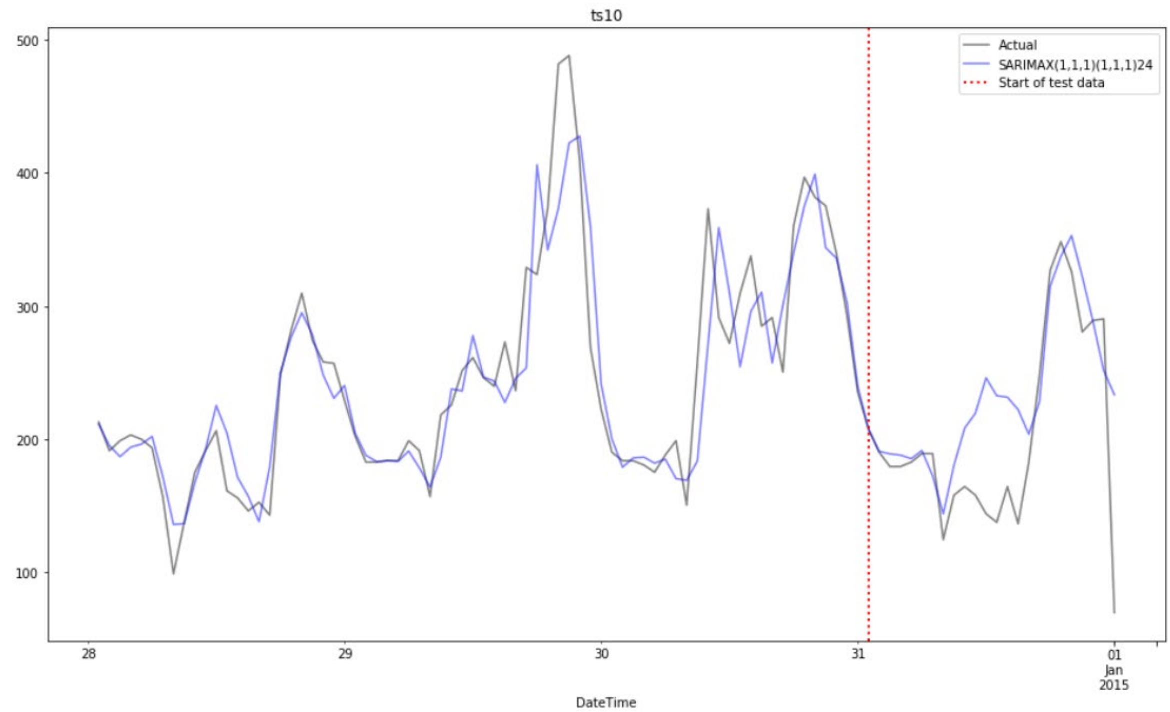

Figure 9 – Actual and forecasted time series values (SARIMAX model)

The SARIMAX model reproduces the actual dynamics of the time series quite accurately. It should be noted that the coincidence of the actual and predicted values is not always an indicator of high model quality. It is also necessary to evaluate other metrics, such as sMASE, and conduct comparative analysis with other models to determine the best option.

The DeepAR model. DeepAR is a time series forecasting model based on deep learning and available on the AWS SageMaker platform. The main idea of DeepAR is to teach the model to understand patterns and regularities in the time series and use this knowledge to predict future values. The DeepAR architecture uses recurrent neural networks (RNNs), in particular LSTM (Long Short-Term Memory) or GRU (Gated Recurrent Unit), to analyze the dynamics of time series. These types of neural networks are particularly effective for working with sequences, as they can remember previous states and take them into account when processing new input data [4]. For the research in this work, the DeepAR architecture was used in integration with the AWS SageMaker platform, which allows using the power of deep learning for time series forecasting in a convenient and efficient environment. Fig. 10 shows a graph showing the actual data (blue lines) and the data used to train the model (dashed black lines) for one of the time series from the dataset.

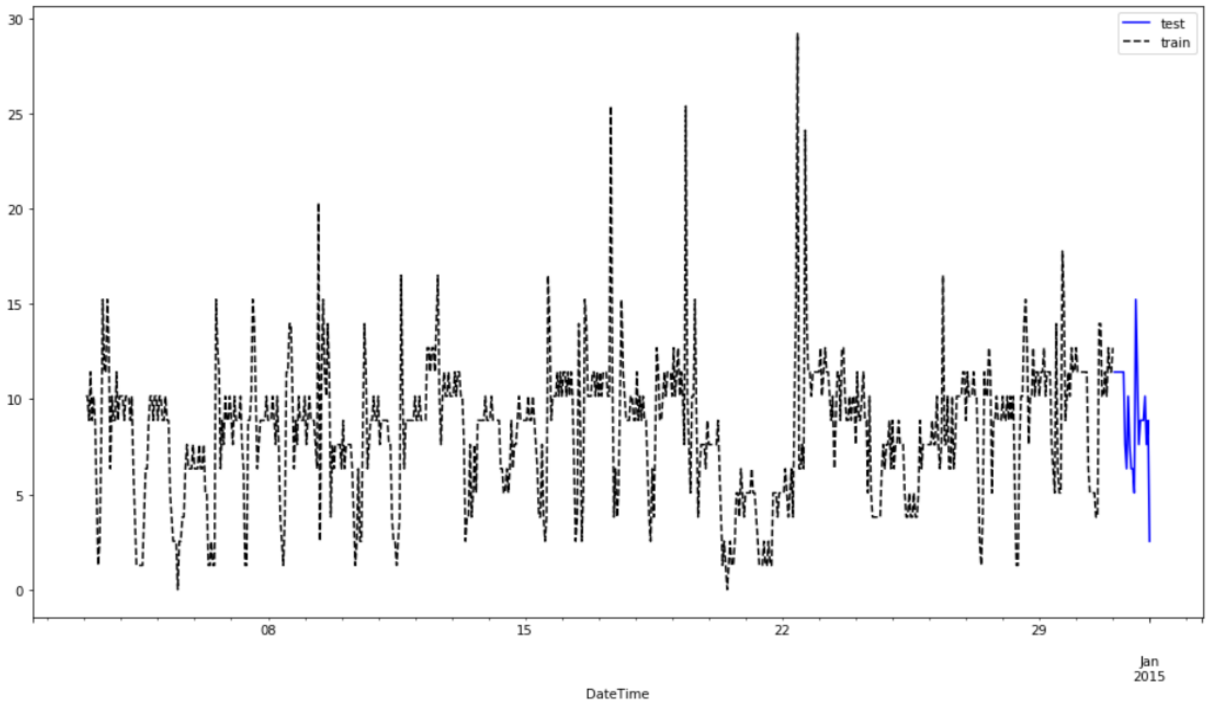

Figure 10 – Comparison of actual and training data for the DeepAR model

The blue lines represent the actual data for the period used to evaluate the model's forecasting accuracy. These data are typically the most recent values in the time series and are used to test how well the model predicts future values. The dashed black lines represent the data used to train the model. They are the portion of the historical time series that is used to train the model before predicting future values.

The Prophet model. The Prophet model is an additive model for automatic time series forecasting that includes three main components: trend, seasonality, and holiday effects. Holiday effects allow the model to account for the impact of holidays, events, or other important dates on the time series. According to this model, the predicted values of the time series are determined as follows:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \, , \quad (2)$$

where $y(t)$ is the value of the time series at time t; $g(t)$ is the trend that models the change in the mean value of the time series over time; $s(t)$ is seasonality, which represents recurring cyclical changes in time; $h(t)$ is holiday effects or other random changes; ϵ_t is the random component of the time series.

Figure 11 shows a graph showing the results of forecasting using the Prophet model for one of the time series in the dataset. The black dots on the graph are the actual data from the training set. The red dots are the actual data from the test set that the model did not use during training. The blue line on the graph is the Prophet model's prediction of the change in the time series values. It shows how the model predicts that the series values will change in the future. The light blue area is the confidence interval of the forecast (at the 80% level), within which the time series values can be found with some probability.

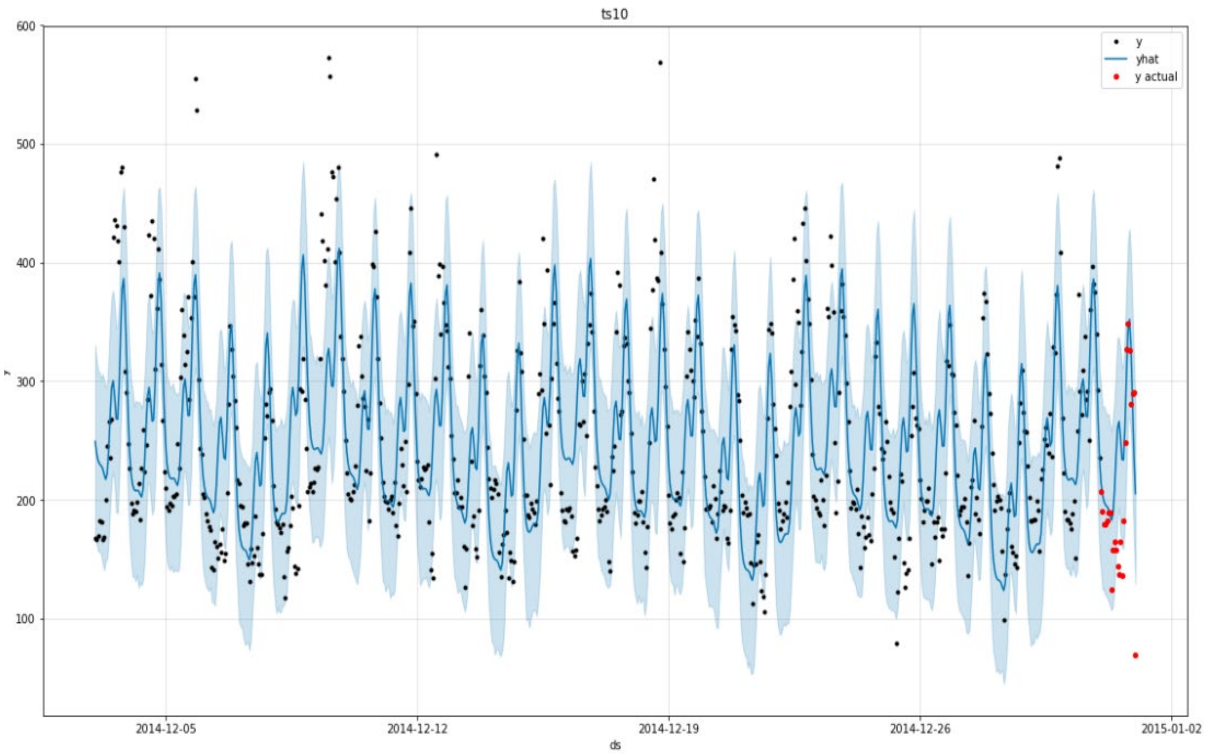

Figure 11 – Comparison of actual and predicted data for the Prophet model

The following graph (Fig. 12) shows the results of a similar forecast using a modified Prophet method that uses the concept of a saturated minimum value. The predicted values are shown as a blue line, and the actual values are marked as red dots.

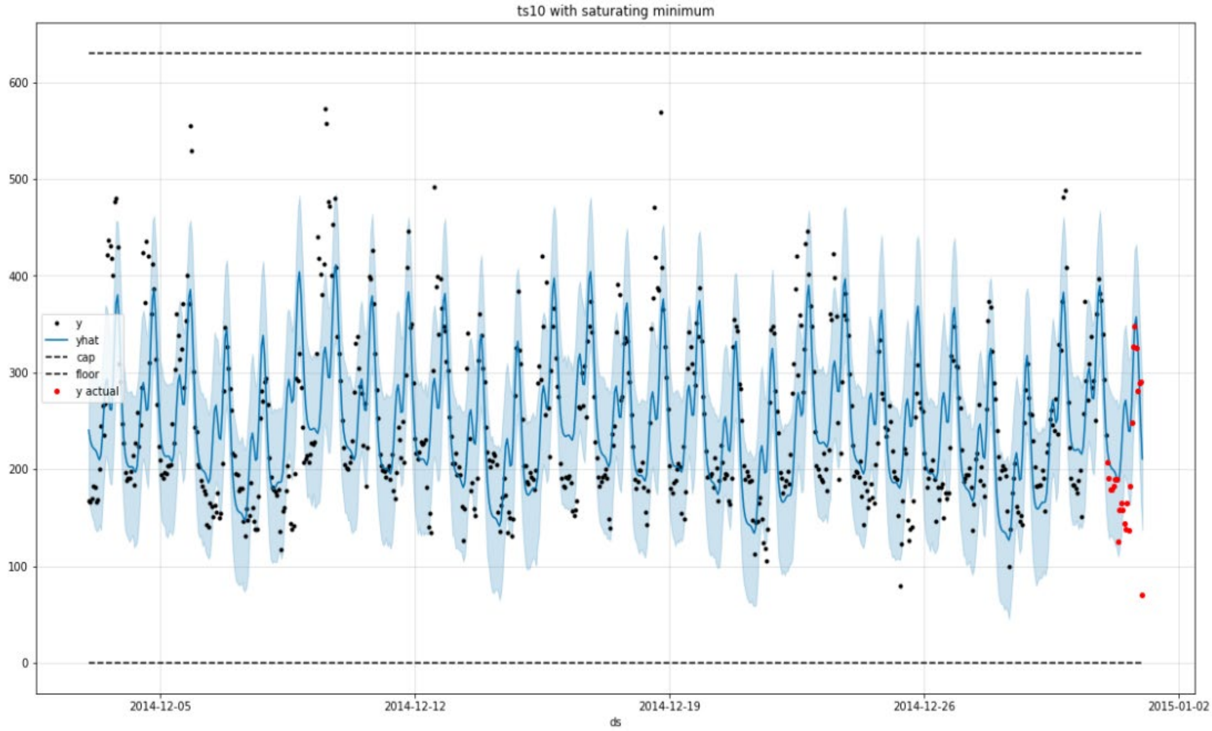

Figure 12 – Comparison of actual and predicted data for the modified Prophet model

Fig. 13 shows a graph of the predicted values obtained using the Prophet model with logarithmically transformed data. The black line shows the forecast, the blue color indicates the 80% confidence interval, and the red dots show the actual values of the time series.

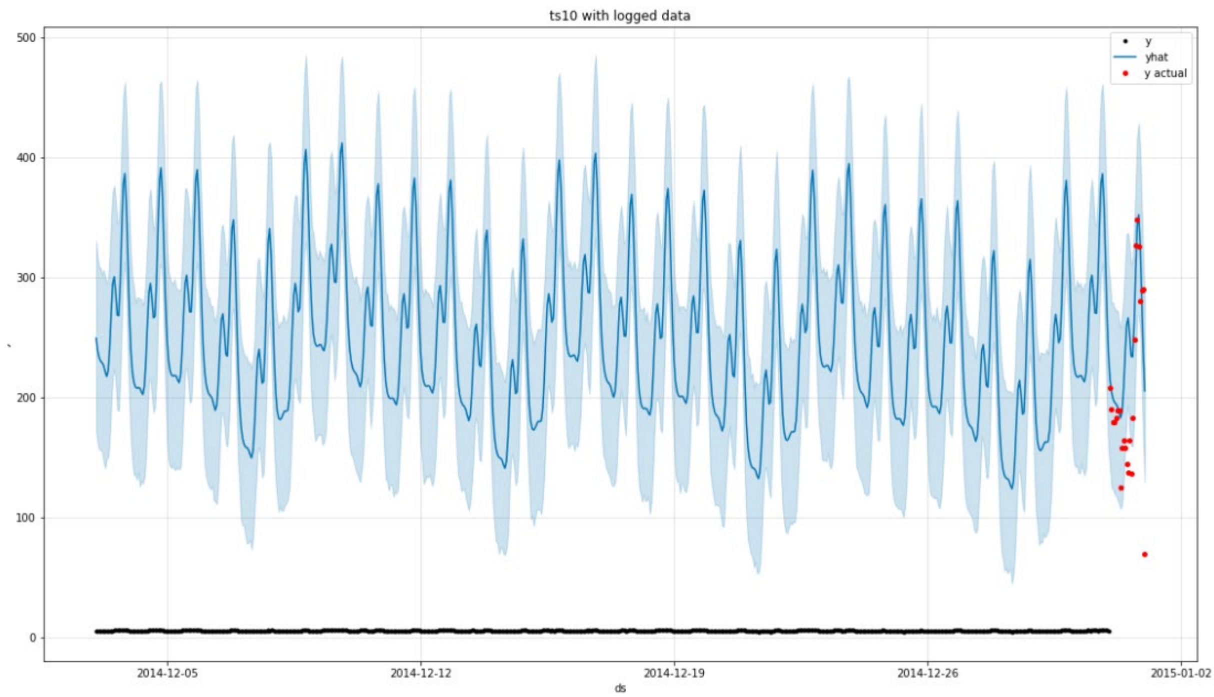

Figure 13 – Forecasting with logarithmically transformed data (Prophet model)

It is worth noting that the model stores logarithmic values for the training data, so the black dots on the graph do not show the actual values. It should be noted that the Prophet models show acceptable results according to the sMASE metric, which takes into account the quality of the forecasts compared to the actual data. The root mean square value of sMASE for the Prophet models is smaller than for the other models discussed above.

It should be noted that each of the considered forecasting models has its own advantages and limitations when used for the selected dataset. Table 1 shows some results of the corresponding comparative analysis.

Table 1 – Forecasting results for each method

Метод	Помилка прогнозування, %
SARIMA	1.60976
SARIMAX	1.61433
DeepAR	1.69218
Prophet	1.63378
ARIMA	2.04262
ARIMAX	2.01298

The most effective models according to the experimental results were SARIMA, SARIMAX and Prophet. SARIMA and SARIMAX are classic models that have already proven their effectiveness in time series forecasting. They are well suited for modeling and forecasting time series with various seasonal and trend components. Both methods have a fairly wide range of applications and can be successfully used in many situations. Prophet is a more modern model, characterized by the ability to automatically take into account seasonality, trend change and change in the subsystem.

Given the success of these three methods individually, it should be expected that their combination can lead to improved accuracy and reliability of forecasts [5]. In addition, each of these models has its own unique features that can complement each other in the ensemble, which makes them attractive choices for research.

Conclusions. The results showed that the considered forecasting methods have their advantages and disadvantages depending on the specific conditions and characteristics of the studied time series. For example, the SARIMA and SARIMAX methods are effective in the presence of seasonality and trends. The DeepAR method, based on deep learning, shows acceptable results in forecasting taking into account the complex dynamics of time series.

In the context of electricity forecasting, the use of ensemble models SARIMA, SARIMAX and Prophet can significantly improve forecasting results by combining the strengths of individual models and reducing the impact of their shortcomings. Overall, this study confirms the importance of careful analysis and selection of the optimal method for specific forecasting tasks (in particular, electricity consumption forecasting tasks).

References:

1. Kholiavka Y., & Parfenenko Y. (2023). Forecasting peak load on the power grid. Computer Systems and Information Technologies, (3), 12–22. https://doi.org/10.31891/csit-2023-3-2

2. Hovorushchenko T, Medzatyi D, Voichur Y, Lebiga M. (2023). Method for forecasting the level of software quality based on quality attributes. 2023, Journal of Intelligent & Fuzzy Systems, 1-15. https://doi.org/10.3233/JIFS-222394.

3. Hyndman R.J., & Athanasopoulos G. (2018) Forecasting: principles and practice, 2nd edition, Texts: Melbourne, Australia. URL: https://otexts.com/fpp2/expsmooth.html.

4. David S, Valentin F, Jan G. (2019). DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. arXiv 2019. arXiv https://doi.org/10.48550/arXiv.1704.04110.

5. Zahra F, Minh H, Elena Z, Zamir S, Xiaojun D. (2023). Mitigating Cold-start Forecasting using Cold Causal Demand Forecasting Model. arXiv 2023. https://doi.org/10.48550/arXiv.2306.09261.

Article sent: 20.12.2025