UDC 925.17

Modeling the energy consumption of smart buildings using artificial intelligence

Eugene Yu. Shchetinin

Department of Data Analysis, Decision Making and Financial Technologies
Financial University under the Government of the Russian Federation
Leningradsky pr. 49, Moscow, 117198, Russia

Email: riviera-molto@mail.ru

Intelligent energy saving and energy efficiency technologies are the modern large-scale global trend in the energy systems development. The demand for smart buildings is growing not only in the world, but also in Russia, especially in the market of construction and operation of large business centers, shopping centers and other business projects. Accurate cost estimates are important for promoting energy efficiency construction projects and demonstrating their economic attractiveness. The growing number of digital measurement infrastructure, used in commercial buildings, has led to increased availability of high-frequency data that can be used for anomaly detection and diagnostics of equipment, heating, ventilation, and optimization of air conditioning. This led to the use of modern and efficient machine learning methods that provide promising opportunities to obtain more accurate forecasts of energy consumption of the building, and thus increase energy efficiency. In this paper, based on the gradient boosting model, a method of modeling and forecasting the energy consumption of buildings is proposed, and computer algorithms are developed to implement it. Energy consumption dataset of 300 commercial buildings was used to assess the effectiveness of the proposed algorithm. Computer simulations showed that the use of these algorithms has increased the accuracy of the prediction of energy consumption in more than 80 percent of cases compared to other machine learning algorithms.

Key words and phrases: Energy consumption, smart buildings, smart meters, machine learning, random forest, gradient boosting.

In: K. E. Samouylov, L. A. Sevastianov, D. S. Kulyabov (eds.): Selected Papers of the IX Conference "Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems", Moscow, Russia, 19-Apr-2019, published at http://ceur-ws.org

1. Introduction

The most important direction of the world economic development is to improve the energy efficiency of the industrial and consumer sectors of the economy. The state program of the Russian Federation "Energy saving and increase of energy efficiency for the period up to 2030" was approved as one of the most important directions of development of the Russian economy. Several energy efficiency solutions have been implemented to reduce the environmental impact and costs of the commercial building sector. For example, long-term energy saving targets have been set at the state and federal levels in Russia, and these targets should be achieved with the help of energy efficiency programs. Measuring and verifying energy efficiency is a process of assessing savings and is therefore crucial to determining the value of energy efficiency for building owners, utility tariff consumers and service providers. Today, the growing availability of data from smart meters and devices, combined with data mining, allows the process to be optimized by increasing the level of automation while maintaining or improving the accuracy of the forecasting consumption. The development of intelligent networks in industry, finance and services creates new opportunities for the development and application of effective methods of machine learning and big data analysis [1, 2]. The introduction of smart meters provides benefits to end users, energy suppliers and network operators by providing consumers with near-real-time consumption information that will help them manage real energy consumption, save money and reduce greenhouse gas emissions. At the same time, smart meters facilitate the planning and operation of the distribution network as well as demand management. In this regard, smart metering data will allow more accurate forecasting of demand, increase the efficiency of the distribution networks, reduce the recovery time of supplies, as well as reduce operating costs of networks. Intelligent technologies for collecting, recording and monitoring energy consumption data create a wealth of data of different nature for use by energy providers and network operators. The amount of data varies depending on the number of smart meters installed, the number of smart meter messages received, the size of the message, and the frequency of measurement records, for example, every 15 or 30 minutes. These data can be used for optimal network management, improving the accuracy of load forecasting, identifying anomalous effects of electricity supply (peak load condi-tions), formation of flexible price tariffs for different groups of consumers. The basic models used in estimating the energy consumption profiles, are regres-sion models that link the consumption of energy in buildings with such parameters as, for example, the temperature of the external environment, humidity, characteristics of the building, etc. Traditionally, for building such models, the data of monthly bills for utilities are used. However, the increasing of data availability from smart meters with an hour and 15-minute intervals helped us to create new methods for more accurate energy consumption estimation and forecasting. In recent years, several approaches to energy consumption modeling using smart meter data have appeared. These methods are based on traditional linear regression, nonlinear regression, and machine learning methods. The linear regression model, described in paper [3], includes the time of day, the day of the week, and two temperature functions that allow different combinations of heating and cooling. It can also include humidity and holidays as variables. This model is well described by the usual regression, which is a common practice in such cases [7, 15]. Over the past two decades, significant progress has been made in the development of new methods of machine learning, among which the most promising in terms of prediction accuracy are the approaches of the family of ensemble learning algorithms. Ensemble methods construct a model by training several relatively simple models, and then combine them to create a more complex model with higher predictive properties. The most well-known ensemble learning algorithms are bagging, random forests and gradient boosting GBM [4, 5, 13]. The bagging, decision trees and random forests are based on a simple averaging of the base model, while boosting algorithms are built on a constructive iterative strategy. Although historically these ensemble machine learning algorithms have been used with great success in many fields, they are just beginning to be applied to problems in the field of the buildings energy consumption modeling.

In this paper, we presented a new computer algorithm for estimating and fine-tuning the parameters of the model of the energy consumption profile of a conglomerate of commercial buildings [11]. For this we used machine learning algorithms of ensembles, such as random forest and GBM gradient boosting [4, 5, 13] and compared them with the regression model. We also developed effective numerical algorithms for estimating and fine tuning the parameters of the GBM model and estimating the accuracy of energy consumption forecasting in the buildings.

2. Methods of energy consumption modeling using machine learning algorithms

The growing availability of data from smart meters in combination with data mining allows to optimize the process of power consumption by increasing the level of automation in storage and improving the accuracy of forecasting [1–3]. The main models used in the assessment of energy consumption profiles are the regression models that relate energy consumption in buildings to such parameters as ambient temperature, humidity, individual characteristics of the building, etc. [3, 7]. Data from monthly utility bills have traditionally been used to build such models, but the growth of up-to-date data from smart meters with hourly and 30-minute intervals allows new methods to be developed for more accurate forecasting. However, with the introduction of intelligent technologies for the collection, analysis and control of energy consumption data, significant progress has been made in the development of new methods using machine learning algorithms, among which the most promising in terms of prediction accuracy are the approaches of a family of ensembles algorithms. Ensemble methods create a model that trains several relatively simple models and then combines them to create more complex but with better predictive properties. Well-known learning algorithms used for this purpose, such as regression trees, random forests [4], bagging [5], and boosting [13]. Although these machine learning algorithms are used with great success in many areas, they are only beginning to be applied to energy saving modeling problems. For example, in papers [8, 12, 15] the authors used a random forest to predict the hourly energy consumption, in [19] the random forest algorithm was applied to detect anomalies in the energy consumption of the building. In this paper we propose a new algorithm for selecting and fine - tuning the hyperparameters of the energy consumption model and tested it on the example of conglomerate of the commercial buildings. To solve this task, we used the GBM gradient boosting algorithm and developed algorithms to estimate the accuracy of energy consumption forecasting. Using the decision trees as a regression method has several advantages, one of which is that the splitting rules are an intuitive and simple graphical way to visualize the results. In addition, by their design, they can simultaneously process numerical, categorical and other types of input parameters. They are also robust to outliers and can effectively handle missing data in the input parameter space [18, 19]. The hierarchical structure of the decision tree automatically models the interaction between input parameters and selects a variable, for example, if the input parameter is not used at all during the split procedure, the prediction is independent of that input parameter. Finally, decision tree algorithms are easy to implement and computationally efficient with large amounts of data [9, 10]. The GBM algorithm was first proposed for classification problems [13]. Its basic principle is that several simple models, called "weak learning models", should be combined into a iterative scheme to select parameters in order to obtain a so-called strong learning model", that is, model with better forecast accuracy. Thus, the GBM algorithm iteratively adds a new decision tree at each step in a way that best one reduces the loss function. The algorithm continues to run until the maximum number of iterations is reached or the specified precision is reached. This means that at each new step, the decision trees added to the model in the previous steps are fixed. Thus, the model can be improved in those parts of it where it still does not evaluate residuals well enough. The GBM algorithm is more efficient, if at each iteration the contribution of the added decision tree is taken into account using a certain hyperparameter, that describes one of the important characteristics of the algorithm, namely, the model learning rate ν. One of the problems in choosing the hyperparameter ν is that in order to achieve the required accuracy ϵ, depending on the value of ν, an appropriate number of iterations m is necessary. Namely, the smaller the value of ν, the greater the number of iterations m necessary to perform. Thus, it is necessary to develop a numerical procedure for the optimal choice of hyperparameters ν, m of GBM model. Another problem is the presence of autocorrelation in researched datasets, which, as is known, introduces additional distortions in the model parameter estimates [10, 16]. The solution to the above problems in our opinion is the use of randomization in the process of constructing the GBM algorithm. At each iteration, instead of the entire sample, a subsample, randomly extracted from it, is used to evaluate the decision tree. In practice, however, it is hard to allocate a sufficient number of data points to accurately assess the predictive performance of the models without affecting the quality of the assessment. When the number of observations is not enough, reducing the size of the training set can lead to a significant decrease in accuracy [6, 9, 17]. Therefore, to take into account the influence of the subsample size on the quality of the model fit, it is necessary to use subsamples of different sizes k. In this situation, to solve the problems we have developed the numerical algorithm for selecting the optimal values of the hyperparameters of the GBM model using the$k$-fold cross validation and randomization procedure. The k-fold cross-validation method involves splitting the data set into k sub-samples of approximately the same size. First, the model is estimated using $(k - 1)$ folds as a training sample and the k-th fold (test sample) is used to determine the accuracy of the forecast. Secondly, the procedure is repeated k times, and each time a new block is used as a test sample. To estimate the accuracy of CV procedure, we use the root mean square error

$$RMSE_k = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \widetilde{y})^2}, \tag{1}$$

where y_i is the value of the real data set, $\widetilde{y}$ is the predicted value, N- the number of measurements corresponding to the size of the k-th subsample. The k- fold cross-validation method uses the root mean square error as follows

$$RMSE(CV) = \frac{1}{k}\sum_{i=1}^{k}(RMSE_i).$$

Our GBM model has four hyperparameters that need to be configured: 1) d - the depth of the decision trees; 2)m - the number of iterations; 3)ν - the learning rate, which is usually accepts the values between 0 and 1; 4) k - the number of splits of sample for sub-samples that are used at each iteration step in cross-validation procedure.

The global problem of machine learning algorithms is their overfitting. This phenomenon is that it is usually a disadvantage of too complex model. In the case of the GBM model, this can happen if too many iterations m or too depth of decision trees with d parameter are selected. Thus, it is necessary to choose the optimal combination of hyperparameters to avoid overfitting and at the same time to ensure the best accuracy of the forecast. Most effective method to solve this problem is grid search method [14]. This approach consists of determining a grid of combinations of hyperparameter values, constructing a model for each combination, and selecting the optimal combination using indicators that determine the model in terms of prediction accuracy. In the k-fold cross-validation algorithm described above, we used a grid search method to achieve the optimal values of the hyperparameters of the model (1).

The pseudocode of the presented algorithm is as follows:

1. The user specifies the depth of the decision trees d, the number of iterations m, the learning rate value ν and the accuracy forecast ϵ;

2. Initial step equal $x_0 = \overline{y}$ and $\widetilde{f^0} = 0$, where $\overline{y}$ is an average value of y was suggested as the initial value for decision tree f;

3. For $j = 1, ..., m$ the following steps need to be completed:

a) Randomly select a subsample (x_i, y_i), $i = 1, ..., N$ from the training dataset , where N- the size of the subsample;

b) Build the decision tree $\widetilde{f}^j$ with d depth using the values (x_i, y_i) and considering the residuals z^j ;

c) Update the residuals $z^{j+1} = z^j + \nu \widetilde{f}^j$;

4. If $RMSE(CV) < \epsilon$ or $j = m$ then go to 5. Else go to 3;

5. End.

3. Computer experiments and analysis the results

To test the algorithms considered in the article, the data of power consumption indicators of the urban conglomerate of buildings in Dublin, Republic Ireland were used [11]. The data are 15-minute measurements of electricity consumption in (Kw) for the period 29.03.2011-20.02.2013. A typical graph of the studied time series is shown in Fig. 1. For each building, time series data were splitted into training and forecasting periods. The forecast period was defined as the different periods of the last 12 months of the explored data set. Models are trained using two different training periods, which are 6 and 12 months.

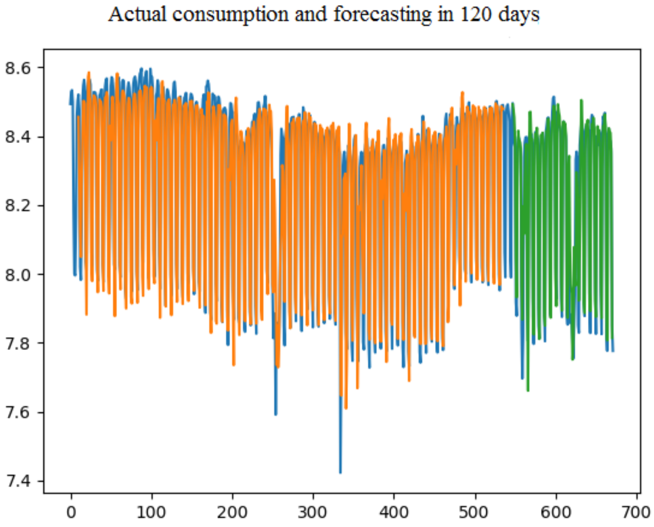

Figure 1. Energy consumption time series for one of the buildings. The measurements are indicated in a logarithmic scale and days.

As for the results of the evaluation of the accuracy with RMSE(CV), the GBM-1d and GBM-7d models are superior to the Random Forest and GBM models in both periods of training. The accuracy of the GBM models improved significantly when their training period was increased from 6 to 12 months, while the accuracy R^2 of the regression model decreased slightly. Recall that higher values are desirable for R^2, whereas for RMSE(CV) values it is desirable to have them close to zero. Table 1 shows the percentage of buildings for which GBM models seemed more accurate than regression and RF models, respectively. For RMSE(CV) the columns represent the percentage of buildings with a lower RMSE (CV) than the regression model. These results confirm that GBM-1d and GBM-7d algorithms have better accuracy than regression and RF models.

Table 1

Estimates of the accuracy of the energy consumption models in the test sample for 6 and 12 months.

Model	R^2 $6m$	RMSE(CV) 6m	R^2 $12m$	RMSE(CV) 12m
GBM	33	47	62	76
$GBM-1d$	57	63	67	81
$GBM-7d$	77	70	81	86
$RandomForest$	28	42	35	48
$Regression$	17	30	28	38

Hyperparameters of the gradient boosting model were tuned automatically with the a grid search algorithm and the various implementations of cross-validation method, mentioned above. The depth of regression trees d was chosen from the set $(3, \ldots, 10)$, the learning rate ν was chosen between 0.05 and 1, the number of iterations m have been taken the values from 1 to 1000 with steps of 10 iterations. Three variants of the standard CV were used: 5-fold CV, 5-fold CV and one day as a block (CV-1day), 5-fold CV with a week as a block for CV (CV-7day). Thus, taking into account the above proposed cross-validation algorithms with fine-tuning of hyperparameters, we have three different implementations of the gradient boosting model: 1) GBM model with the choice of parameters by the standard method 5-fold CV; 2) GBM model c CV-1day (GBM-1d; 3) GBM model with CV-7days (GBM-7d). The results of computer experiments showed that the value of the learning rate $\nu = 0.1$ is too low, because the algorithm turned out to be too sensitive to both the number of iterations and the depth of the decision trees. It was also found that at the learning rate $\nu = 0.2$ and the depth of the decision tree $d = 5$, the algorithm did not reach the optimal number of iterations at $m = 500$. When the learning rate was increased to $\nu = 1$ due to the need to increase the number of iterations and due to the increased complexity of the model, the algorithm began to overfitting. Thus, the optimal learning rate was assumed to be $\nu = 0.5$. The behavior of the learning error for different learning values of the GBM algorithm is shown in Fig. 2.

Accuracy estimation indicators such as the determination coefficient R^2 and square mean error CV(RMSE) were calculated for the entire data set. Their values demonstrated the accuracy decreasing while reducing the training period from 12 to 6 months. Computer experiments showed that the values of R^2 for GBM models exceeds the corresponding values of R^2 for Random Forests and regression models. This is especially noticeable for GBM-1d and GBM-7d models. As expected above, the GBM model, using the standard CV, has less accuracy than the other two versions, which use the CV in the form of k-fold blocks. With the reduction of the training period to 6 months in the standard GBM model, there was a slight decrease in R^2 and its slight decrease for GBM-1d, GBM-7d and RF models. However, the accuracy of the regression model did not improved, which means that for this dataset, the regression model does not improve accuracy as the number of observations increases.

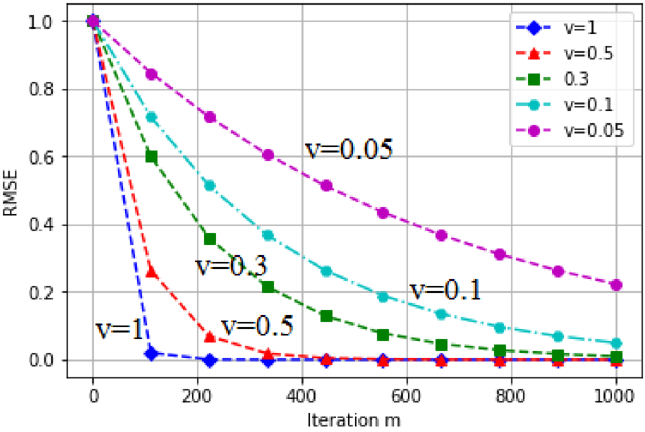

Figure 2. RMSE standard error for different values of the learning rate ν under the training GBM model.

As for the results of the estimation of the accuracy with RMSE(CV), the GBM-1d and GBM-7d models are superior to the Random Forest and GBM models in both periods of training. The accuracy of the GBM models improved significantly when their training period was increased from 6 to 12 months, while the accuracy R^2 of the regression model decreased slightly. It should be reminded that higher values are desirable for R^2, whereas for RMSE(CV) values are desirable to be close to zero. Table 1 shows the percentage of buildings for which GBM models seemed more accurate than regression and RF models, respectively. For RMSE(CV) the columns represent the percentage of buildings with a lower RMSE(CV) than the regression model. These results confirm that GBM-1d and GBM-7d algorithms have better accuracy than regression and RF models.

Based on the constructed models, modeling and forecasting of energy consump-tion were also carried out on the example of indicators of one of the buildings. The results of modeling and forecasting the energy consumption for 120 days using RGBM-7d algorithm are presented in Fig.1.

4. Discussion and conclusions

In this paper, we propose a method for estimating the consumption of electric power by large commercial centers and business buildings, based on the GBM gradient boosting algorithm with adaptive tuning of hyperparameters using the k-fold cross-validation and randomization procedures. To find the optimal values of the hyperparameters, a modern computer algorithm was developed using a grid search method for sets of values of the hyperparameters. The capacity and effectiveness of GBM in solving the problem of energy efficiency was tested on both model and real data of energy consumption from smart meters of building conglomerate. As a whole, the GBM model showed higher forecasting accuracy than the regression and random forest models for all tested training periods. The results of computer experiments showed that the use of the GBM model can improve the accuracy of energy efficiency assessment as a separate building and a complex of buildings as a whole. It also turned out that the use of a 6-month training period to build GBM models led to a slight decrease in the accuracy of the energy consumption forecast, compared to those obtained for the 12-month training period, which is usually used for the entire building. As we can see from Table 1, this result is not true for regression analysis models and for Random Forest algorithm. Similar conclusions also were made for a number of machine learning algorithms in papers [7,15]. Thus, the application of GBM algorithms allows not only to improve the accuracy of energy saving assessment in general, but also to reduce the total time required for energy saving assessment of the entire complex of buildings.

A comparison of different hyperparameter tuning algorithms showed the importance to take into account time series autocorrelation. Indeed, the results showed that the use of standard k-fold cross-validation reduces the accuracy of the GBM algorithm. The reason is that when using the standard approach of cross-validation the observations in test and training data sets are not independent (due to autocorrelation of measurements obtained from smart sensors), which leads to the model overfitting. In order to overcome the effect of autocorrelation, a randomization procedure was included in the algorithm for estimating the accuracy of the energy consumption forecast. It was also shown that the difference in use as a forecast period of one week or one day does not have a significant impact on the results of the assessment and forecast of energy consumption. Therefore, we can conclude that for most cases, using one day as a default block, when estimating the accuracy of the energy forecast, is a good choice. It is known that one of the main advantages of the ensemble models is its flexibility and reliability when used a large number of input parameters [18]. In addition, compared to models such as regression, there is no need to modify the algorithm to handle additional input parameters such as building occupancy, humidity or illumination. Instead, it is sufficient to include these variables in the input table of the algorithm without the need to determine the specific shape of the model for each of the parameters, as in the case with most of the standard regression algorithms used in modern practical applications. In addition, the ability to select GBM model variables allows to exclude parameters that do not affect the model, without reducing its predictability.

In conclusion we resulted that the GBM model has a number of obvious advantages over regression models, consisting in its ability to maintain accuracy for shorter training periods, improve overall accuracy with respect to energy efficiency indicators and ease of inclusion of additional explanatory variables. Key areas of future work will be the application of the GBM model to address energy efficiency issues [16, 17, 21], such as forecasting energy consumption in long-term load planning, continuous anomaly detection, and quantifying the reduction of demand-side load [19, 20]. Also, further development of the GBM model is aimed at expanding its capabilities in the direction of application of the methods of deep learning and neural networks.

5. Program Code

PYTHON code for parameter estimation using grid search with block K-fold cross-validation

```
from __future__ import print_function

from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import classification_report
from sklearn.svm import SVC

# Loading the elec-consumer dataset
digits = datasets.load_elec-consumer()

# To apply an classifier on this data, we need to flatten the image, to
# turn the data in a (samples, feature) matrix:
```

```
n_samples = len(elec-consumer.images)
X = elec-consumer.images.reshape((n_samples, -1))
y = elec-consumer.target

# Split the dataset in two equal parts
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.5, random_state=0)

# Set the parameters by cross-validation
tuned_parameters = [{’kernel’: [’rbf’], ’gamma’: [1e-3, 1e-4],
                     ’C’: [1, 10, 100, 1000]},
                    {’kernel’: [’linear’], ’C’: [1, 10, 100, 1000]}]

scores = [’precision’, ’recall’]

for score in scores:
    print("# Tuning hyper-parameters for  score)
    print()

    clf = GridSearchCV(SVC(), tuned_parameters, cv=5,
                       scoring=’s_macro’ score)
    clf.fit(X_train, y_train)

    print("Best parameters set found on development set:")
    print()
    print(clf.best_params_)
    print()
    print("Grid scores on development set:")
    print()
    means = clf.cv_results_[’mean_test_score’]
    stds = clf.cv_results_[’std_test_score’]
    for mean, std, params in zip(means, stds, clf.cv_results_[’params’]):
        print("0.3f (+/-0.03f) for r" (mean, std * 2, params))
    print()

    print("Detailed classification report:")
    print()
    print("The model is trained on the full development set.")
    print("The scores are computed on the full evaluation set.")
    print()
    y_true, y_pred = y_test, clf.predict(X_test)
    print(classification_report(y_true, y_pred))
    print()
```

PYTHON code for Gradient Boosting Machine Algorithm

```
import numpy as np
import matplotlib.pyplot as plt

from sklearn import ensemble
from sklearn import datasets

X, y = datasets.make_hastie_10_2(n_samples=12000, random_state=1)
X = X.astype(np.float32)

# map labels from {-1, 1} to {0, 1}
labels, y = np.unique(y, return_inverse=True)
```

```
X_train, X_test = X[:2000], X[2000:]
y_train, y_test = y[:2000], y[2000:]

original_params = {'n_estimators': 1000, 'max_leaf_nodes': 4,
 'max_depth': None, 'random_state': 2,'min_samples_split': 5}

plt.figure()

for label, color, setting in [('No shrinkage', 'orange',
     {'learning_rate': 1.0, 'subsample': 1.0}),
       ('learning_rate=0.1', 'turquoise',
       {'learning_rate': 0.1, 'subsample': 1.0}),
        ('subsample=0.5', 'blue',
        {'learning_rate': 1.0, 'subsample': 0.5}),
        ('learning_rate=0.1, subsample=0.5', 'gray',
        {'learning_rate': 0.1, 'subsample': 0.5}),
        ('learning_rate=0.1, max_features=2', 'magenta',
        {'learning_rate': 0.1, 'max_features': 2})]:
    params = dict(original_params)
    params.update(setting)

    clf = ensemble.GradientBoostingClassifier(**params)
    clf.fit(X_train, y_train)

    # compute test set deviance
    test_deviance = np.zeros((params['n_estimators'],), dtype=np.float64)

    for i, y_pred in enumerate(clf.staged_decision_function(X_test)):
        # clf.loss_ assumes that y_test[i] in {0, 1}
        test_deviance[i] = clf.loss_(y_test, y_pred)

    plt.plot((np.arange(test_deviance.shape[0]) + 1)[::5],
    test_deviance[::5], '-', color=color, label=label)

plt.legend(loc='upper left')
plt.xlabel('Boosting Iterations')
plt.ylabel('Test Set Deviance')
```

References

1. C. W. Gellings, The smart grids: enabling energy efficiency and demand response. The Fairmont Press Inc., 2009.
2. N. Arghira, L. Hawarah, S. Ploix, Prediction of appliances energy use in smart homes, Energy. 2012. V. 48. P. 128-134.
3. R.K. Jain, K.M. Smith, P.J. Culligan, Forecasting energy consumption of multi-family residential buildings using support vector regression:investigating the impact of temporal and spatial monitoring granularity on performance accuracy, Appl. Energy. 2012. V. 123. P. 168–178.
4. L. Breiman, Random forests, Machine learning. 2001. V. 45, 1, P. 5-32.
5. L. Breiman, Bagging predictors, Machine learning. 1996. 24(2), V.123–140.
6. C. Bergmeir, J.M. Benítez, On the use of cross-validation for time series predictor evaluation, Information Sciences. 2012. 191, P.192-213.
7. V. Ediger, S. Akar, ARIMA forecasting of primary energy demand by fuel in Turkey, Energy Policy. 2007. 35, P.1701–1708.

8. S. Cincotti, G. Gallo, L. Ponta, Modeling and forecasting of electricity spot-prices: Computational intelligence vs. classical econometrics, AI Commun. 2014. V. 27, pp. 301–314.
9. F.J. Ardakani, M.M. Ardehali, Novel effects of demand side management data on accuracy of electrical energy consumption modeling and long-term forecasting, Energy Convers. Manag. 2014. V. 78, pp. 745–752.
10. E. Yu. Shchetinin, Cluster-based energy consumption forecasting in smart grids, Springer Communications in Computer and Information Science (CCIS), 2018, 919, P. 446-456.
11. https://data.gov.ie/dataset/energy-consumption-gas-and-electricity-civic-offices-2009-2012/.
12. A. Liaw, M. Wiener, Classification and Regression by randomForest, R News. 2002. V. 2(3), P. 18-22.
13. J. Friedman, Greedy function approximation: a gradient boosting machine, Ann. Stat. 2001. V.29(5). P. 1189-1232.
14. K.P. Burnham, D.R. Anderson, Model Selection and Multi-Model Inference: A Practical, Information-theoretic Approach. Springer Verlag, 2002.
15. M.J. Kane, N. Price, M. Scotch M., Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks, BMC Bioinformatics, 2014, 15, P. 276- doi:10.1186/1471-2105-15-276.
16. J. Granderson, S. Touzani, C. Custodio, Accuracy of automated measurement and verification techniques for energy savings in commercial buildings, Applied Energy, 2016, 173, 296–308.
17. R.K. Jain, K.M. Smith, P.J. Culligan, J.E. Taylor, Forecasting energy consumption of multi-family residential buildings using support vector regression:investigating the impact of temporal and spatial monitoring granularity on performance accuracy, Appl. Energy, 2014, 123, P. 168–178.
18. Z.H. Zhou, Ensemble learning , Encycl. Biom., 2015, P. 411–416.
19. A.L. Zorita, M.A. Fernández-Temprano, L. A. García-Escudero, O. Duque-Perez, A statistical modeling approach to detect anomalies in energetic efficiency of buildings, Energy Build, 2016, 110, P.377–386.
20. M. Amozegar, K. Khorasani, An ensemble of dynamic neural network identifiers for fault detection and isolation of gas turbine engines, Neural Network, 2016, 76, P. 106–121.
21. E. Yu. Shchetinin, V.S. Melezhik, L.A. Sevastyanov, Improving the energy efficiency of the smart buildings with the boosting algorithms, Proceedings of the Selected Papers of the 12th International Workshop on Applied Problems in Theory of Probabilities and Mathematical Statistics in the framework of the Conference on Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems (APTP+MS'2018), CEUR Workshop Proceedings, 2018, 2332, P.69-78.
22. E.Yu. Shchetinin, E.A. Popova, Smart buildings energy savings with gradient boosting algorithm, Communications in Computer and Information Science, CEUR Workshop Proceedings, 2018, Proc. of selected papers of the 8th International Conference 'Distributed Computing and Grid-technologies in Science and Education", GRID-2018, 2267, P. 318-322.