LBNL-2001097

BERKELEY LAB

Lawrence Berkeley National Laboratory

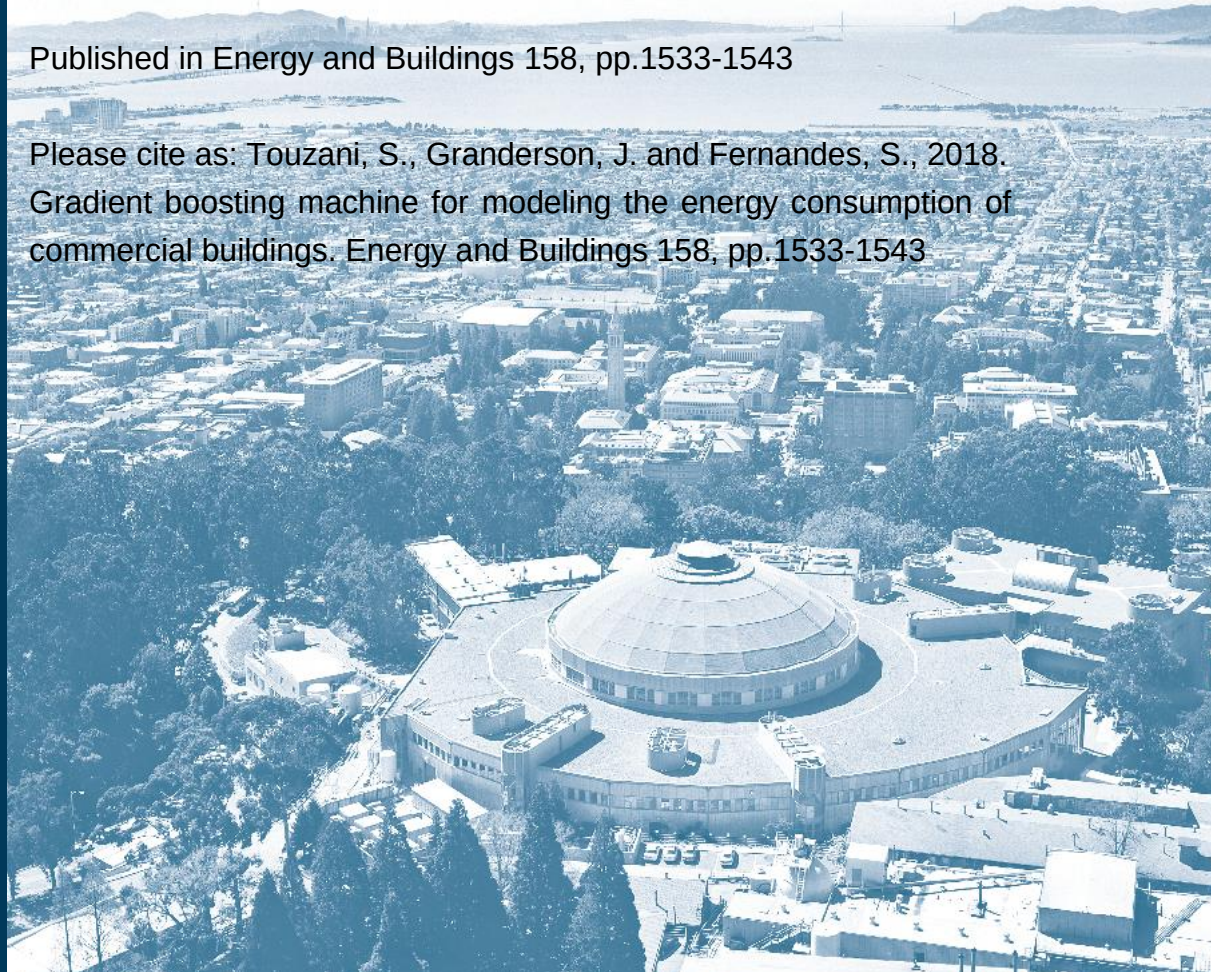
Gradient boosting machine for modeling the energy consumption of commercial buildings

Samir Touzani, Jessica Granderson, Samuel Fernandes

Energy Technologies Area
January, 2018

Published in Energy and Buildings 158, pp.1533-1543

Please cite as: Touzani, S., Granderson, J. and Fernandes, S., 2018. Gradient boosting machine for modeling the energy consumption of commercial buildings. Energy and Buildings 158, pp.1533-1543



Gradient boosting machine for modeling the energy consumption of commercial buildings

Samir Touzani[*1], Jessica Granderson[1], Samuel Fernandes[1]

[1]*Lawrence Berkeley National Laboratory, 1 Cyclotron Rd., Berkeley, CA 94720, USA*

ABSTRACT

Accurate savings estimations are important to promote energy efficiency projects and demonstrate their cost-effectiveness. The increasing presence of advanced metering infrastructure (AMI) in commercial buildings has resulted in a rising availability of high frequency interval data. These data can be used for a variety of energy efficiency applications such as demand response, fault detection and diagnosis, and heating, ventilation, and air conditioning (HVAC) optimization. This large amount of data has also opened the door to the use of advanced statistical learning models, which hold promise for providing accurate building baseline energy consumption predictions, and thus accurate saving estimations. The gradient boosting machine is a powerful machine learning algorithm that is gaining considerable traction in a wide range of data driven applications, such as ecology, computer vision, and biology. In the present work an energy consumption baseline modeling method based on a gradient boosting machine was proposed. To assess the performance of this method, a recently published testing procedure was used on a large dataset of 410 commercial buildings. The model training periods were varied and several prediction accuracy metrics were used to evaluate the model's performance. The results show that using the gradient boosting machine model improved the R-squared prediction accuracy and the CV(RMSE) in more than 80 percent of the cases, when compared to an industry best practice model that is based on piecewise linear regression, and to a random forest algorithm.

Keywords: Gradient boosting machine, machine learning, statistical regression, baseline energy modeling, energy efficiency, savings measurement and verification.

[*] Corresponding Author: stouzani@lbl.gov; 1 Cyclotron Rd., Berkeley CA, 94720; (510) 486-6772.

1. Introduction

According to the Commercial Buildings Energy Consumption Survey 2012 (EIA 2012), 6,963 trillion British thermal units (Btu) of total site energy (e.g., energy delivered to buildings) have been used by approximately 5.6 million U.S. commercial buildings, which roughly represent 19 percent of the U.S. total primary energy consumption (Kelso 2012). The energy sources of this consumption are distributed as: 4,241 trillion Btu of electricity, 2,248 trillion Btu of natural gas, 134 trillion Btu of fuel oil (diesel, kerosene, and distillate fuel oil), and 341 trillion Btu of district heat. Between 1979 and 2012, the total electricity consumption of commercial buildings has almost doubled, while the natural gas consumption slightly decreased. The burning of coal and natural gas to supply these buildings with electricity, in addition to the direct burning of natural gas, gives commercial buildings one of the largest shares of the U.S. carbon dioxide emission. To reduce the environmental and cost impacts associated with the commercial buildings sector, several energy efficiency programs have been implemented. For example, at the state and federal level, long term energy savings targets have been established, and these targets must be achieved by utility and non-utility program administrators through energy efficiency programs. These various programs are generally implemented through energy service companies (ESCO) whose annual revenues were evaluated at around $7 billion, with roughly 75 percent of those revenues associated with energy efficiency projects (Satchwell et al. 2010).

In the energy efficiency industry, *measurement and verification* (M&V) is the process of estimating savings, and is therefore critical to establishing the value of energy efficiency to building owners, utility rate payers, and service providers. However, M&V can be quite costly and time consuming, and questions as to the accuracy of the estimated savings remain. Depending on the M&V methods employed and whether third-party evaluation is included, M&V costs can range from 1 to 5 percent of project portfolio costs (Jayaweera and Haeri 2013). Today, the growing availability of data from smart meters and devices, combined with advanced data analytics, offers the potential to streamline the M&V process through increased levels of automation, while maintaining or improving the accuracy of the result. These smart meter- and data analytics-based approaches are increasingly referred to as *M&V 2.0*. The M&V 2.0 methods are currently receiving a surging level of attention in the industry, particularly in the context of utility energy efficiency programs, due to their promise to reduce program time and costs, and unlock untapped savings through whole-building focused programs (Goldberg et al. 2015; Rogers et al. 2015).

The baseline models used in M&V 2.0 are empirical models that relate energy usage to parameters such as outdoor air temperature, humidity, or building operating schedule. These models are developed using consumption data before the efficiency measure was implemented. They are projected into the post-measure period to estimate what the energy use would have been if the measure had not been implemented. The difference between the estimated and the metered energy consumption is taken as the *avoided energy use* or energy savings. Traditionally, monthly utility bill data were used to build the baseline models; however, the increasing availability of hourly and 15-minute interval meter data have enabled new models with the potential for more accurate M&V.

In recent years, several baseline energy modeling approaches that use interval meter data have been introduced in the literature. These methods are based on traditional linear regression, nonlinear regression, and machine learning methods. In the framework of linear regression, the model described in Price et al. (2011) includes time of the day, day of the week, and two temperature variables to allow different heating and cooling slopes. It also can include humidity

and holidays as variables. This model is fit with ordinary least squares regression, which is the common state of practice in M&V applications. The Time-of-Week-and-Temperature model (Mathieu et al. 2011) is a regression model that includes time of week and a piecewise-continuous linear outdoor temperature response with several change points. In the framework of nonlinear regression, Srivastav et al. (2013) introduced a Gaussian mixture model for modeling the energy of a retail store building. In Heo and Zavala (2012) and Burkhart et al. (2014) a Gaussian process modeling approach was used to determine building energy savings. A model based on a kernel smoother was proposed in Brown et al. (2012). This model included 4 parameterized measurements of time, temperature, humidity, and wind velocity, with a total of 14 input parameters. Finally, in the framework of the machine learning methods, the most widely used method in the building energy modeling is the artificial neural network, which has shown to be efficient at capturing complex energy consumption trends. Another well-known machine learning method that has been successfully applied to predict building energy consumption is the support vector machine. Support vector machines have the advantage of being able to effectively approximate nonlinear effects with even a small number of training points. However, both support vector machines and neural network algorithms are harder to tune than the more recent gradient boosting machine learning algorithm that is explored in this paper. Specifically, selecting the right kernel for the support vector machines, or the right topology of the artificial neural network can be very challenging tasks. An extensive review of the application of neural networks and support vector machines to building energy consumption prediction can be found in Zhao and Magoulès (2012).

The past two decades have seen substantial advances in the development of new machine learning methods, and the most promising approaches in terms of prediction accuracy are part of the ensemble learning family of algorithms. Ensemble methods build a model by training several relatively simple base models (also known as *weak learners*) and then combine them to create a more predictive model. The most well known ensemble learning algorithms use bootstrap aggregation, also known as *bagging* (Breiman 1996); *random forests* (Breiman 2001); *extremely randomized trees,* also called *extra trees* (Geurts et al. 2006); and *boosting* (Schapire 1990). The bagging, extra trees, and random forests are based on a simple averaging of the base learner, while the boosting algorithms are built upon a constructive iterative strategy. While historically these ensemble machine learning algorithms are applied with good success in multiple fields, they are just beginning to be applied to problems in building energy modeling. For instance, Raftery and Hoyt implemented extra tree and random forest algorithms in their automated measurement and verification tool (Raftery and Hoyt 2016). In Ahmad et al. (2017) the authors used random forest for predicting the hourly HVAC energy consumption. Araya et al. (2017) applied the random forest algorithm for anomaly detection in building energy consumption.

This work adds to the research and practitioner communities' growing interest in new methods for efficiency applications that leverage advances in machine learning. It evaluates a baseline model based on gradient boosting machine (GBM) method, which is part of the boosting family algorithm, including it's development and testing for accuracy in predicting commercial building electricity consumption. Recently this approach has gained increasing interest from various scientific and engineering fields, such as ecology (Oppel et al. 2012), physics (Aad et al. 2014), and atmospheric science (Sayegh et al. 2016). GBM models have been shown to have strong predictive performance and high flexibility.

2. Background on Gradient Boosting Machine

2.1. Decision trees: base learner

Efficient and conceptually simple, *decision trees*, also known as *regression trees*, are regression methods that consist of partitioning the input parameters space into distinct and non-overlapping regions following a set of if-then rules. The splitting rules identify regions that have the most homogeneous response to the predictor, and within each region a simple model, such as a constant, is fitted. In the regression framework the constant usually corresponds to the average output of the training dataset in the corresponding region. For a commercial building energy baseline application, a simple example is depicted in Figure 1, where the two input parameters T and t represent the outdoor air temperature (OAT) and the time of the week, respectively. The output E corresponds to the energy consumption of the building. T_1 and T_2 are the split points of the OAT, and t_1 and t_2 are the splits points of time of the week. $E_{1,\dots,5}$ are the terminal nodes, also called *leaves of the tree* (the outputs). The split points are chosen to minimize a loss-function, which in the case of regression trees is usually the mean squared error. The splitting continues until a stopping criterion is reached, e.g., the number of training points within a region reaches some defined threshold. These different splitting steps correspond to the depth of the tree (i.e., the complexity of the tree). To make a prediction for new data points, the data are split following the trained split points, and the same constants in the terminal nodes are used to make the predictions.

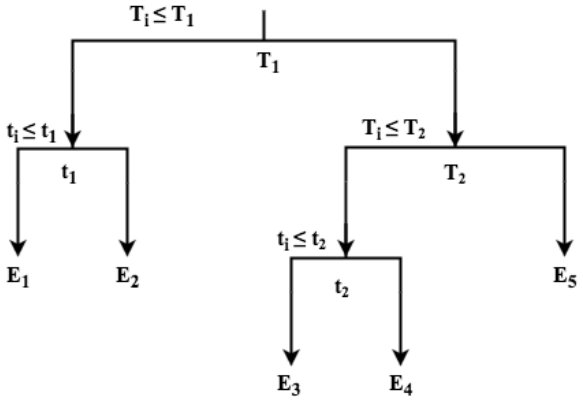


Fig. 1. Decision tree with two input parameters: the outdoor air temperature T and time of the week t.

The use of decision trees as a regression technique has several advantages, one of which is that the splitting rules represent an intuitive and very interpretable way to visualize the results. In addition, by their design, they can handle simultaneously numerical and categorical input parameters. They are robust to outliers and can efficiently deal with missing data in the input parameters space. The decision tree's hierarchical structure automatically models the interaction between the input parameters and naturally performs variable selection, e.g., if an input parameter is never used during the splitting procedure, then the prediction does not depend on this input parameter. Finally, decision trees algorithms are simple to implement and computationally efficient with a large amount of data.

In spite of these advantages, decision trees are usually less accurate than other regression methods, which is due to several limitations. Since they can feature high depth (i.e., high complexity), they may not effectively generalize the relation between the input parameters and

the output, and therefore can be prone to over-fitting, which can lead to poor prediction performance. Some techniques can be used to avoid this problem (James et al. 2013). The splitting rules are strongly dependent on the training data, and a small change in the training data might generate a different tree. They are not optimal to approximate smooth functions such as a straight line, because a single decision tree extrapolates the input/output relation with a constant value. Nevertheless, several methods have been introduced to bypass these limitations and to increase the predictive performance of the decision tree methods, among which, the most popular and most efficient are bagging (Breiman 1996), random forest (Breiman 2001), and gradient boosting machine (Friedman 2001). While these methods are similar in that the prediction is based on an ensemble of decision trees models, the way that the ensembles are created differs significantly. For random forest and bagging the decision trees are created independently - they have maximum depth and each one of them has the *same* contribution to the final result. The structure of these two algorithms makes them reduce the variance of a large number of complex decision trees (high depth), however they cannot achieve the bias reduction (for more details see Kuhn et al. 2013). In contrast, the decision trees in gradient boosting machine have small depth and are built, (as explained in the following section of the paper), by sequentially modeling the residuals. In this way, the decision trees in gradient boosting machine are dependent and have *different* contributions to the final prediction. Thus the gradient boosting machine is capable of reducing the model variance by averaging several decision trees and it is also capable of reducing the bias through the sequential error modeling.

2.2. Gradient boosting machine

Boosting algorithms were originally introduced by the machine learning community (Schapire 1990; Freund 1995; Freund and Schapire 1996) for classification problems. The principle approach is to combine iteratively several simple models, called "weak learners," to obtain a "strong learner" with improved prediction accuracy. Friedman et al. (2000) introduced a statistical point of view of boosting, connecting the boosting algorithm to the concepts of loss functions. Friedman extended the boosting to the regression by introducing the gradient boosting machines method (GBM) (Friedman 2001). The GBM method can be seen as a numerical optimization algorithm that aims at finding an additive model that minimizes the loss function. Thus, the GBM algorithm iteratively adds at each step a new decision tree (i.e., "weak learner") that best reduces the loss function. More precisely, in regression, the algorithm starts by initializing the model by a first guess, which is usually a decision tree that maximally reduces the loss function (which is for regression the mean squared error), then at each step a new decision tree is fitted to the current residual and added to the previous model to update the residual. The algorithm continues to iterate until a maximum number of iterations, provided by the user, is reached. This process is so-called *stage wise*, meaning that at each new step the decision trees added to the model at prior steps are not modified. By fitting decision trees to the residuals the model is improved in the regions where it does not perform well.

The GBM algorithm has better results if at each iterative step the contribution of the added decision tree is shrunk using a shrinkage parameter α, called the *learning rate.* The idea behind the shrinkage procedure in the context of GBM is that a higher number of small steps provide a higher accuracy than a lower number of large steps. The learning parameter α can take a value between 0 and 1 and the smaller it is, the more accurate the model will be. However, choosing a stronger shrinkage (smaller α) implies a higher number of iterations to achieve convergence, since the value of α is inversely proportional to the number of iterations.

Another way to increase the predictive accuracy of the GBM algorithm is to add randomization into the fitting process (Friedman 2002). At each iterative step, rather than using the full training dataset, a randomly selected (usually without replacement) subsample is used to fit the decision tree. When the number of observations is large enough, the default fraction of the data used at each iteration is usually equal to 0.5, which means that 50 percent of the dataset is used at each iteration. However, one should check several values of a subsample fraction to evaluate the impact of decreasing the number of data points on the fitting quality of the model. In addition to improving the accuracy of the GBM model, the subsampling has the useful effect of reducing the computational cost of the algorithm by a factor equivalent to the factor of the subsampling.

A simplified illustration of this algorithm is provided by the following pseudo-code:

1. The user selects the depth of the decision trees *d*, the number of iterations *K*, the learning rate α, and the subsample fraction η.
2. Initialization: set the residual $r_0 = y$ and $\hat{f} = 0$. The mean value of y has also been suggested as an initial guess of $\hat{f}$ (Kuhn et al. 2013).
3. For $k = 1,2,\ldots,K$, do the following:
    a. Randomly choose a subsample $\{y_i,x_i\}^{N'}$ from the full training dataset, with N' is the number of data points corresponding to the fraction η
    b. Using $\{y_i,x_i\}^{N'}$ fit a decision tree $\hat{f}^k$ of depth *d* to the residual r_{k-1}
    c. Update $\hat{f}$ by adding the decision tree to the model
    $$\hat{f}(x) \leftarrow \hat{f}(x) + \alpha \hat{f}^k(x)$$
    d. Update the residual
    $$r_k \leftarrow r_{k-1} - \alpha \hat{f}^k(\mathrm{x})$$
4. endFor

In the GBM there are four hyper-parameters that need to be tuned: (1) *d* the depth of decision trees, which also controls the maximum interaction order of the model; (2) *K* the number of iterations, which also corresponds to the numbers of decision trees; (3) α the learning rate, which is usually a small positive value between 0 and 1, where decreases lead to slower fitting, thus requiring the user to increase *K;* (4) η the fraction of data that is used at each iterative step. The next section presents the goal of tuning these hyper-parameters and the method followed.

3. GBM Hyper-Parameter Tuning

As for any machine learning predictive method, over-fitting is a concern for the GBM algorithm. *Over-fitting* is the tendency of the model to fit the training data too well, at the expense of the predictive accuracy, which means that the estimated pattern is not likely to be generalized to new data points. This happens because an over-fitting model will fit the characteristics of the noise that is present in the training data rather than identifying the general pattern of the input/output relation. The over-fitting is usually the downside of an unnecessarily over-complex model. In the case of a GBM model this may happen if a practitioner selects a very high number of iterations *K* and a too large depth of the decision trees *d*.

The question remains as to how to choose the right combination of hyper-parameters to avoid over-fitting and at the same time provide the best predictive accuracy. In the statistical and machine learning literature several approaches have been introduced and studied. However, the most popular and conceptually easy to understand is the *search grid* method. This approach consists of defining a grid of combinations of hyper-parameter values, building a model for each combination, and selecting the optimal combination using metrics that quantify the model performance in term of predictive accuracy. It is clear that it is not advised to use the same observations that have been used as training data to estimate the models to compare the predictive performance. Therefore, one should assess the accuracy on an independent set of data points. Ideally, the available data should be split on two samples: the training sample and the validation (testing) sample. However, in practice, it is not often possible to hold out enough data points, to estimate accurately the predictive performance of the models without affecting the estimation quality. When the number of observations is not very large, reducing the number of training points might produce a poor estimation of the input/output relation. Cross validation (CV), and especially the k-fold-CV, is the most traditional method to overcome a scarcity of data.

The k-fold cross validation method consists of randomly splitting the training dataset into k subsamples, called *folds*, of roughly equal size. The first model is estimated using k-1 folds as a training dataset and the held-out fold (test set) is used to estimate the prediction accuracy metric. In this study the root mean squared error (RMSE) was used: $RMSE = \sqrt{\frac{1}{n}\sum_i^n (y_i - \hat{y}_i)^2}$, where y_i is the actual metered value, $\hat{y}_i$ is the predicted value. This procedure is reiterated k times, and at each time a different fold is used as a test set. The k-fold CV estimate of the RMSE is given by equation (1):

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

When used with the search grid method, the k-fold CV estimates of the RMSE are computed for each combination of the hyper-parameters. The optimal combination is the one that minimizes the RMSE^{CV}. Although, there is no formal rule of choosing the value of k, in practice k = 5 or k = 10 is used. Note that the higher the value of k is, the more the approach is computationally demanding.

In the context of energy baseline modeling using interval meter data, one should consider that the energy consumption time series usually have an intrinsic serial correlation, i.e., the consecutive observations are dependent. Traditionally, in the time series forecasting literature, the out-of-sample evaluation method is used instead of the k-fold CV, where a block of data at the end of the time series is held out for testing. However, as previously discussed, this can be problematic for relatively small datasets, and the error can be completely misestimated if the statistical properties of the time series of the test set are different from the training data. For example, in the case of a school, if the test set takes place during the vacation period the estimation of the performance metric will be biased. To account for these issues, a modified version of the k-fold CV is proposed in this paper. This method, called k-fold-blocks CV, consists in randomly selecting blocks of data rather than randomly selecting unique observations, when the k splitting is performed. More precisely, in this work, two types of blocks are considered: the first corresponds to a day of the week, and the second corresponds to a week.

4. Application and Results

In this section the GBM algorithm is applied to construct baseline energy use models. The temporal variation in electricity consumption in commercial buildings is subject to several sources, including weather, holidays, and daily and weekly periodicity. For example, the electricity consumption in office buildings is expected to be lower during weekends and nights. To capture these effects three different input variables are considered: outside air temperature, time of the week, and U.S. federal holidays (defined as a dummy variable: 1 when it is a holiday and 0 elsewhere). To facilitate the application of the described GBM baseline model, we have provided open-source R package, which is available at https://github.com/samirtouzani/GBMbaseline. The implementation of the proposed baseline modeling method is based on the XGBoost R package (Chen and Guestrin 2016). This relatively new implementation of the GBM algorithm has achieved state-of-the-art results in several machine learning competitions.

The model predictive accuracy is evaluated using the testing procedure described in (Granderson et al. 2015; Granderson et al. 2016). For each building in the dataset the metered whole-building electricity data is divided into hypothetical training periods and prediction periods, and meter data from the prediction period is "hidden" from the model. The trained model is used to forecast the load throughout the prediction period, and predictions are then compared to the actual meter data that had been hidden. The GBM results are compared to a random forest based baseline model (RF) and to the Time-of-Week-and-Temperature model (TOWT).

The RF model was developed using the randomForest R package (Liaw and Wiener 2002). Similar to the GBM models, outside air temperature, time of the week, and U.S. federal holidays dummy variable were considered as the input variables. The two RF hyper-parameters that were considered in the tuning process were: the number of input variables randomly sampled as candidates at each split (mtry) and the number of trees to grow (ntree). To tune these two hyper-parameters the search grid method and k-fold-blocks CV procedure were used with *day* defined as the block and with k=5. Thus mtry is selected in the set {1,2,3} and ntree is selected in the set {50,250,500}. Note that the 5-fold-blocks CV was used rather than the standard 5-fold CV method; this choice was motivated by the fact that empirical results showed that using 5-fold-blocks CV improved the accuracy of the RF models. For a more detailed description of the RF algorithm refer to Breiman (2001) and Ahmad et al. (2017).

The TOWT model is a piecewise linear model where the predicted energy consumption is a combination of two terms that relate the energy consumption to the time of the week and the piecewise-continuous effect of the temperature. Each time of the week has a different predicted energy consumption, and the temperature effect is estimated separately for periods of the day with high and low energy consumption in order to capture the pattern for occupied and unoccupied building periods. Further description of the TOWT model can be found in Mathieu et al. (2011). This model was chosen as a useful benchmark because in previous studies (Granderson et al. 2016, Granderson et al. 2017) it was shown to be highly accurate, equaling or outperforming industry standard models.

4.1. Office building case study

A study on a single building was conducted to examine the parameters tuning process of the GBM model. A 5-fold-block CV was considered, where a day is defined as the block. For simplicity, the following notation was used to characterize this configuration of the GBM model:

"GBM_day." The results were compared to TOWT model. These two algorithms were used for baseline modeling of electricity consumption of a large office building located in Seattle, Washington. For this study, 24 months of data, at 15-minute granularity, were available. The outside air temperature data were acquired, using the zip code of the building and the closest weather station from the Weather Underground service (wunderground 2015). Both models were trained using the first 12 months of the dataset, and the prediction was performed using the last 12 months of data. Thus the 24 months were divided into 12 months of training period and 12 months of prediction period.

To study the behavior of the GBM_day hyper-parameters, the model was tuned using a relatively fine search grid: the depth of the decision trees d is selected in the set {3,4,5,6,7,8,9,10}, the learning rate α is chosen between the set {0.01,0.05,0.1,0.5}, the number of iterations K is selected within a set spanning from 50 iterations to 4000 with a granularity of 50 iterations (i.e., {50,100,…,3950,4000}), and the subsample fraction η is fixed at {0.5,0.75,1} that correspond to 50 percent, 75 percent, and 100 percent of the training data.

The impacts of the GBM hyper-parameters are shown in Figure 2 and Figure 3, where in each plot the vertical axis represents the logarithm of $RMSE^{CV}$ (marked as log(RMSE)), where $RMSE^{CV}$ is the root mean square error (as defined in Section 3) calculated by the 5-fold-block CV with day defined as the block. In each plot of Figure 2 the convergence of the accuracy metric versus the iterations number are shown for the four learning rates: the red circles correspond to the learning rate of 0.01, the green triangles correspond to 0.05, the blues squares to 0.1 and purple plus signs to 0.5. In addition, each plot of this figure corresponds to the results of a specific depth of the decision trees. These depths are displayed in grey across the top plots. For more clarity of the results presentation and because of the small variation in the results between consecutive depths, only four of the studied depths are displayed.

The results indicate that overall, a value of 0.5 for the learning rate was too high, since with this value the algorithm was too sensitive for both the number of iterations and the depth of the decision trees. It also seems that at a learning rate of 0.01 and a decision tree depth of 5 (optimal depth) the algorithm did not achieve the optimal number of iterations at 4000 iterations. This likely explains why the optimal learning rate for this example was 0.05, which has a faster convergence rate. Furthermore, the decrease in predictive accuracy are attributable to the fact that at one point by increasing the number of iterations and the model complexity the algorithm starts to over-fit the training data, except for the learning rate of 0.01 at depth of decision tree of 5.

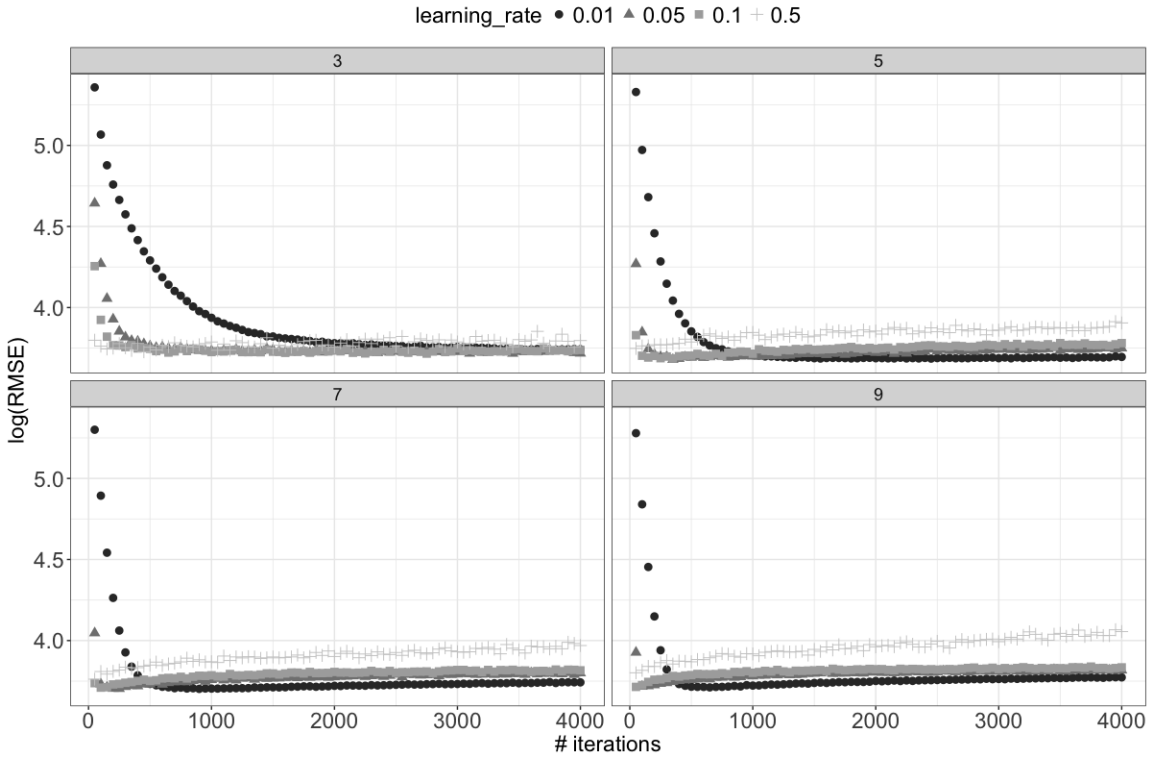


Fig. 2. The relationship between the learning rate, number of iterations, and the depth of the decision trees (the numbers in the top gray boxes)

The impact of the subsampling rate is summarized in Figure 3, where each plot shows the convergence of the accuracy metric (log(RMSE)) versus the iterations number at the optimal depth of decision trees (e.g., 5). The circles correspond to the subsample rate of 0.5, the triangles to a rate of 0.75, and the squares to the situation where no subsampling was applied. Each plot displays the results for each learning rate that is depicted in the grey heading of each plot. The impact of the subsampling ratio on the algorithm accuracy increases slightly when the learning rate increases, which means that with a higher learning rate the algorithm needs more observations to achieve the best accuracy.

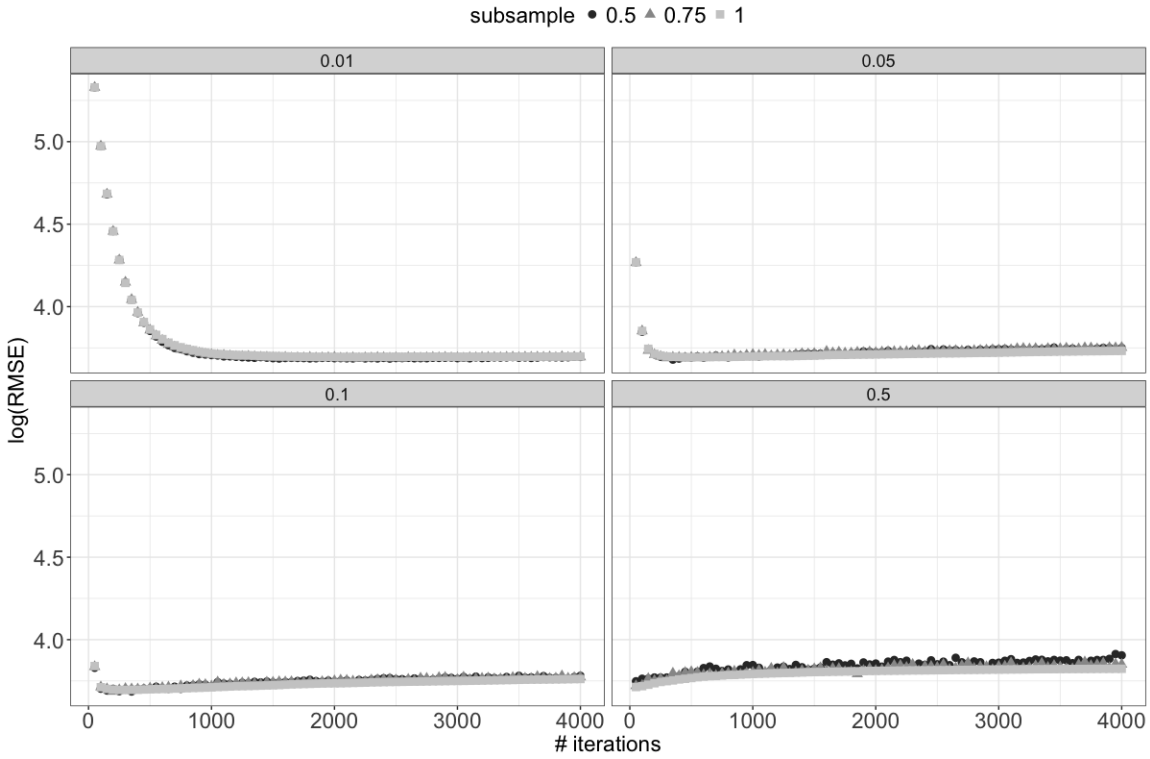


Fig. 3. The relationship between the learning rate (the numbers in the top gray boxes), number of iterations, and the fraction of the subsampling

The optimal hyper-parameters which were selected by the 5-fold-block CV (with *day* defined as the block) are: 300 for the number of iterations; 5 for the decision tree depth; 0.5 for the subsampling ratio, and 0.05 for the learning rate. In addition to the GBM_day model, the TOWT and RF models were also trained. Figure 4 depicts the density of the relative prediction errors of the results from GBM_day versus RF and GBM_day versus TOWT. The relative prediction error is defined by equation (2):

$$err_i = 100 \times (y_i - \hat{y}_i)/y_i. \qquad (2)$$

with y_i is the actual metered value, $\hat{y}_i$ is the predicted value of the prediction training period data. Table 1 summarizes the accuracy evaluation metrics (defined in the appendix) computed on the prediction period. These results show that for this case study GBM_day has higher predictive accuracy than TOWT and RF models.

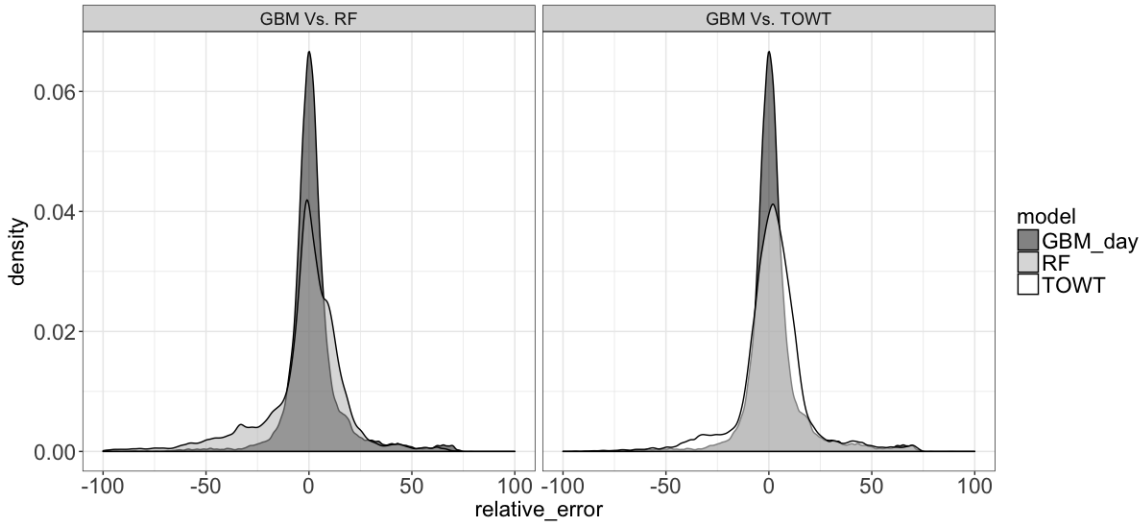


Fig 4. Density plots of the relative errors (in %) of the prediction of GBM_day Vs. RF and GBM_day Vs. TOWT

Model	R^2	CV(RMSE)	NMBE
GBM_day	**87.71**	**18.16**	**3.69**
RF	85.17	19.95	3.17
TOWT	83.06	21.32	4.17

Table 1. The accuracy metrics of GBM_day, RF and TOWT models.

As previously described, the GBM algorithm incorporates two type of randomization. The first randomization is related to the k-fold CV procedure and the second is related to the subsampling of the training data that occurs at each iteration. In order to assess the impact of these stochastic effects, one hundred models were trained using the same data from the office building, and at each time the seed number that initializes the pseudorandom number generator was changed (i.e., one hundred different seeds were used). The accuracy metrics R^2, CV(RMSE), and NMBE were computed for the prediction period across the one hundred models and summarized in Table 2. The results show that the variation in the accuracy results is relatively small, and as such, the seed number does not have a significant impact on the prediction accuracy.

Metric	25th	50th	75th
R^2	88.10	88.18	88.32
CV(RMSE)	17.70	17.81	17.87
NMBE	3.50	3.53	3.57

Table 2. Percentiles of the R^2, CV(RMSE) and NMBE when the seed number was varied.

4.2. Results of application on an extensive group of commercial buildings

4.2.1. Data description

To evaluate model performance on a larger number of buildings we used a dataset that comprises whole-building electricity consumption metered data gathered from 410 commercial buildings. The buildings are located in Northern and Central California (n = 201), Washington, D.C. (n = 179), and Seattle (n = 30), which represent ASHRAE climate zones 3 and 4. These buildings are "untreated" in terms of efficiency interventions. That is, they are not known to have

implemented major efficiency measures. The data have been measured at 15-minute intervals for each building. As in the case of the single building investigation, the outdoor air temperature data were acquired using the ZIP code of each building and the closest weather station from the Weather Underground service (wunderground 2015). This dataset represents a subset of the dataset that used in a prior study of model predictive accuracy (Granderson et al. 2016).

For each building, the time series data have been split into training period and prediction period. The prediction period was defined as the last 12 months of the available data. A 12-month post period (prediction period) is generally the standard for whole-building M&V of energy savings. The models are trained using two different training durations, which are 6 months and 12 months. These training periods correspond to the periods that immediately precede the prediction period. The 6 month duration is analyzed because being able to have a shorter baseline period is advantageous for sites that may not have a long record of pre-existing consumption. Thus, each prediction and training period of 12 months contains approximately 35,000 observations, and each training period of six months has roughly 17,500 observations. All buildings from the dataset had 24 months of electricity consumption and outside air temperature data. However, some missing data may be present in the time series but their number is lower than 1 percent of the total number of the observations.

As for the previous case study, the GBM hyper-parameters were tuned automatically using the search grid method with cross validation methods. However, given that the dataset used in this test was considerably large, the considered search grid was restricted compared to the first experiment. Thus, the depth of the decision trees d was selected in the set $\{3,\ldots,10\}$, the learning rate α was chosen between the values 0.1 and 0.05, the number of iterations K was selected within a set spanning from 15 iterations to 300 with a granularity of 15 iterations (e.g., $\{15,30,\ldots,285,300\}$), and the subsample fraction η was fixed at 0.5. To show the importance of using a k-block cross validation rather than a standard k-fold cross validation, three different versions of the GBM model were studied: A version using a standard 5-fold CV (named GBM), a version using 5-fold-block CV where a day is used as a block (GBM_day), and a version using 5-fold-block CV where a week is used as a block (GBM_week).

The experiments were run on an Intel i7 3.5 Ghz workstation with 4 cores (8 threads). The implemented GBM and RF baseline algorithms used 5 threads to perform in parallel the 5-fold CV (as well as the 5-fold-block CV). On average, for a 12 month training period, the execution time for RF model was approximately 17 minutes, and for the GBM models it was approximately 7 minutes. For the TOWT model, the average execution time was approximately 2 minutes.

4.2.2. Results

The accuracy metrics R^2, CV(RMSE), and NMBE were computed across the full dataset of buildings and summarized in Table 3, Table 4, and Table 5. An insight into accuracy degradation as the training period is shortened from 12 months to 6 months is also shown.

Table 3 shows that in terms of R^2 the GBM models outperform the benchmarks RF and TOWT models. This is especially true for the GBM_day (with day defined as the block) and GBM_week (with week defined as the block) configurations, and as expected the GBM model that used the standard k-fold CV method had a lower accuracy than the other two versions that used the k-fold block CV. When the training period was reduced, a significant degradation was noticed for the GBM model (with standard k-fold CV) and a small decrease in R^2 occurred for GBM_day, GBM_week and RF models, while accuracy improved in the TOWT model, which means that for

this dataset the TOWT model does not benefit from an increase in the number of observations. For 50 percent of the buildings, the GBM_day and GBM_week algorithms produce accurate baseline models that explain ~70 percent and higher of the electricity consumption variability. This number drops to ~60–65 percent for the GBM, TOWT and RF models.

Model	12 months			6 months		
	25th	50th	75th	25th	50th	75th
GBM	34.44	65.15	85.45	21.36	59.44	82.55
GBM_day	45.24	70.52	85.82	45.22	69.28	83.96
GBM_week	45.10	70.34	85.90	45.25	69.27	83.95
RF	27.19	62.51	82.60	35.19	61.11	80.95
TOWT	6.18	60.24	82.74	11.99	61.24	83.24

Table 3. Percentiles of the R^2 for each model, for a 12- and 6-months training period and for a 12-month prediction period

The results for the CV(RMSE) metric are summarized in Table 4. The GBM_day and GBM_week algorithms marginally outperformed the TOWT, the RF and the GBM in term of CV(RMSE). The GBM models' accuracy improved when their training period was increased from 6 months to 12 months, while the accuracy of the TOWT model was slightly lowered. ASHRAE Guideline 14 (2014) specifies that the CV(RMSE) calculated on the training period should be less than 25 percent if 12 months of post-measure data are used. While not directly comparable, because in this study the CV(RMSE) is computed on the prediction period (which is usually higher than the one computed on the training period), approximately 70 percent of buildings are likely to meet the ASHRAE requirement if GBM_day or GBM_week baseline models are used. If the TOWT and RF models are used, the requirement would be met in fewer cases – approximately 63 percent and 61 percent of buildings, respectively.

Model	12 months			6 months		
	25th	50th	75th	25th	50th	75th
GBM	12.67	18.35	30.14	13.90	20.74	32.19
GBM_day	12.44	17.42	27.17	13.17	18.73	27.61
GBM_week	12.47	17.32	27.20	13.11	18.74	27.64
RF	13.61	19.53	33.57	13.78	19.45	30.33
TOWT	13.57	19.34	29.55	13.30	18.48	29.09

Table 4. Percentiles of the CV(RMSE) for each model, for a 12- and 6-month training period and for a 12-month prediction period

Table 5 shows that there are no significant differences in the NMBE distribution among the three GBM models, where the NMBE ranges from ~-4 percent to ~4 percent for ~50 percent of the buildings. The TOWT and the RF models have a higher tendency of over-predicting (negative NMBE) the electricity usage than the GBM models do. However, these NMBE values should be carefully interpreted, because this variability in the NMBE may be the results of *actual* decreases (or increases) in building electricity, as opposed to a characteristic of the models.

Model	12 months			6 months		
	25th	50th	75th	25th	50th	75th
GBM	-4.22	-0.27	4.82	-4.30	-0.66	3.20
GBM_day	-4.20	-0.26	4.62	-4.41	-0.64	3.16
GBM_week	-4.27	-0.29	4.68	-4.47	-0.74	3.31
RF	-4.86	-1.07	4.74	-4.54	-0.99	2.81
TOWT	-5.74	-1.18	3.69	-5.77	-1.70	2.11

Table 5. Percentiles of the NMBE for each model, for a 12- and 6-month training period and for a 12-month prediction period

Table 6 and Table 7 summarize the percentage of buildings in the studied dataset for which the GBM models have are more accurate than the TOWT and RF models. Recall that for the R^2 metric higher values are desired, while for CV(RMSE) and NMBE values closer to zero are desired. For the CV(RMSE) columns the percentage represents the ratio of buildings that have lower CV(RMSE) in comparison with the TOWT model. Finally, for the NMBE columns the percentage corresponds to the ratio of buildings for which the absolute value of NMBE is lower than the absolute value of the TOWT NMBE. These results confirm that the GBM_day and GBM_week algorithms have the best accuracy in comparison to the TOWT and the RF models.

Model	12 months			6 months		
	R^2	CV(RMSE)	NMBE	R^2	CV(RMSE)	NMBE
GBM	77	64	57	42	27	52
GBM_day	88	81	59	65	55	51
GBM_week	88	81	57	67	57	50

Table 6. Percentage of buildings for which the GBM models have higher predictive accuracy than the TOWT model

Model	12 months			6 months		
	R^2	CV(RMSE)	NMBE	R^2	CV(RMSE)	NMBE
GBM	73	73	56	46	46	53
GBM_day	83	83	52	83	83	55
GBM_week	82	82	51	83	83	54

Table 7. Percentage of buildings for which the GBM models have higher predictive accuracy than RF model

The last step of this analysis was to evaluate the tuning process across each tuned hyper-parameter and each training period. In other words, the distributions of the optimal hyper-parameters were studies. In what follows, and for reasons of simplification, the analysis was restricted to the GBM_day model. The plots in Figure 5 show the distributions of the tuned hyper-parameters of GBM_day models across the full population of the commercial buildings, for both training periods. The plots of the first line of Figure 5 depict the distributions of the numbers of iterations needed to obtain the optimum solutions, and it appears that these distributions differ for 12 months and for 6 months. Indeed, it seems that GBM_day models need more iteration to converge when 6 months of training data was used. In addition, it appears that for the training period of 6 months, the number of times that the 0.1 learning rate was selected as the optimal learning rate was almost similar to the number of times 0.05 was selected, while when the training period was 12 months, the number of times that 0.05 appeared to be the optimal learning rate was three times higher than 0.1. Thus, from the plots of the first and the second line of Figure 5, one can note that when more observations were available for the training period, less iteration were needed for the GBM_day algorithm to converge to the optimal solution. Furthermore, a high

number of larger learning rate (α = 0.1) selected with a 6 months training period can be explained by the fact that with a smaller training period the number of iteration needed to converge when α = 0.05 is used is higher than 300, which was the maximum considered iterations in this test. To confirm this statement an additional analysis was performed the case with training period of 6 months and baseline model GBM_day. In this analysis the number of iterations *K* was selected within a set spanning from 15 iterations to 600 (with same granularity of 15 iterations). For convenience, in this analysis, this model is called GBM_day (600). The obtained results show that with a higher number of iteration the proportion of the optimal learning rates α for 6 months training period is almost the same as for 12 months training period, which means that for a higher proportion of buildings the optimal α was 0.05 (see first plot of the second line of Fig.6). In addition, a significant number of buildings have an optimal number of iterations *K* higher than 300 (see first line of Fig.6). However, the increase in the number of iterations *K* didn't produce significant change in the overall accuracy (see Table 8).

Finally, the plots from the third line of Figure 5 summarize the distribution of the optimal decision tree depth. It can be seen that the GBM_day model needs more observation (bigger training period) to select a more complex model (i.e., a higher depth for the decision tree) as an optimal model. Also, it appears that with the maximum number of iterations fixed at 300 on the search grid, it is not necessary to explore decision trees with depth higher than 8 or 7. In addition, from Figure 6 one can note that the increase of number of iterations *K* did not change the the distribution of the optimal decision tree depth for the 6 months training period. Note that if for example a higher number of input variables, such as occupation and/or solar radiation, are available, and if these parameters are correlated to the electricity consumption, it is highly likely that the distribution of the depth of the decision tree will be completely different, since more complex models can be optimal.

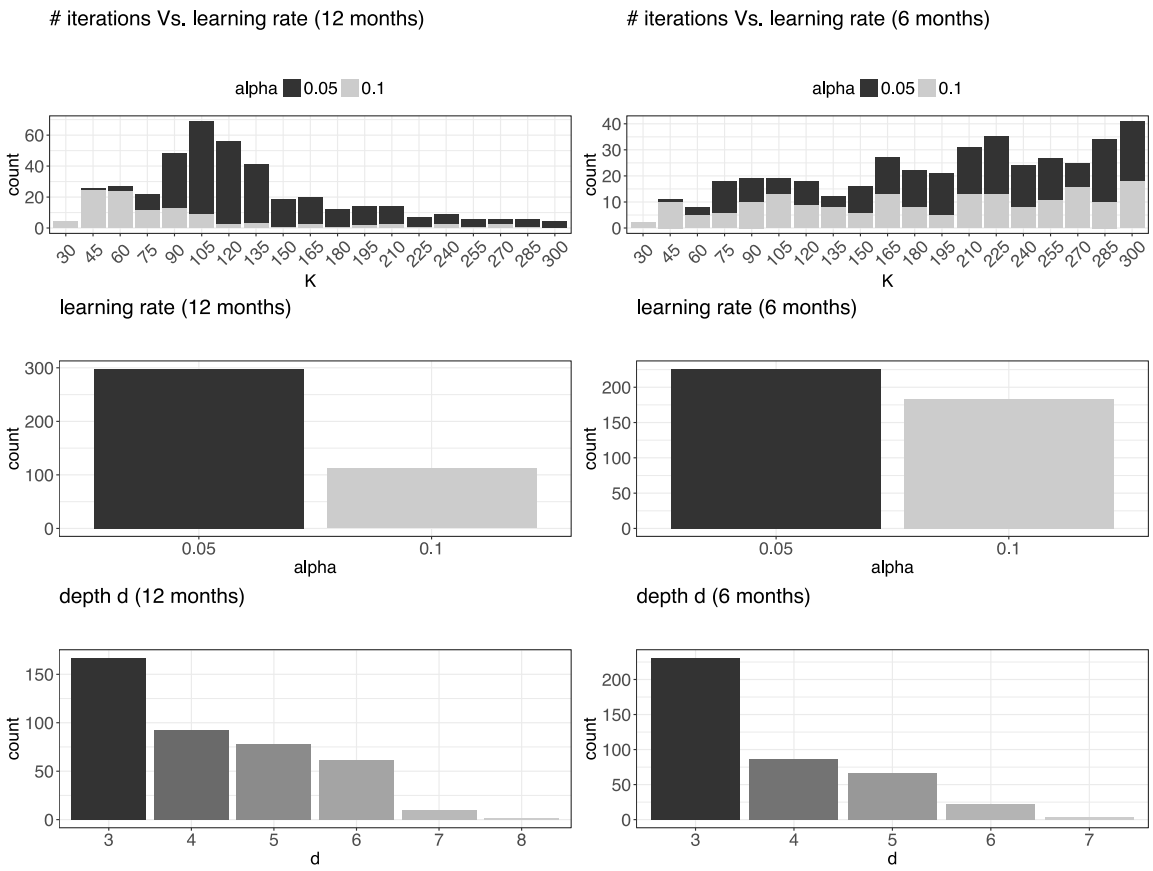


Fig. 5. GBM_day tuned hyper-parameter distribution for the studied group of 410 commercial buildings

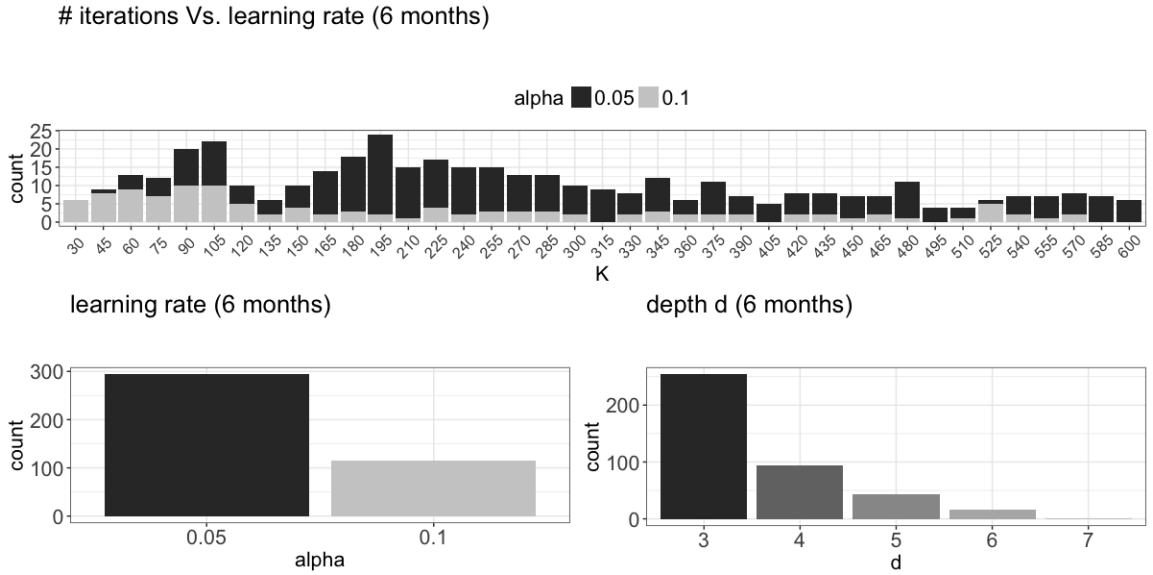


Fig. 6. GBM_day (600) tuned hyper-parameter distribution for the studied group of 410 commercial buildings.

Model	GBM_day			GBM_day (600)		
	25th	50th	75th	25th	50th	75th
R^2	45.22	69.28	83.96	45.21	69.13	83.72
CV(RMSE)	13.17	18.73	27.61	13.12	18.73	27.72
NMBE	-4.41	-0.64	3.16	-4.44	-0.67	3.14

Table 8. Percentiles of the R^2, CV(RMSE) and NMBE for GBM_day models GBM_day **and** GBM_day (600) that respectively have the number of iterations *K* selected within a set spanning from 15 iterations to 300 and a set spanning from 15 iterations to 600 with a granularity of 15 iterations.

5. Conclusions

A baseline modeling method based on a gradient boosting machine (GBM) algorithm was introduced and applied to extensive real commercial building energy consumption data. The performance of this method was compared to the Time-of-Week-and-Temperature (TOWT) model, a publicly available benchmark reference that was accuracy tested in prior work, and to a model based on the RF algorithm, which is representative of a state of the art machine learning method. Overall, the GBM model using a k-fold-blocks CV procedure to tune the hyper-parameters showed higher predictive accuracy than the TOWT and the RF models. The results of this work show that using the GBM model holds promise for increasing the accuracy of whole-building energy savings estimation and related analyses that require future predictions of building energy use. The results also showed that the use of a 6-month of training period for building GBM baseline models generated accuracy results that were just slightly lower than those based on the 12-month training period that is typically used for whole-building M&V applications. This implies that the total length of time required for M&V may be able to be shortened, reducing the total time necessary to conduct a whole-building level savings assessment. However, this finding should be tempered in consideration of the coverage factor of independent variables that can be observed in shorter baseline periods. Specifically, baseline model projections for values of input variables (such as the outside air temperature) that are beyond those observed in the training period may under or over estimate the savings estimates. For example, if a baseline model is constructed with training data that spans 50-75°F, it may not prove reliable in predicting energy consumption for 90°F conditions in the post retrofit period. The recommendation of the ASHRAE Guideline 14 (2014) is to generate savings estimations only for data points of the post period where the input variables are no more than 110% of the maximum and no less than 90% of the minimum values of the input variables used in training period.

The comparison between the different cross validation approaches for hyper-parameter tuning have shown that it is important to consider the time series autocorrelations by using the block random approach rather the standard random approach. Indeed, the results demonstrate that using the standard cross validation decreases the accuracy performance of the GBM algorithm. This is because when the standard k-fold CV approach is applied the observations in the test and training datasets are not independent (due to the intrinsic serial correlation of the interval meter data), which leads the model to over-fit the training data (Opsomer et al. 2001, Bergmeir and Benítez 2012, Bergmeir et al. 2015). It was also shown that the difference of using a week or a day as

block did not have a significant impact on the results, so one can conclude that for the majority of cases, using day as a default block size is a good trade-off. Using the same case study, a comprehensive analysis of the GBM hyper-parameter tuning was performed to demonstrate how the variation of each one of these parameters impacts the accuracy performance of the GBM algorithm.

It is known that one of the biggest practical advantages of using ensemble trees model and more specifically the GBM model (Natekin and Knoll 2013, Ogutu et al. 2011, Caruana et al. 2008) is its flexibility and robustness when used with a large number of input parameters (i.e., a high-dimensional setting). In addition, in comparison to models like the TOWT there is no need to modify the algorithm to handle these additional input parameters, such as building occupancy, humidity, or solar radiation. Rather, one can just include these variables in the algorithm input table without having to define a specific model form for each one of the parameters, as is the case with the majority of standard regression algorithms that are used in practical application today. In addition, the variable selection capabilities of the GBM model allow the inclusion of non-influent parameters without decreasing the predictability of the model. In summary, the GBM model may offer M&V application advantages over commonly used regression models in its ability to maintain accuracy with shorter training periods, improved overall accuracy with respect metrics used in M&V protocols and Guidelines, and ease of incorporation of additional explanatory variables. Moreover, these gains are achieved without significant increases in computation time that could compromise scalability. One potential disadvantage in practical applications is that M&V practitioners are less familiar with machine learning methods than with common regressions. Although the GBM can be delivered to practitioners in packaged routines that preclude the need for manual algorithm refinement, practitioners may be less comfortable using models that they are less familiar with. Key areas of future work will be the application of the GBM model to other energy efficiency related problems such as near future forecasting of energy consumption, continuous anomaly detection, and quantification of demand responsive load reduction.

Acknowledgement

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Building Technologies Program, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

References

Aad, G., et al. 2014. "Search for direct top-squark pair production in final states with two leptons in pp collisions at\ sqrt {s}= 8 TeV with the ATLAS detector." *Journal of high energy physics*, *2014*(6), pp.1–66.

Ahmad, M.W., Mourshed, M. and Rezgui, Y., 2017. Trees vs Neurons: Comparison between Random Forest and ANN for high-resolution prediction of building energy consumption. *Energy and Buildings*.

ASHRAE Guideline 14 (2014). ASHRAE Guideline 14-2014 for Measurement of Energy and Demand Savings. American Society of Heating, Refrigeration and Air Conditioning Engineers, Atlanta, Georgia.

Araya, D.B., Grolinger, K., ElYamany, H.F., Capretz, M.A. and Bitsuamlak, G., 2017. An ensemble learning framework for anomaly detection in building energy consumption. *Energy and Buildings*, *144*, pp.191-206.

Bergmeir, C. and Benítez, J.M., 2012. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, *191*, pp.192-213.

Bergmeir, C., Hyndman, R.J. and Koo, B., 2015. A note on the validity of cross-validation for evaluating time series prediction. *Monash University Department of Econometrics and Business Statistics Working Paper*, *10*, p.15.

Breiman, L. 1996. "Bagging predictors." *Machine learning*, *24*(2), pp.123–140.

Breiman, L. 2001. "Random forests." *Machine learning*, *45*(1), 5–32.

Brown, M., C. Barrington-Leigh, and Z. Brown. 2012. "Kernel regression for real-time building energy analysis." *Journal of Building Performance Simulation*, *5*(4), pp.263–276.

Burkhart, M. C., Y. Heo, and V. M. Zavala. 2014. "Measurement and verification of building systems under uncertain data: A Gaussian process modeling approach." *Energy and Buildings*, *75*, pp.189–198.

Caruana, R., Karampatziakis, N. and Yessenalina, A., 2008, July. An empirical evaluation of supervised learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning* (pp. 96-103). ACM.

Chen, T. and C. Guestrin. 2016. "Xgboost: A scalable tree boosting system." *arXiv preprint arXiv:1603.02754*.

EIA. 2012. Commercial Buildings Energy Consumption Survey. Energy Information Administration, U.S. Department of Energy. http://www.eia.gov/consumption/commercial/reports/2012/energyusage/index.php

Friedman, J. H. 2001. "Greedy function approximation: A gradient boosting machine." *Annals of statistics*, pp.1189–1232.

Friedman, J. H. 2002. "Stochastic gradient boosting." *Computational Statistics & Data Analysis*, *38*(4), pp.367–378.

Friedman, J., T. Hastie, and R. Tibshirani. 2000. "Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors)." *The annals of statistics*, *28*(2), pp.337–407.

Freund, Y. 1995. "Boosting a weak learning algorithm by majority." *Information and computation*, *121*(2), pp. 256–285.

Freund, Y., and R. E. Schapire. 1996. July. Experiments with a new boosting algorithm. In *ICML* 96, pp. 148–156.

Geurts P., D. Ernst, and L. Wehenkel. 2006. Extremely randomized trees. *Machine learning*, *63*(1), pp.3–42.

Goldberg, M., et al. 2015. The changing EM&V paradigm: A review of key trends and new industry developments, and their implications on current and future EM&V practices. Northeast Energy Efficiency Partnerships.

Granderson, J., P. N. Price, D. Jump, N. Addy, and M. D. Sohn. 2015. "Automated measurement and verification: Performance of public domain whole-building electric baseline models." *Applied Energy*, *144*, pp.106–113.

Granderson, J., S. Touzani, C. Custodio, M. D. Sohn, D. Jump, and S. Fernandes. 2016. "Accuracy of automated measurement and verification (M&V) techniques for energy savings in commercial buildings." *Applied Energy*, *173*, pp.296–308.

Granderson, J., Touzani, S., Fernandes, S. and Taylor, C., 2017. "Application of automated measurement and verification to utility energy efficiency program data". *Energy and Buildings*, *142*, pp.191-199.

Heo, Y. and V. M. Zavala. 2012. "Gaussian process modeling for measurement and verification of building energy savings." *Energy and Buildings*, *53*, pp.7–18.

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2013. *An introduction to statistical learning* (Vol. 6). New York: Springer.

Jayaweera, T., and H. Haeri. 2013. "The Uniform Methods Project: Methods for Determining Energy Efficiency Savings for Specific Measures." *Contract*, *303*, pp. 275–300.

Kelso, J. D. 2012. *Buildings energy data book.* Department of Energy.

Kuhn, M., and K. Johnson. 2013. *Applied predictive modeling*. pp. 389–400. New York: Springer.

Liaw A. and Wiener M. (2002). Classification and Regression by randomForest. R News 2(3), 18--22.

Mathieu, J. L., P. N. Price, S. Kiliccote, and M. A. Piette. 2011. "Quantifying changes in building electricity use, with application to demand response." *IEEE Transactions on Smart Grid*, *2*(3), pp. 507–518.

Natekin, A. and Knoll, A., 2013. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, *7*.

Ogutu, J.O., Piepho, H.P. and Schulz-Streeck, T., 2011, May. A comparison of random forests, boosting and support vector machines for genomic selection. In *BMC proceedings* (Vol. 5, No. 3, p. S11). BioMed Central.

Oppel, S., A. Meirinho, I. Ramírez, B. Gardner, A. F. O'Connell, P. I. Miller, and M. Louzao. 2012. "Comparison of five modeling techniques to predict the spatial distribution and abundance of seabirds." *Biological Conservation*, *156*, pp. 94–104.

Opsomer, J., Wang, Y. and Yang, Y., 2001. Nonparametric regression with correlated errors. *Statistical Science*, pp.134-153.

Price, S., A. Mahone, N. Schlag, and D. Suyeyasu. 2011. *Time Dependent Valuation of Energy for Developing Building Efficiency Standards.* Report prepared for the California Energy Commission.

Raftery, Paul, and Tyler Hoyt. 2015. Mave: Software Automated Measurement and Verification. Center for the Built Environment, University of California Berkeley. https://github.com/CenterForTheBuiltEnvironment/mave

Rogers, E. A., E. Carley, S. Deo, and F. Grossberg. 2015. *How Information and Communications Technologies Will Change the Evaluation, Measurement, and Verification of Energy Efficiency Programs*. American Council for an Energy Efficient Economy research report. IE1503.

Satchwell, A., C. Goldman, P. Larsen, D. Gilligan, and T. Singer. 2010. *A Survey of the US ESCO Industry: Market Growth and Development from 2008 to 2011.* Lawrence Berkeley National Laboratory. Report LBNL-3479E.

Sayegh, A., J. E. Tate, and K. Ropkins. 2016. "Understanding how roadside concentrations of NOx are influenced by the background levels, traffic density, and meteorological conditions using Boosted Regression Trees." *Atmospheric Environment*, *127*, pp.163–175.

Schapire, R. E. 1990. "The strength of weak learnability." *Machine learning*, *5*(2), pp.197–227.

Srivastav, A., A. Tewari, and B. Dong. 2013. "Baseline building energy modeling and localized uncertainty quantification using Gaussian mixture models." *Energy and Buildings*, *65*, pp.438–447.

wunderground. The Weather Channel LLC. 2015. Weather Underground. Available at: http://wunderground.com

Zhao, H. X., and F. Magoulès. 2012. "A review on the prediction of building energy consumption." *Renewable and Sustainable Energy Reviews*, *16*(6), pp. 3586–3592.

Appendix

Statistical Metrics to Assess Model Accuracy

To evaluate the effectiveness of a baseline model, several statistical metrics can be used, and these different metrics provide different insights into aspects of accuracy measurement. Relying on just one metric is usually not sufficient to fully understand the weakness and strengths of a specific baseline model. The three metrics that are used in this work are the coefficient of the

determination, or R-squared (R^2); the normalized mean bias error (NMBE); and the coefficient of variation of the root mean squared error (CV(RMSE)). These three metrics provide complementary views of model performance for M&V applications. They also provide a means to assess relative model-to-model comparisons across several buildings simultaneously.

Defined by the equation (3), the R^2 corresponds to the percentage of the energy use variance explained by the model. The R^2 value ranges between 0 percent and 100 percent, with 0 percent indicating that the model explains none of the output variability, and 100 percent indicating that the model explains all the output variability.

$$R^2 = \left[1 - \frac{\frac{1}{n}\sum_i^n (y_i - \hat{y}_i)^2}{var(y)}\right] \times 100 \qquad (3)$$

For the energy baselining application, y_i is the actual metered value, $\hat{y}_i$ is the predicted value, *n* is the total number of data points, and *var(y)* the variance of the actual metered value.

The NMBE (4) is the mean of the error in the predictions divided by the mean of the actual energy use. In other words, it gives a sense of the total difference between model predicted energy use, and actual metered energy use, with intuitive implications for the accuracy of avoided energy use calculations. If the value of NMBE is positive, it means that the prediction of the total energy used during the entire prediction period is lower than the measured value. A negative NMBE means that the prediction is higher. The NMBE is defined in the following equation, where $\bar{y}$ is the average of y_i.

$$NMBE = \frac{\frac{1}{n}\sum_i^n (y_i - \hat{y}_i)}{\bar{y}} \times 100 \qquad (4)$$

The value of NMBE is independent of the timescale for which it is evaluated, which means that the value of the metric will be the same if the timescale is 15-minute, hourly, or daily.

The CV(RMSE) (5) is the root mean square error normalized by the mean of the measured values, which provides a quantification of the typical size of the error relative to the mean of the observations. This metric also gives an indication of the model's ability to predict the overall energy use shape that is reflected in the data. CV(RMSE) is also familiar to practitioners, and is prominent in resources such as ASHRAE Guideline 14. The CV(RMSE) is defined by the equations below, where y_i is the actual metered value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the average of the y_i, and *n* is the total number of data points.

$$CV(RMSE) = \frac{\sqrt{\frac{1}{n}\sum_i^n (y_i - \hat{y}_i)^2}}{\bar{y}} \times 100 \qquad (5)$$

In contrast to the NMBE, R^2 and CV(RMSE) quantify the predictive accuracy at the timescale of the data and prediction; in other words, if the predictions and measured data apply to 15-minute timescales, then this metric summarizes the accuracy in 15-minute predictions.