Doi: http://doi.org/10.5281/zenodo.4560626
Journal of Energy & Technology (JET)
Vol. 1, No. 1, 2021

Roadmap for Utilizing Machine Learning in Building Energy Systems Applications: Case Study of Predicting Chiller Running Capacity for School Buildings Using Stacking Learning

Rodwan Elhashmi*, Kevin P. Hallinan, and Abdulrahman Alanezi
Department of Mechanical and Aerospace Engineering University of Dayton, Dayton, OH
*Corresponding Author's Email: elashmir1@udayton.edu

***Abstract*— Cooling accounts for 12-38% of total energy consumption in schools in the US, depending on the region. In this study, stacking learning is utilized to predict chiller running capacity for four school buildings (regression) and to predict the chiller status for four another schools (classification) using a collection of interval chiller data and building demand. Singular and multiple measurement periods within one or more seasons are considered. A generalized methodology for modeling building energy systems is posited that informs selection of features, data balancing to attain the best model possible, ensemble-based stacked learning in order to prevent over-fitting, and final model development based upon the results from the stacked learning. The results show that ensemble-based stacked learning improves the model performance substantially; providing the most accurate results for both regression and classification. for both classification and regression. For, classification, the balanced accuracy is 99.79% while Kappa is 99.39%. For regression, the R-squared value, the mean absolute error (MAE) error, and the root mean squared error (RMSE) are 1.78 kW, 2.77 kW, and 0.983 respectively.**

Keywords*— *School Buildings; Machine Learning; Cooling Load; Chiller Demand; Regression; Classification; Data-Subset, Imbalanced data; Ensemble Learning

I. INTRODUCTION

High energy use and greenhouse gas emissions are a substantial challenge in the world today. According to the United States Environmental Protection Agency (EPA), the total emission of CO2 in 2017 was 6,457 million metric tons. High energy use and greenhouse gas emission not only effect the environment, but they also effect the economy. For instance, K-12 schools (schools from kindergarten to the 12th grade) in the United States spend $8 billion on energy bills each year; higher than what this sector spends on computers and textbooks combined. Moreover, around 30% of these schools were built before 1960 [1]. For these pre-1960 schools there were no energy code requirements. These schools particularly have more potential for energy efficiency improvements.

Figure 1 shows typical school energy use by category [1]. Of course, the actual disaggregated energy use can vary from school to school; depending upon on many other factors such as climate region, level of reinvestment in energy systems, geometry, schools' activities, appliances and HVAC efficacies, and energy supply resources. Cooling for example, can vary from 12% of total energy consumption in cold and humid climates to 38% of total energy consumption in hot and dry climates [1]. There appear to be substantial opportunities for savings which ultimately could permit reinvestment into improving the education of students. For example, a US EPA reference in 2008 suggested that school retrocommissioning could save a typical 9290.304- Square meters school building between US$10,000 and $16,000 annually, and simple behavioral and operational measures alone can reduce energy costs by up to 25 percent [2]. Similarly, a 2020 US Department of Energy study documented that as much as 30 percent of a district's total energy is used inefficiently or unnecessarily [3].

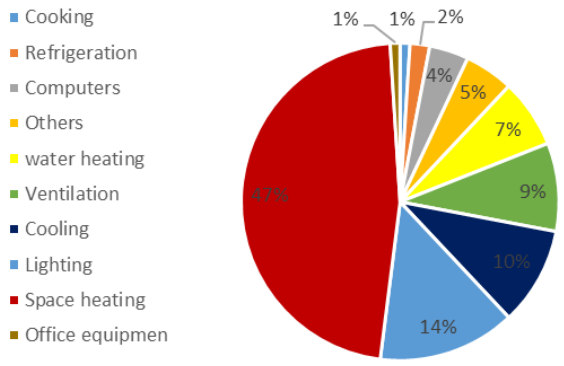


Figure 1: Average energy use for schools in the US [1]

This research addresses only a case-study associated cooling energy component in schools; with a specific focus being to predict chiller running capacity for eight K-12 school buildings using short-term collection of interval chiller data and building demand over singular and multiple measurement periods within one or more season. The intent is to provide a means for energy service companies to quickly assess opportunities for savings from efficiency upgrades and to provide a means for low-cost continuous commissioning of the chiller from interval building demand alone.

But even more importantly, this research provides a general framework for modeling time-varying building energy systems potentially applicable to a myriad of applications and systems. The following provides background particularly relevant to the general modeling framework posed.

II. LITERATURE REVIEW

The problem of estimating cooling and heating loads based on data driven models was first addressed by Kashiwagi and Tobi (1993). The authors implemented a Neural Network to predict heating and cooling loads. For the network model and learning algorithm, Kohonen's Feature Map and Vector Quantization (LVQ) were chosen. The authors used three months of measured data from August 1 to October 31 to build their model. The August data was used for training and the September and October data for testing. The results were promising and showed the feasibility of applying data mining techniques in the energy consumption arena [4]. Similarly, Ushiro et al. (1999) proposed a three-layered neural network to predict the cooling load for the next day. They used a simplified robust filter to eliminate missing data and outliers. Five weeks of data was used to build the model (80% for learning and 20% for testing). The normalized mean squared error (NMSE) of the model was 1.2×10^{-3} [5].

Ben-Nakhi and Mahmoud (2004) employed state of the art building simulation software, ESP-r to simulate the thermal performance of three public buildings and an office building in a very hot climate. The purpose of their study was to forecast next day hourly cooling load before the actual weather data was known using neural networks (NN). Three years of simulated data was used to train the model, one year was used for testing, and one was used for validation. One goal was to generalize the neural networks to fit different buildings. To find the optimum general regression neural networks, six neural networks for each business hour were considered and trained [6]. In a study by Yao et al. (2004), the authors combined four different techniques to predict hourly cooling loads for the next-24-hours. The techniques considered were multiple linear regression (LM), autoregressive integrated moving average (ARIMA), artificial neural network (ANN), and grey model (GM). The weight of each model was evaluated by three criteria: 1) degree of fitting to the historical data, 2) adaptability, and 3) reliability. Analytic Hierarchy Process (AHP) was used to combine and connect these models to enhance the prediction. After the first ten hours of forecasting, a larger error rate was observed [7].

Hou et al. (2006) integrated an artificial neural network (ANN) with rough sets (RS) based on multiple sensors readings to predict cooling load for a building with eleven air-handling units. Hourly data was obtained by averaging the measured five min interval data. The researchers first found the most significant factors contributing to the cooling load and used these as predictors for the ANN model. Since the data was collected from different sources instead of one, the authors developed multiple models to take advantage of redundant data. The optimum principle was used to define the weights of each model and a best model was chosen. The relative error between actual and predicted loads for their best models was within 4% [8].

Li et al. (2009) conducted two different studies to predict hourly cooling load for an office building in Guangzhou, China. The first study was done by utilizing four prediction model. These models are a back propagation neural network (BPNN), a radial basis function neural network (RBFNN), a general regression neural network (GRNN), and a support vector machine (SVM). The inputs variables for a given time were the current and previous hour and previous two hours normalized outdoor temperature; current normalized humidity; and current and previous hour normalized solar radiation. The hourly cooling load was estimated using DeST software. Ultimately, the hourly cooling load and weather data for the month of July was used to train the models. The data of May, June, August, and October were used for testing data. Their results illustrated that SVM and GRNN were slightly better in estimating the cooling load [9] .

The second study was done by applying support vector machine (SVM) and compared their findings with results from a back-propagation (BP) neural network model. Their results showed SVM to have higher accuracy and better generalization. The predictors for their study were normalized outdoor dry-bulb temperature for the current and previous hour and the previous 2 hours, normalized relative humidity, and normalized solar radiation intensity for the current and previous hour. The normalized cooling load was the target. The month of July was used to train the model. May, June, August, and October data were used to the model. Moreover, the simulation software (DeST) was used to calculate the office building's hourly cooling loads [10]. Lixing et al. (2009) implemented a least squared support vector machine (LS-SVM) to predict hourly cooling load using the mySVM software tool. They used current and historical hourly outdoor temperature, humidity, and solar radiation as inputs to predict cooling load. The cooling load was simulated using DeST and the weather data was obtained from the climate database for Guangzhou for a typical meteorology year for the period from May to September. May and June data was used for training and July, August and September data was used to test the model. The results were compared to those obtained from using back propagation neural network (BPNN). LS-SVM seems to perform better accuracy compare to BPNN [11].

Wang et al. (2013) developed a simplified prediction method for a cloud-based continuous commissioning application to predict cooling load. Based on load profile similarity (similarities in occupancy schedule), a reference day for each day was chosen and was used as the initial prediction of the cooling load. They then investigated the correlation among weather data in order to define the most correlated variables. The prediction of these variables was used to calibrate the result of the prediction of the initial load according to the reference day. Lastly, the prediction error of the previous two hours was used to enhance the calibrated load prediction. This method was implemented on a super high-rise building in Hong Kong. The measured data was from a period of time from mid-June to early August in 2011. The root-mean-squared error (RMSE) and the R-squared value the initial load prediction was 0.89 and RMSE 2144 kW respectively. The results of calibrated load prediction were improved. When errors of the past 2 hours were used as predictors, the results showed the best agreement with the actual

data with 0.96 R-squared and 1058 kW RMSE [12]. It should be noted however that for a continuous commissioning application, the errors may be associated with a change in performance and, thus, are a desired result of such modeling and frankly may not be suitable to have as a predictor to improve the quality of the prediction.

Huang and Huang (2013) used Autoregressive Moving Average with Exogenous inputs (ARMAX) model, Multiple Linear Regression (MLR) model, Artificial Neural Network (ANN) model and Resistor–Capacitor (RC) network (a simplified physical model) to predict the cooling load for an office building in Hong Kong. The inputs variables were the previous four-hour cooling load, dry bulb outdoor air temperature, solar horizontal radiation, and room temperature set point. The results show that MLR and ARMAX models have better performance with the smallest mean MBE and mean standard deviation [13].

Sun et al. (2013) applied a general regression neural network (GRNN) with single stage (SS) and double stages (DS) to predict load. In a DS model, the first step is to predict the weather data for the next 24 hours; the second step is to predict cooling load. Two hotels in China were chosen to test and validate the models. These researchers found that the DS method showed some success, but the predictive control system was too difficult due to measuring and predicting many weather data. In comparison, the SS approach was found to be more effective in predicting cooling load [14].

Chou and Bui (2013) employed five different models to predict heating and cooling loads for twelve different building types simulated using the software tool Eco-tect. Support vector regression, artificial neural network, classification and regression tree, chi-squared automatic interaction detector, and general linear regression models were first developed. Then, each model was evaluated and ranked based on its performance. Finally, another model was introduced by combining the two highest ranked models into an ensemble model. The results illustrated that combining support vector regression and an artificial neural network model yielded the highest accuracy in predicting cooling load, while support vector regression alone had the highest accuracy predicting the heating load [15].

Tian et al. (2015) applied an improved multivariable linear regression model to predict the average daily cooling load for an office building. The actual data was obtained by measuring the cooling load for two office buildings in Tianjin Chine for the purpose of validating the model. The first step in their method was to define the most significant weather data variables affecting the cooling load and transfer them into new uncorrelated variables by using principal component analysis (PCA). Secondly, the cumulative effect of high temperature (CEHT) was used to study the effect of higher outdoor temperature on the cooling load. Then, newly measured data was used as feedback by updating the current data point to be used in the next prediction point. The mean absolute relative error MARE was less than 8% [16].

Elhashmi et al. (2021) utilize a short-term chiller and total building demand to predict annual chiller demand. The authors used two data collection scenarios. The first relies upon use of multiple weeks of data collection that includes very warm periods and season transitional periods. The second scenario employs use of contiguous data for a several weeks during only the warmest period of the year. The results show that using two or more separate time periods to envelope most of the weather data yields a much more accurate model in comparison to use of data for only one time period [17]

III. CASE STUDY

This study focuses on eight school buildings located in Dayton, Ohio, USA. The data used here are real time kW and chiller running capacity for each school for five-minute interval for four schools and fifteen-minutes interval for the remaining four schools to provide a record of power and chiller running capacity from June 6, 2018 until October 31, 2018. The data for four of the eight schools shows only the compressor status (ON or OFF). Thus, for these schools, the target is the chiller status (classification), while for the remaining four schools, the target is to predict the actual chiller running capacity (regression). The relatively long period of data collection enveloped the entirety of a cooling season, extending into a period of time where cooling was infrequent. Thus, this data afforded an opportunity to test different periods of time for training data.

IV. OBJECTIVES

The aim of this study is to predict chiller running capacity for eight school buildings and to provide a clear road map for utilizing machine learning in energy engineering applications with that help reduce prediction error and reduce over-fitting from data, algorithm, and process considerations. Real-world data often contains missing values, outliers, and in many cases it is imbalanced. Pre-processing of the data can improve the models developed and enhance the general applicability of the developed models. Algorithmic considerations for machine learning are associated by tuning the model hyperparameter. Process level considerations include the possibility of combining multiple learning algorithms through what is termed ensemble learning, utilizing a combination of techniques with high bias and low variance and low bias and high variance. These considerations also include the estimation of generalization error from cross-validation to ensure the development of a final model that will deliver nearly equal performance on training and validation data (e.g., data not used in the model training). In particular, this research studies the impact of the following on predictive model accuracy: 1) possibility of using a portion of the data to represent the entire population (define statistic tests and apply them); 2) data balance and its impact on model accuracy; 3) use of three types of ensemble machine learning algorithms (bagging, boosting, and stacking); and 4) tuning models through hyperparameter optimization.

V. METHODOLOGY

The process used to predict the chiller status (classification) and to predict the actual chiller running capacity (regression) is summarized in Figure 2. First, possible predictors for the chiller running capacity are hypothesized (feature engineering). The second step is data pre-processing. This includes removing anomalous observations (large spikes), imputing missing values since they are unpredictable, and normalizing data. The third step involves validating the appropriateness of a selected

training data set for developing a model generalizable to a much longer time frame. Here we consider using training data acquired over 20 days and four week intervals. Pearson's chi-square test and Kolmogorov-Smirnov tests are used to validate whether the training data sufficiently represents the longer duration data for which the developed model will ideally be applicable for. This step has to our knowledge not been included in prior studies, but is essential for developing generalizable models from short-term data collection.

The fourth step involves balancing the data for classification models using Under-Sampling, SMOTE, ROSE. Lastly, we build the predictive model using ensemble learning. This approach has rarely been used to for machine-learning based energy modeling. This approach involves modeling in several stages. In the first stage, seven or eight different algorithms for classification and regression are considered independently, with tuning parameters trained and optimized using the whole training dataset. In the second stage, the meta-model is trained on the outputs of the models that have different variable importance. It is worth mentioning here that to avoid overfitting, the base learners' models were selected based on their different structure and their different hyperparameter settings. The third stage involves use of ten-fold cross-validation four times is used to validate each mode. Variable importance shows only how each model utilizes the predictors. Thus, it could be a good indicator that these models work differently and reduce the overfitting.

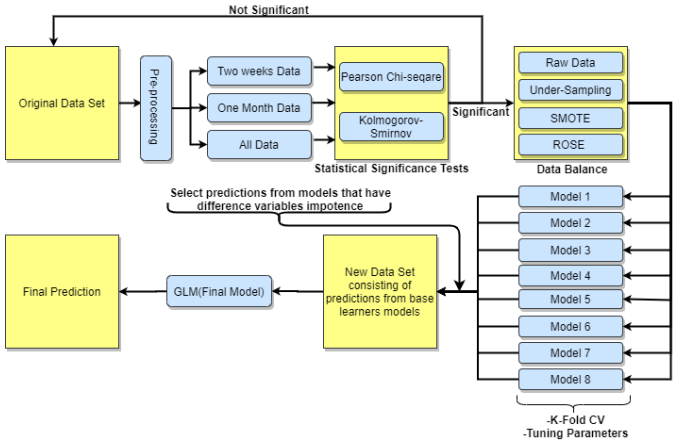


Figure 2: Summary of Methodology Employed

The following provides greater detail about each of these steps.

a. Data-Subsetting:

Training datasets based upon different data collection periods were established in order to establish requirements for training data necessary for developing generalizable models capable of predictions over much longer time frames (ideally yearly or at least seasonally). To successfully address the statistical significance of the subset data and demonstrate appropriate representation by the subset data of the original data, we need to compare each predictor in the subset to the predictor in a complete dataset (taken over a longer period of time). Statistically, this means that the sample feature should have the same probability distribution of the feature in the complete or original population. Pearson's chi-square Kolmogorov-Smirnov tests are common approaches for respectively comparing categorical and numerical variables. Moreover, both tests work under the null hypothesis whereby the subset data is presumed to not be representative of the complete or original population [18].

b. Dealing with Imbalanced data:

Imbalanced data is associated with non-uniform distributions of the features in a dataset [19]. This means the number of observations for one or more classes are significantly higher or lower than other classes. Hence, the performance of a standard algorithm may yield poor predictions of the minority class. In the past years a few solutions have been developed to address the problem of imbalanced data for both data and algorithms. These include: 1) data re-sampling, random oversampling with replacement, random under-sampling, directed oversampling, and directed under-sampling, directed under-sampling; and 2) adjustments in the costs of the various classes, e.g., adjusting the probabilistic estimate at the tree leaf and adjusting the decision threshold [19]. This latter approach means the minority class or low probability density regions of a feature can be weighted more heavily.

The resampling methods used here to handle imbalanced data for classification are under-sampling, synthetic minority over-sampling technique (SMOTE), and ROSE (Random Over-Sampling Examples). In the under-sampling method, observations from the majority class are eliminated until the majority class is equal to the minority class. The drawback of this technique is that removing observations may lead to loss of some information relating to the majority class. The SMOTE methodology is based on over sampling of the minority class by generating synthetic data for a minority class. These synthetic data are created by joining the points of the minority class with line segments and then placing artificial points on these lines. More specifically, for each minority class observation, the algorithm gets its K-nearest-neighbors and a synthetic point anywhere on the line bH-based resampling technique that is used to handle imbalanced data for binary classification problems by producing synthetic examples from a conditional density evaluation of the two classes [20].

c. K-Fold Cross-Validation:

K-fold cross-validation is the most common statistical method for estimating generalization error in predictive models and comparing the performance of different algorithms [21]. The general procedure of creating a predictive model is to train the model on the whole training dataset, and then test the model on one or several validation datasets which were not used for training the model. But in K-fold cross-validation the procedure is different and as described in the following steps [21]:

1) Randomly spilt the dataset into unique k equal folds (subsets).
2) For each unique subset,
   a) train the model using k-1 folds and use F_i for validation.
   b) calculate the model accuracy.

3) Repeat this process until the number of folds is reached.
4) Estimate the final accuracy of the model by averaging all accuracies obtained from all subsets.

This validation approach is especially important in the ensemble learning approach described in the next section.

d. Ensemble Learning

Ensemble learning was originally introduced by Dasarathy and Sheela (1979) [22]. It is a method that combines multiple learning algorithms (models) called base learners to build a robust model that solves a particular problem yielding a more accurate prediction [23] [24]. Furthermore, ensemble learning reduces model errors by creating several learners with relatively similar or fixed bias and then combines the outputs from these learners to reduce the variance [23]. The generalization error in the predictive model can be broken down into following:

Total Error = Bias2 + Variance + Irreducible Error

In this equation, bias is the difference between the model actual mean and the actual mean value of the estimate while the variance represents the change of the target function behavior if different training data was used. Irreducible error is the error that cannot be controlled or reduced. In simple words, bias is the difference between predicted values and the true values while variance is the sensitivity of an algorithm to specific training data sets [21] [23].

The ideal model has low bias and low variance. Nevertheless, the bias and variance seem to have tradeoff relationship. As the bias increases, the variance decreases, and vice versa [23]. Finally, usually models with high variance perform well on training data and poorly on the data beyond the training set, while high bias models perform poorly in predicting both training and testing sets.

The most common methods for ensemble learning are bagging, boosting, and stacking. In bagging, the training dataset is randomly spilt into multiple subsets, then a base model is built for each subset. All base models could run independently in parallel and final predictions are estimated from all the base models [25] [26] [24].

Boosting is a sequential learning process where multiple weak models are created based on correcting the errors of the previous model. The first model is built on a subset from the training dataset where all data points are given equal weights. Then, this model is used to make predictions on the whole training set. Based on the prediction results, a new weight is assigned to the observations that were poorly predicted. Similarly, several models are built and each model learns from the performance of the previous models [27] [26].

Stacking learning is a two-stage procedure. First, different predictive models are built using the same training data. Then, new datasets are created using the outputs from each model. These datasets are then used to build the final model [26]. The procedure of stacking learning is as following:

1. Train each individual base learners (models) on the training set.
2. Predict and test using base learners.
3. Build a new model based on the predictions from the base models (the predicted values from the base model are used for training and testing the final model)

The advantage of this technique is to reduce bias and variance, which leads to an improved predictive model compared to a single model.

Ensemble learning also can be formed by a simple formula such as averaging, majority voting, and weighted average. This can be done by developing several models and then applying a simple formula on the outputs of these models for the final prediction. The advantage of this technique is to improve the overall performance because the prediction decision is made after considering more opinions from different predictive models instead of relying on a single predictive model (same thing can be said about stacking learning). Finally, in general predictive models, a specific percentage of data is used for training, testing, and validating. If this senior is implemented in ensemble learning, where different models stacked together, we will end up having only a single evaluation on our test set for each model and this result could be biased or obtained by a chance. But with 10 fold cross-validation, each individual model is trained and evaluated 10 times. When the results of all 10 folds are consistent and similar or almost similar for each of the folds, then we are confident that our model is robust enough to be generalized and it will have similar performance on new datasets.

e. Evaluate Performance for Classification

The choice of the base learners models is extremely important for achieving the highest accuracy prediction. The eight algorithms used for classification (note: the chilling target in some of the schools are discrete chilling load percentages) as base learners and their tuning parameters are presented Table 1. The selection of these models was based on their different structure and their different hyperparameter settings.

Table 1: Base learners' models and tuning parameters (After R software)

Model		Tuning parameters
Generalized Linear Model	GLM	- No tuning parameters
Recursive Partitioning and Regression Trees	RPART	- Complexity Parameter(cp)
Random Forest	RF	-Number of Randomly Selected Predictors(mtry)
Neural Network	NNET	-Number of Hidden Units (size) -Weight Decay (decay)
Stochastic Gradient Boosting	GBM	-Number of Boosting Iterations -Max. Tree Depth -Min. Terminal Node Size n.minobsinnode
Support Vector Machines with Radial Basis Function Kernel	SVM	-Sigma Cost (C)
Sparse Partial Least Squares	SPLS	-Number of Components (K) -Threshold (eta) -kappa
Naive Bayes	NB	- Laplace Correction (fL) - Distribution Type (usekernel) -Bandwidth Adjustment(adjust)

The goal in assessing model performance is to evaluate its overall accuracy. However, this is much easier said than done.

There is one primary issue in developing some metric that quantifies the overall accuracy. Doing so assumes uniformly distributed class. However, in many cases the class distribution in a dataset is not uniform, or even close to it [28]. For instance, if we have a dataset with two class responses and one class has 10,000 observation and another class has only 200, it is easier for an algorithm to accurately predict the class with the high number of observations then predict the minor class. Thus, reliance upon accuracy alone is not enough to judge the model performance.

Table 2: Confusion matrix

Predicted	Class1	Class2
Class 1	True Positive (TP)	False Positive (FP)
Class 2	False Negative (FN)	True Negative (TN)

$$Overall\ Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (1)$$

One means to ensure accuracy across responses is use of the Kappa statistic (Cohen's Kappa) which was introduced by Jacob Cohen 1960. It measures the agreement between actual and predicted class responses by considering the accuracy that generated randomly according to the frequency of each class Kappa is defined as [29]:

$$Kappa = \frac{acuracy - baseline}{1 - baseline} \quad (2)$$

where the baseline prediction can be calculated as [29]:

$$Baseline = \sum_{i=1}^{k} \frac{(TP + FN) \times (FP + TN)}{TP + FN + FP + TN} \quad (3)$$

This metric measures the agreement between actual and predicted class responses by considering the accuracy that generated randomly according to the frequency of each class [30].

The no-information rate is the accuracy obtained in predicting the majority class [30]. For instance, if the target variable has two classes and one class represents 90% of the data or this example, a model with 90% no-information rate indicates that this model fails to predict any observation from minor class and accurately predicted all the observations for the major class. A 95% confidence interval of a model represents within a confidence level of 95% where a prediction will reside. The sensitivity (true positive rate) of a model characterizes the accuracy of predicting the positive class (event of interest) for all samples having the event. Conversely, specificity (true negative rate) characterizes the proportion of accurately predicting the negative class [30].

$$sensitivity = \frac{TP}{TP + FN} \quad (4)$$

$$specificity = \frac{TN}{TN + FN} \quad (5)$$

Prevalence is the proportion of all positives in in our data [30]:

$$Prevalence\ is = \frac{TP + FN}{TP + FP + TN + FN} \quad (6)$$

The positive predictive value (precision) is the proportion of an accurately predicted positive class to the total positive class. Whereas, the negative predictive value is the proportion of accurately predicting the negative class to the total negative class [30].

$$Positive\ Predictive\ Value = \frac{TP}{TP + FP} \quad (7)$$

$$Negative\ Predictive\ Value = \frac{TN}{TN + FN} \quad (8)$$

Finally, the detection rate is the proportion of the correctly predicted TP to the total number of observations, while detection prevalence is the proportion of the predicted true positive values (TP) to the total number of observations.

f. Evaluate Performance for Regression

Figure 3 shows the actual chiller running capacity (target). Since the target has many zeros associated with non-operation of the chiller system, a two-step machine learning approach is used to solve the regression problem. This methodology is depicted in Figure 4. The first step encodes the target where all nonzero are set to ON and zeros are set to OFF. Then, the data is treated as a classification problem following same procedure described in the previous section. The next step is to build a regression model for all nonzero predicted values. The base learners regression models and their tuning parameters are shown in Table 3.

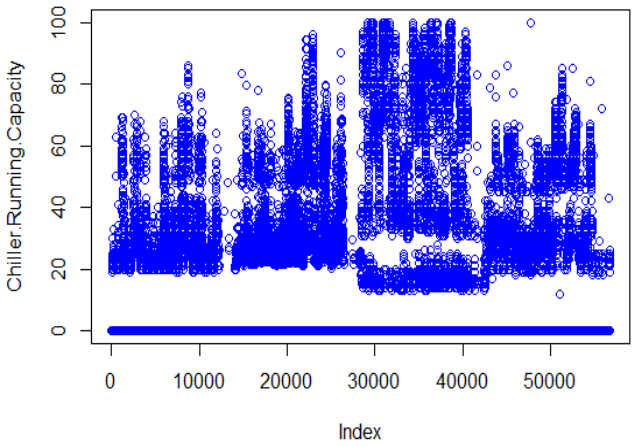


Figure 3: Actual chiller running capacity

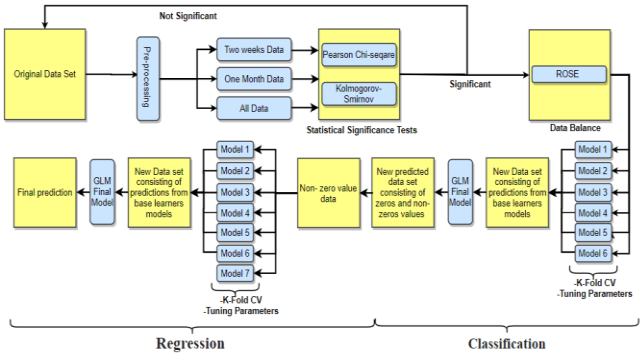


Figure 4: Two-step machine learning approach

Table 3: Base learners regression models and tuning parameters

Model Type		Tuning parameters
Generalized Linear Model	GLM	-No tuning parameters
Recursive Partitioning and Regression Trees	RPART	-Complexity Parameter(cp)
Random Forest	RF	-Number of Randomly Selected Predictors(mtry)
Neural Network	NNET	-Number of Hidden Units (size) -Weight Decay (decay)
Stochastic Gradient Boosting	GBM	-Number of Boosting Iterations -Max. Tree Depth -Min. Terminal Node Size
Support Vector Machines with Radial Basis Function Kernel	SVM	-Sigma Cost (C)
Principal Component Analysis	PCR	-Number of Components (ncomp)

The literature now appears unsettled as to the most appropriate indicator of average model performance. Concerns about both methods has been raised since 2005. Some researchers suggest that RMSE might be a misleading indicator while others suggest RMSE could be more beneficial than MAE [31]. Here, the Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) error metrics are used. Since studying the difference between the RMSE and MAE is beyond the scope of this work, RMSE and MAE will be used together to diagnose the errors and asses the predictive models in regression. Mathematically, RMSE and MAE are defined as [31]:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (9)$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (10)$$

where y_i is the actual value and $\hat{y}_i$ is the predicted value.

VI. Results and Discussion

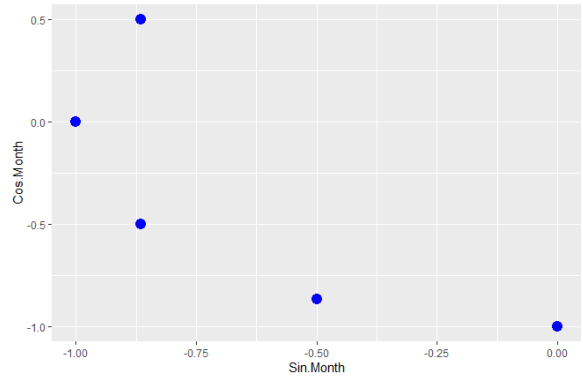
a. *Feature Engineering:*

Time of day, day of the week and month were encoded by adding sin and cos transformations according to:

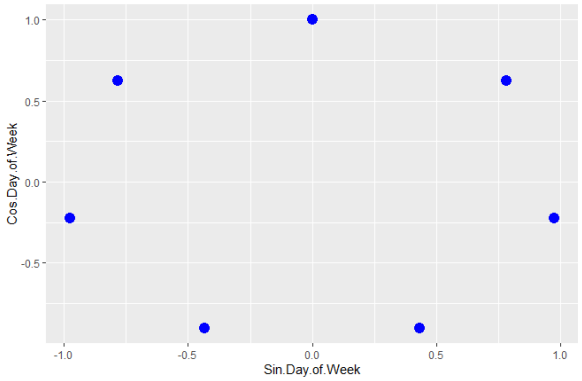
$$\sin of\ variable = \sin(variable * (2\pi/period)), \text{ and}$$
$$\cos of\ variable = \cos(variable * (2\pi/period))$$

The advantage of this transformation is that it provides a full cycle for theses variables as shown in Figure 5 for month, day and hour of day. The month sin and cos graph does not show full cycle because in the data we have only five months. Were we to be looking at 12 months of data, we would expect a full cycle that represents all 12 months.

(a)



(b)



(c)

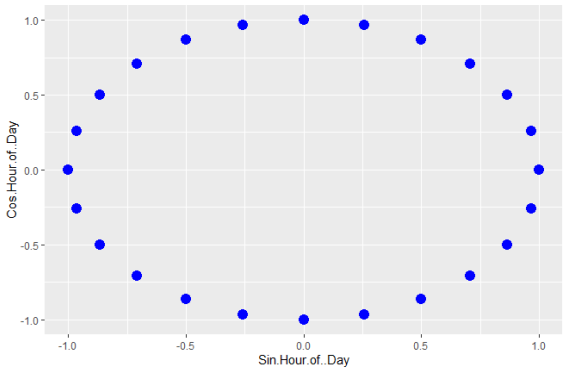


Figure 5: Time transformation for month, day of week, and hour of day

b. Results for classification

i. Evaluating the Suitability of Different Training Periods

The p-values from Pearson's chi-square test and Kolmogorov-Smirnov test for each indicator in both training data subsets (twenty days data and one month data) are illustrated in Table 4. When the indictors in these training datasets (20days and a month) are compared to the indictors in total dataset, we notice that most variables have a low p-value which suggests rejection of the null hypothesis. Thus, it can be concluded these subsets do not represent the original data except for the hour of day features. Both training datasets do not seem to well reflect the complete dataset.

Table 4: p-values for all indictors

Variables	20 Days of Training Data	1 Month of Training Data
	p-values	p-values
Sin day of week	0	0.4982447
Cos day of week	0	0.1402884
Sin month	0	0
Cos month	0	0
Sin hour of day	0.9098922	0.6289666
Cos hour of day	1	1
Previous 24 hours building demand	0	0
Total building demand	0	0
Outdoor air temperature	0	0
Dewpoint	0	0
Outdoor relative humidity	0	0
Chiller running capacity status	8.235395e-35	0.0293871

ii. Checking for data balance for classification

As mentioned previously, the response for the first four of the eight buildings for which we have data is in terms of chiller running capacity status rather than chiller demand. Thus, we know for these buildings if the chiller is ON or OFF (Figure 6). The ratio between ON and OFF is 1: 7.37. This suggests an imbalance of data, leading to potential bias in the predictive models emerging and poor prediction of the minority class.

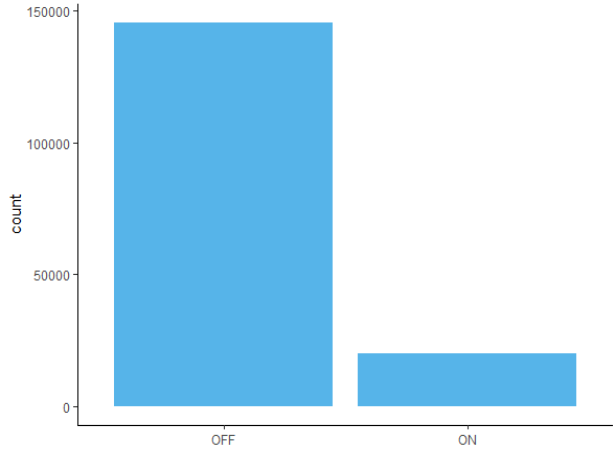


Figure 6: Chiller running capacity

In the next section, models are developed after utilizing the data balancing approaches described in the methodology. Also, it discusses the evolution of all base learners models mode as well as stacking learning for different balancing methods.

iii. Evaluating Individual Model Performance for Different Data Balancing Approaches

The previous section suggests that our data is imbalanced. Therefore, three methods were used to balance the data. Models were developed using the balanced data and compared to each other, as well as the models developed without data balancing. As noted in the methodology sections, three methods of data balancing were evaluated: under-sampling, SMOTE, ROSE. After balancing the data using these techniques, eight base learner models (GLM, RPART, RF, NET, GBM, SVM, SPLS, and NB) were built based using the raw unbalanced data and from data emerging from the three data balancing approaches. The model development involved the stacking of models that have different variable importance (see Section 3.4). The results indicate the SVM, SPLS, and NB have the same variable importance. Therefore, SPLS and NB have been excluded in the stacking stage. In addition, predictions using the models which had differing of variable importance (GLM, RPART, RF, NNET, GBM, and SVM) are used to train the meta model (GLM).

Table 5 provides a brief summary of the performance of the base learners models; while Table 6 shows the results from stacking learning, both results are based on different data reassembling techniques. As Table 6 demonstrates, stacking helps improve the accuracy of all predictions with all types of class balancing. In the end, balanced data using the ROSE technique along with stacking learning provide the best results for predicting both classes. Another observation from Table 5 is that the results utilizing the raw data yields the second-best performance. This is because the ratio between ON and OFF is 1:7.37 does not indicate extreme imbalanced data and some base learner models have the ability to handle some imbalanced data. Moreover, results from under-sampling the majority class appear to be the worst. This probably is because removing observations may lead to loss of information about the majority class. Finally, notably both the total accuracy Kappa value using ROSE balancing approach after stacking yields significantly higher accuracy and Kappa value in comparison to Table 5. Finally, since the best results were obtained after balancing data using the ROSE, Figure 7 illustrates the performance of the base learners models and stacking learning that constructed using ROSE.

c. Results for Regression

As described in the methodology, a two-step machine learning approach is used to solve the regression problem for the remaining four schools for which interval chiller demand data is available. Following the procedure described by Figure 2, the results for each step is shown in the following sub-sections.

i. Data Subsetting

The p-values from Pearson's chi-square test and Kolmogorov-Smirnov test for each indicator in both training data subsets (twenty days data and one month data) are illustrated in Table 7. When the indictors in these training

datasets are compared to the indictors in total dataset, we notice that most variables have a low p-value which suggests rejection of the null hypothesis. Thus, it can be concluded these subsets do not represent the original data except for the hour of day features.

Table 5: Evaluating the base learners models performance

Method		Raw Data	Under-Sampling	SMOTE	ROSE
GLM	Accuracy	0.931547	0.860023	0.881802	0.931524
	Kappa	0.625022	0.514884	0.555132	0.624900
RPART	Accuracy	0.954700	0.912643	0.918122	0.953602
	Kappa	0.776374	0.668116	0.679446	0.772627
RF	Accuracy	0.961796	0.925815	0.948612	0.960087
	Kappa	0.812144	0.709376	0.779543	0.804707
NNET	Accuracy	0.945580	0.900973	0.915722	0.942484
	Kappa	0.724476	0.630118	0.669574	0.707744
GBM	Accuracy	0.955775	0.921815	0.949911	0.956032
	Kappa	0.781312	0.696677	0.780850	0.781155
SVM	Accuracy	0.946306	0.914903	0.937333	0.955363
	Kappa	0.720349	0.675820	0.743051	0.776548
SPLS	Accuracy	0.917573	0.866711	0.890460	0.917188
	Kappa	0.495713	0.525640	0.571377	0.497498
NB	Accuracy	0.913863	0.880949	0.838185	0.903812
	Kappa	0.591722	0.440474	0.468209	0.566659

Table 6: Stacking results (final model)

Criteria	Raw Data	Under-Sampling	SMOTE	ROSE
Accuracy	0.994	0.9625	0.9842	0.9987
95% CI	(0.9936, 0.9943)	(0.9616, 0.9635)	(0.9836, 0.9848)	(0.9985, 0.9989)
No Information Rate	0.8837	0.8535	0.8732	0.8813
P-Value	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Kappa	0.9711	0.8378	0.927	0.9939
Sensitivity	0.9952	0.9939	0.9952	0.9988
Specificity	0.9847	0.7798	0.9088	0.9979
Prevalence	0.8830	0.8535	0.8732	0.8813
Detection Rate	0.8787	0.8483	0.8690	0.8803
Detection Prevalence	0.8805	0.8805	0.8805	0.8805
Balanced Accuracy	0.9899	0.8869	0.9520	0.9983

After rejecting the null hypothesis that these subsets represent the original data, the original data is used to build all models. The first step encodes the target where all nonzero values of the chiller demand are assigned 1; all other observations are set to 0. Then a classification algorithm is used to predict zeros and non-zeros. Since all buildings are K-12 schools and experience roughly the same weather and similar operating hours and conditions, the four schools for which actual chiller power is known are expected to behave similar to the schools for which chiller capacity is known (presented in Classification Results section). Thus, the data balancing, model selection, and model stacking process described in the previous section is followed here.

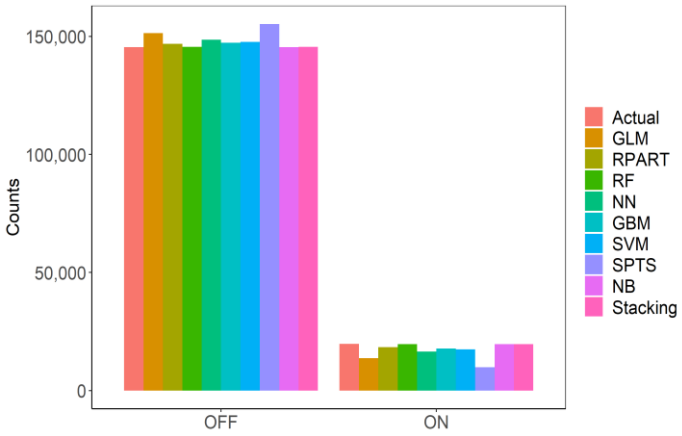


Figure 7: Models performance using ROSE vs. actual chiller running capacity (classification)

Table 7: p-values for all indictors

Variables	20 Days of Training Data	1 Month of Training Data
	p-values	p-values
Sin day of week	0	0.4982448
Cos day of week	0	0.1402885
Sin month	0	0
Cos month	0	0
Sin hour of day	1.331319e-10	7.376722e-11
Cos hour of day	7.376721e-11	1
Previous 24 hours building demand	0	0
Total building demand	0	0
Outdoor air temperature	0	0
Dewpoint	0	0
Outdoor relative humidity	0	0
Chiller running capacity	0.002928241	0.033871312

Tables 8 and 9 summarize the results for individual models (base learners) and stacking learning respectively. After predicting zeros and non-zeros, regression models are built to predict non-zeros values. Only models that have different variable importance and which have acceptable accuracy are selected for stacking learning. Here, the accuracy of the NNET seems way off and thus it is excluded from stacking learning. Table 10 summarizes the results for both base learners models and stacking. Stacking learning scientifically improve the accuracy and reduce the errors. The best individual model in this case is RF which has the lowest MAE and RMSE and the highest R^2 value. Nevertheless, stacking learning has decreased the RMSE by 42% and MAE by 44% relative to this best individual model. Table 9 once again shows a marked improvement in predicting both accuracy and Kappa value if stacking is employed. Finally, A time series plot of the actual chiller running capacity as a function of time for the month of August is shown in Figure 9. The figure compares both the actual and predicted values. It is clear that the two lines representing actual and predicted consumption correspond very well for the stacking compared to all other models.

Table 8: Classification models and their results using ROSE

Method	Criteria	ROSE
GLM	Accuracy	0.8541678
	Kappa	0.7083337
RPART	Accuracy	0.8898505
	Kappa	0.7796982
RF	Accuracy	0.9173758
	Kappa	0.8347597
NNET	Accuracy	0.8999375
	Kappa	0.7998873
GBM	Accuracy	0.9102270
	Kappa	0.8204577
SVM	Accuracy	0.9191084
	Kappa	0.8382210

Table 9: Stacking Results

Method	Criteria	ROSE
Stack GLM	Accuracy	~1
	95% CI	(0.9999, 1)
	No Information Rate	0.5021
	P-Value	< 2.2e-16
	Kappa	~1
	Sensitivity	~1
	Specificity	~1
	Prevalence	0.5021
	Detection Rate	0.5021
	Detection Prevalence	0.5021
	Balanced Accuracy	~1

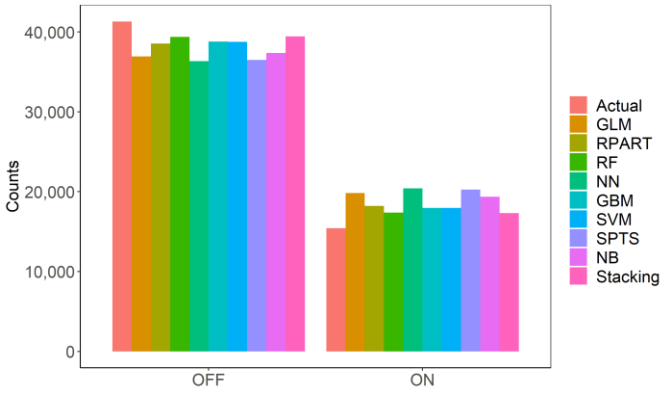


Figure 8: Models performance vs. actual chiller running capacity (first-step in regressing)

Table 10: Results for base learners models and stacking

Method	MAE	RMSE	R^2
GLM	10.59111	13.4486	0.6045816
RPART	4.963909	8.323886	0.8496640
GBM	4.940508	7.470123	0.8784525
PCR	11.89507	14.93703	0.5121906
SVM	4.424521	7.279133	0.8844209
NNET	38.08494	43.67709	0.0417501
RF	3.997286	6.606394	0.9048827
Stack	1.780854	2.7766997	0.9831397

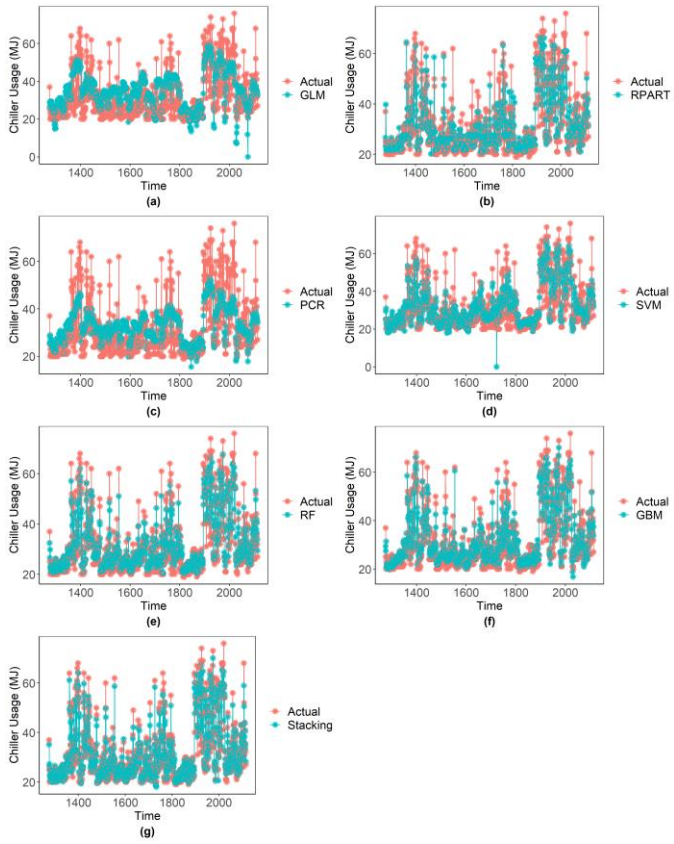


Figure 9: Time series for actual chiller running capacity vs. predicted plots for all models: (a) GLM; (b) RPART; (c) PCR; (d) SVM; (e) RF; (f) GBM; (g) Stacking.

VII. CONCLUSIONS

This study provides a robust generalizable framework for utilizing machine learning in building energy applications for both classification and regression problems. This process begins with the pre-processing of data, which includes removing anomalous observations, imputing missing values, and normalizing data. The next step is to define the appropriate procedure and testing for selecting the training data set and to validate the appropriateness of the training dataset in reflecting the broader dataset. Here, we selected two different data set (20 days and one month). Then we tested these datasets to define whether or not these subsets represent the entire data set using Pearson's chi-square Kolmogorov-Smirnov tests. Almost all variables in subset data have too low of a p-value which suggest rejection of the null hypothesis that these subsets represent the original data. The third step involves data-balancing. Under-Sampling, SMOTE, ROSE are common approaches which could be utilized. The data-balancing is especially important for modeling systems with a preponderance of on-time or off-time. The results demonstrate balancing data using ROSE has significate impact on model accuracy. Finally, the last step involves use of Stacked Learning, developed from individual models based upon several different ML algorithms with different variable importance, different structure, and different hyperparameter settings. Variable importance shows only how

each model utilizes the predictors. Thus, it could be a good indicator that these models work differently and prevent overfitting. The validation results presented here are improved significantly through stacked ensemble learning.

Lastly, the results show significant improvements. A Kappa of 99.39% and an accuracy of 99.79% were achieved in classification, and regression yielded the R-squared value, MAE, and RMSE of 0.9831, 1.78 (kW), and 2.77 (kW) respectively. A stacked learning approach developed from multiple models is based on several different machine learning algorithms with varying variable importance, and different hyperparameter settings. Therefore, it would be an indication that these models work differently and prevent overfitting.

REFERENCES

[1] N. R. E. Laboratory, "Advanced Energy Retrofit Guide- K–12 Schools," U.S. Department of Energy Office of Energy Efficiency and Renewable Energy, 2013.

[2] U. E. P. AGENCY, "Energy Efficiency Programs in K-12 Schools: A Guide to Developing and Implementing Greenhouse Gas Reduction Programs," 2011. [Online]. Available: https://www.epa.gov/sites/production/files/2015-08/documents/k-12_guide.pdf.

[3] E. STAR®, "Schools: An Overview of Energy Use and Energy Efficiency Opportunities," [Online]. Available: https://www.energystar.gov/sites/default/files/buildings/tools/SPP%20Sales%20Flyer%20for%20Schools.pdf. [Accessed 1 11 2020].

[4] K. Norihito and T. Toshikazu, "Heating and Cooling Load Prediction Using a Neural Network System," in International Joint Conference on Neural Networks, 1993.

[5] S. M., U. S., K. K. and O. K., "Cooling load prediction in a district heating and cooling system through simplified robust filter and multi-layered neural network," in IEEE Xplore, Tokyo, Japan, 1999.

[6] A. E. Ben-Nakhi and M. A. Mahmoud, "Cooling load prediction for buildings using general regression neural networks," Energy Conversion and Management, vol. 45, no. 13-14, pp. 2127-2141, 2004.

[7] Y. Ye, L. Zhiwei, L. Shiqing and H. Zhijian, "Hourly cooling load prediction by a combined forecasting model based on Analytic Hierarchy Process," International Journal of Thermal Sciences, vol. 43, no. 11, pp. 1107-1118, 2004.

[8] H. Zhijian, L. Zhiwei, Y. Ye and Y. Xinjian, "Cooling-load prediction by the combination of rough set theory and an artificial neural-network based on data-fusion technique," Applied Energy, vol. 83, no. 9, pp. 1033-1046, 2006.

[9] L. Qiong, M. Qinglin, C. Jiejin, Y. Hiroshi and M. Akashi, "Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks," Energy Conversion and Management, vol. 50, no. 1, pp. 90-96, 2009.

[10] L. Qiong, M. Qinglin, C. Jiejin, Y. Hiroshi and M. Akashi, "Applying support vector machine to predict hourly cooling load in the building," Applied Energy, vol. 86, no. 10, pp. 2249-2256, 2009.

[11] X. Li, J.-h. Lu, L. Ding, G. Xu and J. Li, "Building Cooling Load Forecasting Model Based on LS-SVM," in Asia-Pacific Conference on Information Processing, 2009.

[12] S. Yongjun, W. Shengwei and X. Fu, "Development and validation of a simplified online cooling load prediction strategy for a super high-rise building in Hong Kong," Energy Conversion and Management, vol. 68, pp. 20-27, 2013.

[13] L. Zhengwei and H. Gongsheng, "Re-evaluation of building cooling load prediction models for use in humid subtropical area," Energy and Buildings, vol. 62, pp. 442-449, 2013.

[14] S. Yuying, W. Wei, Z. Yaohua and S. Pan, "Predicting Cooling Loads for the Next 24 Hours Based on General Regression Neural Network: Methods and Results," Advances in Mechanical Engineering, vol. 2013, 2013.

[15] C. Jui-Sheng and B. Dac-Khuong, "Modeling heating and cooling loads by artificial intelligence forenergy-efficient building design," Energy and Buildings, vol. 82, pp. 437-446, 2014.

[16] G. Qiang, T. Zhe, D. Yan and Z. Neng, "An improved office building cooling load prediction model based on multivariable linear regression," Energy and Buildings, vol. 107, pp. 445-455, 2015.

[17] R. Elhashmi, K. Hallinan, and S. Alshatshati, "The Impact of Design Space on the Accuracy of Predictive Models in Predicting Chiller Demand Using Short-Term Data," Journal of Energy & Technology (JET), vol. 1, no. 1, pp. 24-34, 2021.

[18] G. Malato, "How to correctly select a sample from a huge dataset in machine learning," KDnuggets, 5 2019. [Online]. Available: https://www.kdnuggets.com/2019/05/sample-huge-dataset-machine-learning.html. [Accessed 3 12 2019].

[19] N. V. Chawla, N. Japkowicz and A. Kolcz, "Editorial: Special Issue on Learning from Imbalanced Data Sets," ACM SIGKDD Explorations Newsletter, vol. 6, no. 1, pp. 1-6, 2004.

[20] N. Lunardon, G. Menardi and N. Torelli, "ROSE: A Package for Binary Imbalanced," The R Journal, vol. 6, 2014.

[21] T. Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, New York: Springer, 2009.

[22] B. V. Dasarathy and B. V. Sheela, "A composite classifier system design: Concepts and methodology," in IEEE, 1979.

[23] C. Zhang and Y. Ma, Ensemble Machine Learning Methods and Applications, Springer Science & Business Media, 2012.

[24] J. Mendes-Moreira, C. Soares, A. Jorge and J. De Sousa, "Ensemble approaches for regression: A survey," ACM Computing Surveys (CSUR), vol. 45, no. 1, 2012.

[25] L. Breiman, "Bagging predictors," Machine Learning, vol. 24, no. 2, p. 123–140, 1996.

[26] M. Graczyk, T. Lasota, B. Trawiński and K. Trawiński, "Comparison of Bagging, Boosting and Stacking Ensembles Applied to Real Estate Appraisal," in Nguyen N.T., Le M.T., Świątek J. (eds) Intelligent Information and Database Systems, Springer, Berlin, Heidelberg, 2010.

[27] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119-139, 1997.

[28] F. J. Provost, F. E. Tom and R. Kohavi, "The Case against Accuracy Estimation for Comparing Induction Algorithms," in Fifteenth International Conference on Machine Learning, San Francisco, 1998.

[29] P. Czodrowski, "Count on kappa," Journal of Computer-Aided Molecular Design, vol. 28, p. 1049–1055, 2014.

[30] M. Kuhn and K. Johnson, Applied predictive modeling, New York: Springer, 2013.

[31] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)? –Arguments against avoiding RMSE in the literature," Geoscientific model development, vol. 7, no. 3, pp. 1247-1250, 2014.

[32] N. V. Chawla, K. . W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, p. 321–357, 2002.