Predicting the opening state of a group of windows in an open-plan office by using machine learning models

Thi Hao Nguyen, Anda Ionescu, Evelyne Géhin, Olivier Ramalho

Predicting the Opening State of a Group of Windows in an Open-Plan Office by using Machine Learning Models

Thi Hao Nguyen[1], Anda Ionescu[1], Evelyne Géhin[1], and Olivier Ramalho[2]

[1] Univ Paris-Est Creteil, CERTES, F-94010 Creteil, France

{thi-hao.nguyen, ionescu, gehin}@u-pec.fr

[2] Scientific and Technical Center for Building

Champs-sur-Marne 77447, France

{olivier.RAMALHO}@cstb.fr

Abstract. Window operation is among one of the most influencing factors on the indoor air quality (IAQ). The opening state of the windows can modify the air exchange rate and as such the pollutant transfer between indoor and outdoor environments. In this paper, we focus on the modeling of the windows opening state in a real open-plan office with five windows. For this purpose, three machine learning-based models were implemented: (i) Decision Tree, (ii) k-Nearest Neighbors and (iii) Kernel Approximation. IAQ, climatic parameters and the opening state of the windows have been monitored during an entire period of 18 months. The information about: (i) the environmental factors from the previous 24^{th} hour and (ii) the current time (month, day of the week, hour of the day) was used to predict the current state of the windows. The predictor importance estimation and the calculated autocorrelation functions showed that the three most relevant factors were: the previous 24^{th} hour of the windows status, the current time and the previous 24^{th} hour of the prevailing mean outdoor air temperature. The three models perform well with the testing sets according to the different evaluation indicators. The developed methods can be helpful for understanding occupant behavior and also for controlling indoor air pollutants levels in buildings, either as a standalone model or a part of a real-time IAQ monitoring system.

Keywords: indoor air quality · windows opening state · machine learning model · time series · autocorrelation functions · open-plan office

1 Introduction

The outbreak of the COVID-19 virus towards the end of 2019 has left people all around the world with unforgettable memories. This virus rapidly spreads from one to another by interacting in a closed area, which serves as a warning for us to be more concerned about the environment in which we live. According to the World Health Organization (WHO), humans spend more than 90 percent of their time indoors [1]. As a consequence to the lockdown, restricted movement, working from home, and other factors, the time spent indoors increased and research on IAQ became of utmost importance.

The opening state of the windows has an important influence on IAQ, as it can modify the air exchange rate and as such the transfer between indoor and outdoor environments [16]. Opening a window may lead to a sudden increase in the air exchange rate and to both (i) a quick decrease of the concentration of indoors generated pollutant like CO_2 and (ii) a possible increase of the indoor concentration of pollutants coming from outdoors as PM. A research in a mock-up building revealed that the thermal comfort and indoor air quality can be improved by window opening/closing [26]. It is therefore necessary to understand and model the influence of this factor on IAQ.

Window-opening activity is affected by a variety of parameters, such as outdoor temperature, air quality, human presence and season [28, 31, 27]. Occupant's behavior is an important factor but it can vary among individuals [27], leading to different impacts on the indoor environment [28].

On the one hand, theoretical physics-based models (models based on physics rules) struggle to explain the changes in window-opening behavior [9], in the perspective of direct modeling. On the other hand, machine learning models develop computational algorithms designed to simulate human intelligence by learning from their surroundings [13], in the perspective of inverse modeling. Considering the complexity of the underlying relationships, a machine learning model could be a good alternative to a physics-based model and a powerful tool for predicting or forecasting window-opening behavior.

In the last decades, Machine learning (ML) models have been effectively used in the prediction of indoor air quality [37, 7, 23, 28] and energy consumption [2, 12], proving the potential of using machine learning models in indoor environments. Regarding windows opening modeling, a recent study [35] has used the Deep Learning technique for Neural Networks (a specific type of ML) for the detection and recognition of the opening state of the windows by using a camera in order to propose frameworks for energy saving. According to the review paper [9], the common ML models for predicting window-opening behavior include: logistic regression, artificial neural networks (ANN), the Markov chain model, and support vector machines (SVM).

A stochastic window status profile generator (WinProGen) using Markov chains method has been introduced by Calì and colleagues [6]. The model used a database with transition probability matrices from 300 windows in 60 apartments in southern Germany, monitored during 2012 with 1-minute time step. Reliable predictions of buildings' energy performance are obtained when applying these generated window state profiles to the dynamic simulation of two demonstrator buildings. This model has the advantage of appropriately accounting for the process's time dependency. However, according to the authors, this model struggled to deal with a large number of input variables in comparison with the logistic regression method. Therefore, they proposed, as future work, to develop a hybrid model, combing both the Markov chain technique and the logistic regression analysis [6].

Logistic regression [21] is a statistical approach that determines the likelihood of a given event (e.g., opening a window) occurrence based on relevant factor elements (e.g., outdoor/indoor air temperature or PM2.5 concentrations). Most of the research used logistic regression to compute the correlation between the probability of a window opening and the variables of influence [3, 38]. In these two studies, the research was conducted on 19 dwellings in Beijing [38] and 15 residencies in Denmark [3]. Predictive models of the occupants' window opening behavior were established based on multivariate linear logistic regression. This method has the advantage of providing interpretative parameters and could be regularized to minimize over-fitting. However, the model struggles to address the complicated relationships, due to its low flexibility [11].

Other researchers attempted to apply the data-mining approach to discover the effects of the window opening and closing behavior in energy consumption in buildings [10]. This paper proposes a framework for identifying valid window operating patterns, in measured data, by combining logistic regression analysis with two data-mining approaches: (i) cluster analysis and (ii) association rules mining. In this study, 8 non-numerical and 7 numerical variables were used for calculating the probability of opening and closing of a window. In total, a huge quantity of detailed data was used. The authors succeeded to obtain distinct behavioral patterns to serve as a basis for 12 association rules, which classified two typical window opening office user profiles: (i) physical environmental driven and (ii) contextual driven. Based on that, appropriate recommendations for different natural ventilation strategies as well as robust building design could be achieved.

A similar study [22] suggested a generic model that identifies window states using a fully connected feed-forward neural network. For both training and testing processes, this model used around 20 million data samples, which were measured in Germany and USA. The proposed model was evaluated on an additional data set, which was divided into adaptation set and evaluation set. During the adaptation process, the pre-trained weights were adapted by running several tuning iterations, while no hyperparameter tuning or further calibration was required. Based on this procedure, the only required step is the weight adaptation when applied to the other buildings, otherwise, this model did not require any parameter search or calibration. The resulted model could be used by the engineers and designers as a standalone, or as a part of a thermal building simulation.

Six machine learning algorithms were trained in the research of Park et al. [28]. The authors have used monitoring data of 23 sample homes located in Seoul and suburban areas for predicting the occupant's behaviour in the manual control of windows. According to the analysed predictive performance, the k-NN model shows the best fitness with the monitored data set. Regarding the input parameters, the Gini importance score indicated that there are five main driving parameters: (i) prevailing mean outdoor air temperature (PMA), (ii) mean daily temperature, (ii) CO_2 indoor concentration, (iv) relative humidity indoors and (v) the difference between outdoor temperature and the operative temperature indoors.

The Kernel Approximation method has been mainly applied in speech enhancement methods [39]. Regarding the Decision Tree, this method has been used to classify the most important parameters among a large range of variables such as: sociodemographic data, health and lifestyle habits, ergonomic and psychological factors for the Sick Building Syndrome (SBS) [33].

For our study case, we decided to study the ability of different ML classifiers including: Decision Tree, k-NN classification and Kernel Approximation (SVM kernel), to predict the state of the window opening in an open-plan office, as presented hereafter. The reason why we chose Decision Tree is that this method offers the possibility to obtain the extracted rules, which can be applied then for other study cases. Regarding k-NN, this method is recommended as 'a theoretically optimal method of classification' [17]. Finally, we chose Kernel Approximation as it can take into account the non-linearity relationship among the variables. Some information about these three methods is presented in the section 3.

2 Study Case and Features Selection

2.1 The open-plan office

The studied open-plan office is located in the suburban town of Champs-sur-Marne, approximately 30 km East of Paris, France. The office has a total area of $132\,\mathrm{m}^2$ and a volume of $364\,\mathrm{m}^3$. This office is situated on the 2nd floor of the building and occupied by 6 to 15 people, from 8:00 a.m. to 6:00 p.m., from Monday to Friday. The cleaning task is vacuuming, which generally takes place at the end of the week, on Friday, during the end of the day (around 8 p.m.). Figure 1 represents the layout of this office.

The studied building is a relatively modern one, with walls that are around 20 cm thick. It has two floors and several offices, conference rooms, experimental laboratories, etc. Inside and outside the office, measurement devices were installed. The monitoring was performed during 18 months, from January 1$^{\text{st}}$, 2014 to June 30$^{\text{th}}$, 2015. Temperature (T), relative humidity (RH) and carbon dioxide (CO_2) concentration indoors were measured by a Q-Track instrument (TSI Inc.). Particulate matter concentrations in number (PN) were monitored by an optical particle counter (Grimm Dust Monitor 1.108). Concerning the outdoor environment, a permanent weather station located on the roof of the target building automatically recorded the temperature, relative humidity, atmospheric pressure, speed and direction values of wind. It has also detected rainy events. All of the parameters were monitored with 1-minute time-step.

It is possible to calculate the specific humidity (H_s) by calculating first the absolute humidity (H_{abs}) which is based on the relative humidity (RH), the air temperature (T) and the molar mass of the water (M_{water}) and of the air (M_{air}) by using Rankine's formula to approximate the saturated vapor pressure required for the calculation (see Equations (1) and (2)).

$$H_{abs}(\frac{g}{kg}humidAir) = \frac{RH}{100} \times \frac{M_{water}}{M_{air}} \times e^{(13.7-\frac{5120}{T+273.15})} \times 1000 \tag{1}$$

$$H_s(\frac{g}{kg}dryAir) = \frac{H_{abs}}{(1000 - H_{abs})} \times 1000 \tag{2}$$

As it is much easier to obtain the PM (particulate matter in mass concentration) value than the PN one for a real-time model, from the PN concentrations, we calculated the mass fractions of PM2.5 and PM10 according to the method of [8]. The equations (3) and (4) explain how to convert the particle concentrations obtained into mass concentration (μg.m^{-3}) and then calculate the PM2.5 and PM10 fractions. According to [8], we first transform the concentration in number into mass concentration:

$$m(d_{pi}) = C_f \frac{\pi}{6} d_{pi}^3 n(d_{pi}) \tag{3}$$

where i corresponds to the channel number of the optical particle counter, d_{pi} corresponds to the average diameter between the lower and upper limit of the channel, $m(d_{pi})$ is the mass concentration, C_f is the correction factor which corresponds to the particle density and it is fixed at 1 μg.cm^{-3} by default [8] and $n(d_{pi})$ corresponds to the concentration in number. Then, the equation (4) allows the calculation of PM2.5 and PM10 fractions.

$$PM = \sum_{i=1}^{15} m(d_{pi}) f(d_{pi}) \tag{4}$$

where PM corresponds to PM2.5 or PM10 and $f(d_{pi})$ is the fraction of d_{pi} taking into account the collection efficiency of the reference instruments [19]. These contributions can be estimated for each fraction of particles by the equations (5-8) below.

$$f_{PM10}(d_{pi}) = 1 \qquad \text{for} \quad d_{pi} < 1.5\mu m \tag{5}$$

$$f_{PM10}(d_{pi}) = 0.9585 - 0.00408 d_{pi}^2 \qquad \text{for} \quad 1.5 < d_{pi} < 15\mu m \tag{6}$$

$$f_{PM10}(d_{pi}) = 0 \qquad \text{for} \quad d_{pi} > 15\mu m \tag{7}$$

$$f_{PM2.5}(d_{pi}) = [1 + exp(3.233 d_{pi} - 9.495)]^{-3.368} \tag{8}$$

The mean daily temperature and prevailing mean outdoor air temperature (PMA) were calculated using the seven-day weighted running mean outdoor air temperature. According to ASHRAE, equation (9) gives the preferred expression for PMA with "an exponentially weighted, running mean of a sequence of mean daily outdoor temperatures prior to the day in question" [18].

$$\mathrm{PMA} = (1-\alpha)[t_{e(d-1)} + \alpha t_{e(d-2)} + ... + \alpha^6 t_{e(d-7)}] \tag{9}$$

For midlatitude climates, where people are more familiar with synoptic-scale weather variability, a lower value of α could be more appropriate so we chose $\alpha = 0.6$. In Equation (9), $t_{e(d-1)}$ represents the mean daily outdoor temperature for the previous day, $t_{e(d-2)}$ is the mean daily outdoor temperature for two days before, and so on.

The studied office has a permanent mechanical exhaust ventilation. There is no air conditioning and the heating system of the building is a central one. The single flow ventilation system provides a constant air extraction rate of 228 $m^3.h^{-1}$ (measured in 2014 at $\pm$ 6%). Ten air inlets are attached to the joinery of the five sliding windows. These five windows were equipped with contact sensors that detected each opening or closing event and recorded to a local server unit through a wireless zigbee protocol. The main entrance door is equipped with a door contactor. A motion detector was also used to record the occupancy of the office. The collected data is transmitted to and stored on a central server. The monitored window opening states represent time series with irregular time steps. The detection modules send back information as soon as a change of state occurs according to occupants action. Therefore, a pre-processing stage was performed to synchronize all the time series at the same time step (1 minute) [32].

2.2 Features Selection

The data quality and quantity have an influence on the majority of data-driven techniques, including data mining and machine learning. Furthermore, it is important to determine which factors impact the target value (the model output) and how many features (model inputs) can be used to build predictive models. In practice, several environmental factors may influence the accuracy of window opening prediction. However, due to realistic limits, it is impossible to search

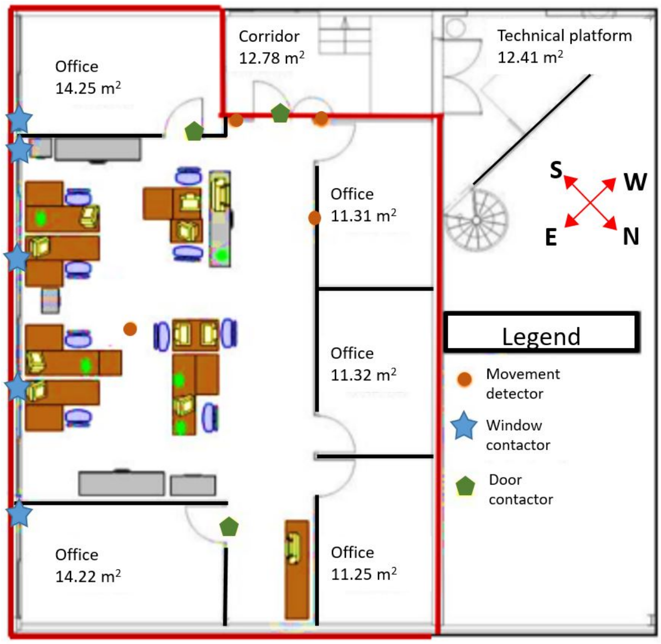


Fig. 1. The studied open-plan office layout.

for all of these features. According to some previous studies, the outdoor temperature, indoor CO_2 concentration and the prevailing mean air temperature were the most important variables in determining the probability of opening/closing windows, followed by indoor air temperature, outdoor and indoor humidity [4, 14, 38, 28].

In addition, non-environmental factors, such as: seasonal change, time of the day and personal preference, also affect the window-opening probability [25]. Thus, in our model, the following variables (features) were used as the initial input selection:

- temperature (T) and specific humidity (Hs) of both indoor and outdoor environments and the prevailing mean outdoor air temperature (PMA);
- indoor CO_2 and indoor particulate matter concentrations (PM2.5 and PM10);
- wind direction, raining condition, door status, occupancy status;
- month, day of the week, hour of the day.

The main statistics of the monitored environmental parameters for the years 2014 and 2015 are displayed in Table 1 and Table 2, respectively. It should be noted that the comparison of these two years is not very representative as 2014 data covered the whole year, and the 2015 monitoring set covered only the first 6 months. However, there are no significant differences between the averaged values of these two years. One can notice that the maximum values of PM2.5 and PM10 concentrations in 2014 are quite higher than those monitored during 2015 (91.87 μg.m^{-3} and 106.78 μg.m^{-3} in 2014 in comparison with 21.3 μg.m^{-3} and 43.71 μg.m^{-3} in 2015). This can be explained by the outdoor pollution episode of particulate matter that happened in March 2014, a quite remarkable event. In addition, higher specific humidity is observed in 2014 compared to 2015, but the monitored data of 2015 does not include July to December.

Table 1. The statistics for environmental parameters of 2014

Features	**Indoor CO_2** (ppm)	**Indoor $PM_{2.5}$** (μg.m^{-3})	**Indoor PM_{10}** (μg.m^{-3})	**Indoor T** (°C)	**Outdoor T** (°C)	**Indoor Hs** (g/kg)	**Outdoor Hs** (g/kg)
Max value	1144.00	91.87	106.78	31.30	35.60	15.11	17.30
Min value	416.80	0.26	0.31	15.00	-4.30	4.28	3.98
Mean value	501.10	2.47	4.32	23.00	13.50	8.88	9.65
Median value	480.50	1.76	3.15	22.40	13.50	8.95	9.66
Std value	64.30	2.87	4.18	2.30	6.00	1.91	2.47

Table 2. The statistics for environmental parameters of 2015

Features	**Indoor CO_2** (ppm)	**Indoor $PM_{2.5}$** (μg.m^{-3})	**Indoor PM_{10}** (μg.m^{-3})	**Indoor T** (°C)	**Outdoor T** (°C)	**Indoor Hs** (g/kg)	**Outdoor Hs** (g/kg)
Max value	1038.82	21.30	43.71	33.33	39.22	13.33	14.94
Min value	421.48	0.13	0.16	18.24	-1.80	3.55	3.48
Mean value	498.45	2.50	4.45	23.10	11.28	6.44	7.11
Median value	477.02	1.93	3.40	22.30	10.30	6.22	6.69
Std value	61.38	2.11	3.70	2.43	7.01	1.55	2.09

In reality, the windows opening status does not change much within a given hour, hence using such a detailed database with a 1-minute time step is not necessary. In addition, some monitored data were missing, therefore we decided to use the hourly average data in this study. Based on the 1-minute time step data, the hourly average values of the selected parameters were calculated as in equation (10). A linear interpolation was applied in order to replace missing values.

$$x_{hourly} = \frac{1}{60} \sum_{i=1}^{60} x_{minute_i} \tag{10}$$

The window opening status for a specific hour was calculated as the mode value (most frequent) of the number of opened windows, according to the equation (11).

$$x_{hourly} = mode(x_{minute_i}) \qquad (0 < i \leq 60) \tag{11}$$

In order to obtain more information about the monitored time series, the autocorrelation functions (ACF) were calculated. The ACF of a time series $Y(t)$ provides a measure of the correlation between y_t and y_{t+k}, where $k = 0, ..., K$ ($k \in \mathbb{Z}$, K is not larger than $T/4$, where T is the total number of observations) and y_t is assumed to be the realization of a stochastic process. According to [5], the autocorrelation r_k for lag k is:

$$r_k = c_k / c_0 \tag{12}$$

where:

$$c_k = \frac{1}{T} \sum_{t=1}^{T-k} (y_t - \overline{y})(y_{t+k} - \overline{y}) \tag{13}$$

and c_0 is the sample variance, $\overline{y}$ is the sample mean of the time series.

The ACF results of the environmental data monitored during 2014 are represented in the Figure 2. Very similar results were obtained for data of the year 2015 so they are not presented here. One can notice the persistence of the temperature (T) and specific humidity (Hs) indoors and outdoors, which means that a value at time t of the temperature or specific humidity is correlated to a value one day later (t+24), two days later (t+48), or even three days later (t+72). In addition, the ACF of the CO_2 concentrations becomes negative and remains at low levels, and then switches back to positive values after a lag of 17 hours. While for outdoor T and Hs (indoors and outdoors), the autocorrelations persist in the positive domain for long delays. In general, temperatures depict the same structures of spectral variability as CO_2: the fundamental frequency peaks at every 24 hours. The ACF of CO_2 alternates sign every 8 hours on a lag of 24 hours. This implies that, instead of using the information of the 'previous hour', in the real-time system, we could use the value of 'the previous 24$^{\text{th}}$ hour' (t-24) environmental data as input for this model, which is easier to access.

Furthermore, the 'weekly periodicity' (at the lag of 168 hours) in the ACF values of CO_2 and PM10 concentration is noteworthy. The information of the 'previous 168th hour' data could be then used as input for the model when the 'previous 24th hour' data is not available. Besides, it can be also noticed that the ACFs of PM concentrations and number of opened windows present high values at a lag of 24 hours (see Figure 2d). We decided to use also the PM concentrations and the number of opened windows, corresponding to the 24 hours lag, as inputs of the prediction model.

In conclusion, non-environmental, environmental features and window status of **the previous 24th hour** moment, were selected as initial inputs of a model built in order to predict the opening status of windows at the **current hour** as presented in the next section.

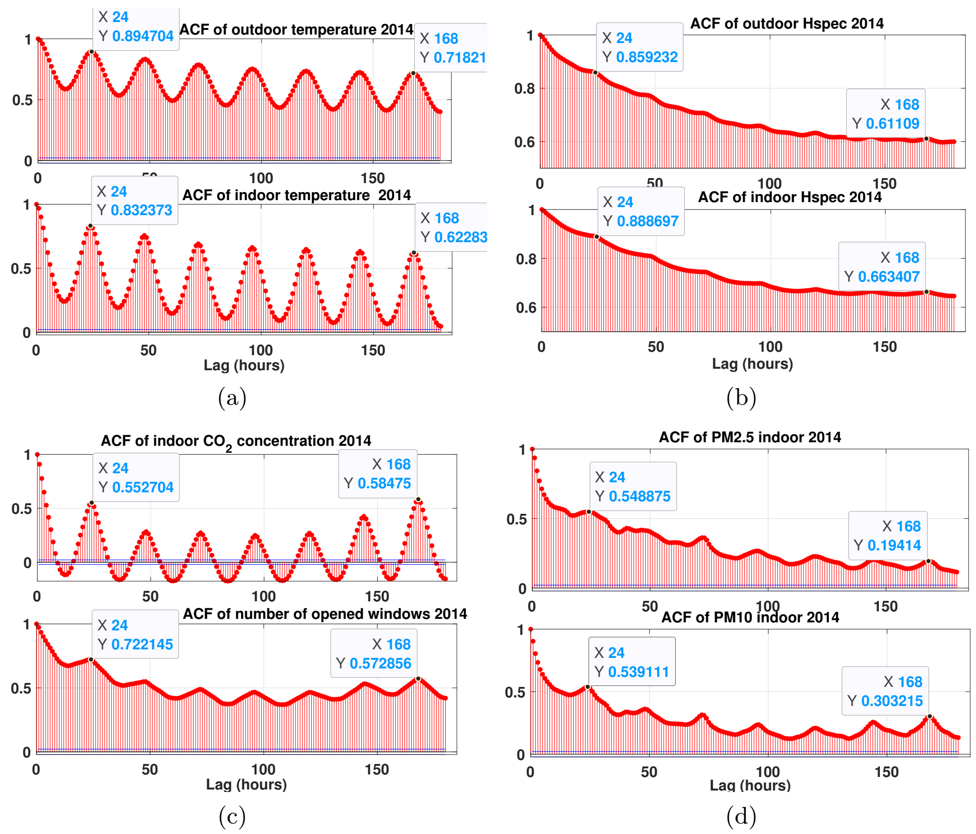


Fig. 2. Autocorrelation values of environmental variables in 2014: (a) Indoor and outdoor temperature, (b) indoor and outdoor humidity, (c) indoor CO_2 and number of opened windows, and (d) indoor PM2.5 and PM10. The 24-hour and 7-day peaks are indicated on the plot of each ACF (X represents the lag and Y represents the ACF value).

3 Modeling implementation

In this section, the different ML models (Decision Tree, kNN classification, Kernel Approximation) are briefly introduced, followed by the data pre-processing and finally the models parameterization.

3.1 Models Description

Decision Tree [29]: Decision Tree is a supervised ML Algorithm that employs a set of rules to make decisions in the same way that people do. Some classification methods, such as Naïve Bayes, are probabilistic, although a rule-based technique is also available. The idea behind Decision Tree is to use dataset attributes to create binary yes/no questions, and then segment the dataset until all the data points from each class become isolated. With this strategy, one can organize the data in a tree structure. A node is added to the tree when a question is asked. Furthermore, the first node is known as the root node. The answer to a question separates the dataset and creates new nodes based on the value of a characteristic. If the process is stopped after a split by some conditions (for example: stop splitting if more than 95% belong to a single class, stop splitting if less than 5 individuals, do not split if the new node has less than 5 individuals, . . .), the final nodes are known as leaf nodes.

The algorithm attempts to partition the dataset into the lowest subset feasible at each split. The aim, like with any other Machine Learning method, is to minimize the loss function as much as feasible [34]. Stochastic Gradient Descent is a popular loss function for classification algorithms. Given that the loss function should be differentiable, it is not possible to use in this circumstance. However, because data points from distinct classes have to be separated, the loss function should assess a split based on the proportion of data points from each class before and after the split. In other words, a loss function that assesses the split based on the cleanliness of the resultant nodes is desirable. Examples of loss functions that compare the class distribution before and after a split are Gini Impurity and Entropy [34].

To summarize, Decision Tree is a rule-based method for solving classification and regression tasks. There is an obvious trade-off between interpretability and performance. A small tree is simple to perceive and comprehend, but it contains a lot of variation. A little modification in the training set can result in an entirely different tree and predictions. A large tree with several splits, on the other hand, produces better classifications. However, it is most likely to remember the training dataset (overfitting).

k-Nearest Neighbor classification [15]: k-Nearest Neighbors models are a type of instance-based model that is used mainly for classification in the Machine Learning field. Its fundamental is as follows: similar objects exist in close proximity. The basic steps of the k-NN algorithm for classification are described below:

1. Load the data
2. Initialize k to your chosen number of neighbors
3. For each sample in the data, calculate the distance between the query sample and the current sample from the data by using distance calculation algorithms (such as Euclidean, Chebyshev, City Block, etc).
4. Return the mode (the value that appears the most often) of k nearest (smallest distance) neighbors.

The k-NN classification is recommended as 'a theoretically optimal method of classification' [17]. However, this method is not easy to interpret and it does not offer the possibility to extract a rules set in order to apply it to another dataset. In addition, the k-NN classification cannot deal with both numerical and categorical data at the same time. It is required to convert numerical data to categorical data.

Kernel Approximation [30]: Kernel approximation is an effective technique for overcoming the low scalability of kernel-based techniques by establishing an explicit mapping ψ: $R^d \rightarrow R^s$ such that $K(x, y) \approx \psi(x)^T \psi(y)$. By doing so, an efficient linear model can be well learned in the transformed space with $O(ns^2)$ time and $O(ns)$ memory while retaining the expressive power of nonlinear methods, where n is the number of samples in the original d-dimensional space and s is the number of features, which is normally a very high number.

The Random Features is one of the most popular techniques to speed up kernel methods in large-scale problems. The Random Kitchen Sinks [30] and Fastfood [36] are two examples of random feature expansions, these schemes tried to approximate Gaussian kernels of the kernel classification algorithm to use for big data in a computationally efficient way. Firstly, they find a random transformation so that its dot product approximates the Gaussian kernel. That is:

$$K(x_1, x_2) = <\phi(x_1), \phi(x_2)> \approx T(x_1)T(x_2)' \tag{14}$$

where $T(x)$ maps x in R^p (p is the number of input features) to a high-dimensional space (R^m). The Random Kitchen Sinks scheme uses the random transformation

$$T(x) = m^{-1/2} exp(iZx')' \tag{15}$$

where $Z \in R^{m\text{x}p}$ is a sample drawn from $N(0, \sigma^{-2})$ and σ^2 is a kernel scale. This scheme requires $O(mp)$ computation and storage.

The Fastfood scheme introduces another random basis V instead of Z using Hadamard matrices combined with diagonal Gaussian scaling matrices.

$$V = \frac{1}{\sigma\sqrt{d}} SHG\mathit{\Pi}HB \tag{16}$$

where $\mathit{\Pi} \in \{0,1\}^{d\text{x}d}$ is a permutation matrix and H is the Walsh-Hadamard matrix. S,G and B are all diagonal random matrices. When the implemented function uses the Fastfood scheme for random feature expansion and uses linear classification to train a Gaussian kernel classification, the model only needs to form a matrix of size $n\text{x}m$, with m typically much less than n for big data, in comparison with support vector machine that requires computation of the $n\text{x}n$ Gram matrix. This random basis reduces the computation cost to $O(m\text{log}p)$ and reduces storage to $O(m)$.

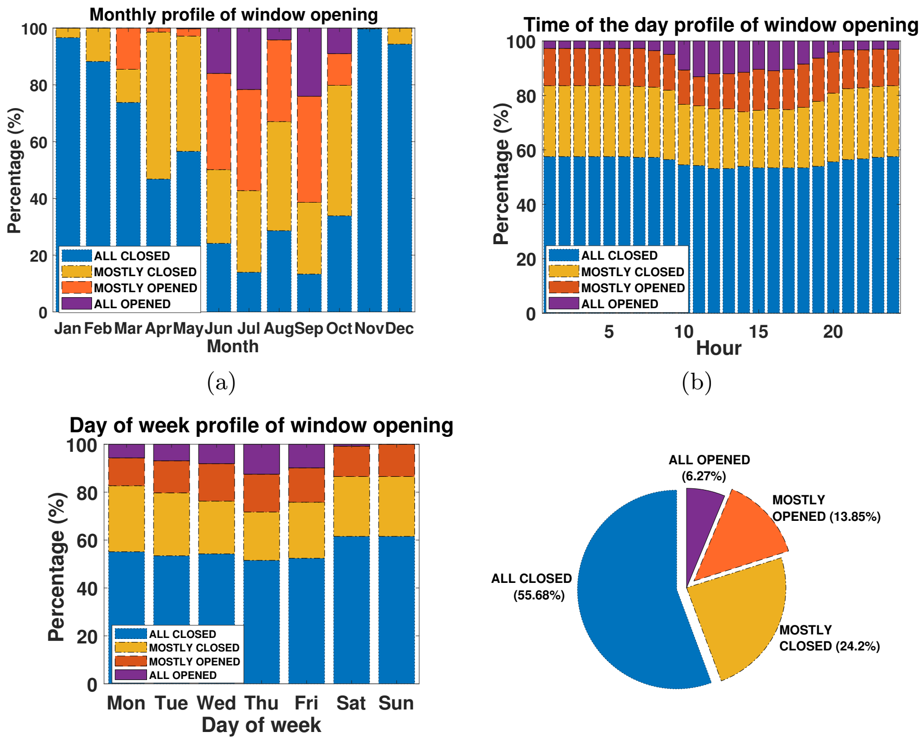
3.2 Data pre-processing

After recalculating the number of opened windows for a specific hour using the mode value (equation (11)), these values were then categorized into four different groups, labeled as follows:

- ALL CLOSED: all of the windows are closed ($x_{hourly} = 0$)
- MOSTLY CLOSED: 1 window is opened ($x_{hourly} = 1$)
- MOSTLY OPENED: 2 or 3 windows are opened ($2 \leq x_{hourly} < 4$)

– ALL OPENED: 4 windows or more are opened ($x_{hourly} \geq 4$)

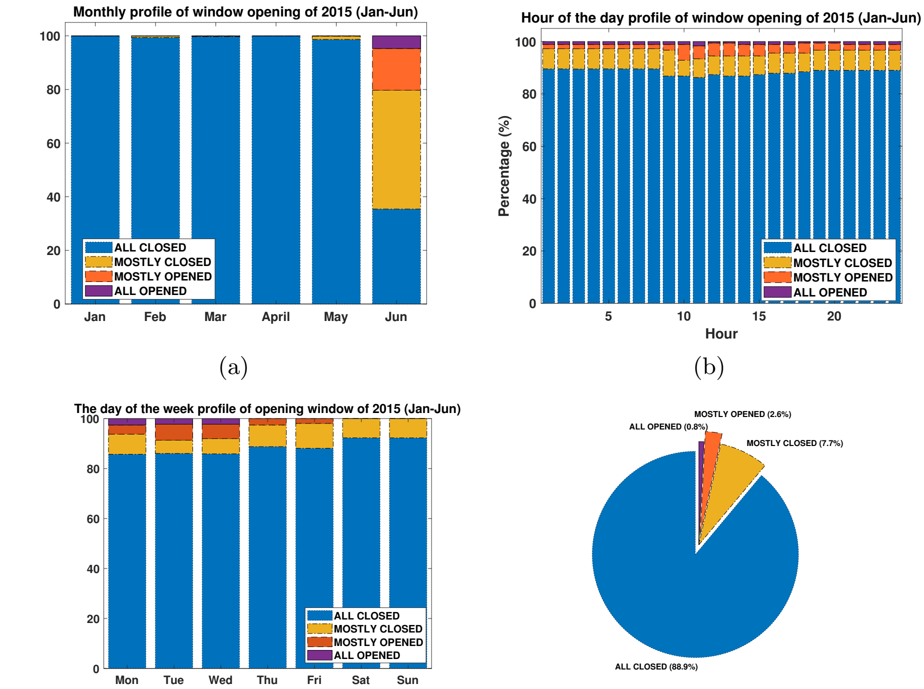
The office is equipped with five windows. In 2015, one window sensor was out of order, thus the respective window remained closed all the time. Therefore, the maximum number of opened windows is five in 2014 and four in 2015. The distribution profiles according to the non-environmental parameters (month, day of the week and hour of the day) and the initial statistics of these four groups during the years 2014 and 2015 are displayed in Figure 3 and Figure 4, respectively.



(a) (b) (c) (d)

Fig. 3. Distribution profile of window opening during 2014 according to the (a) Month, (b) Hour of the day and (c) Day of the week and (d) Statistics for the window opening categories.

Figure 3d shows that in 2014, for more than half of the time (55.68%), the status of this group of windows is ‘ALL CLOSED’. This label is dominant during the winter period (November – March). ‘MOSTLY CLOSED’ and ‘MOSTLY OPENED’ labels are quite equally distributed with 24% and 14%, respectively. The fourth label ‘ALL OPENED’ accounts for just 6.3% of the total time and it appears only in summer and the beginning of autumn (June – October) and during



(a) (b) (c) (d)

Fig. 4. Distribution profile of window opening of 2015 according to the (a) Month, (b) Hour of the day and (c) Day of the week and (d) Statistics for the window opening categories.

the working time (9 a.m. – 6 p.m.). This is expected because "during the working time, the occupants tend to open at least one window, and rarely open the full five windows at the same time" [32].

The statistics for the window opening state according to categories show in 2015 even a higher percentage (88.9%) of the "ALL CLOSED" label. The "ALL OPENED" label is obtained only in June with 0.8% for the 6-month period. The "ALL CLOSED" profile can be observed almost all the time from January to April (Figure 4a). This is quite different in comparison with the distribution profile of the year 2014 without an obvious reason.

Regarding the environmental parameters, Figure 5 represents the mean values and standard deviations of these variables according to the groups. Differences in the mean values of outdoor temperature, specific humidity (indoors and outdoors) and PM10 indoors can be observed for the four windows categories (Figure 5a,b and d). For these parameters, the higher the value, the greater number of windows are opened. For indoor temperature and PM2.5 the differences between groups are small. The indoor mean CO_2 concentration keeps a stable value among these four groups (Figure 5c). Given that the measurement uncertainty is 50 ppm $\pm$ 3% for reading, the range of variation 480-520 ppm is less than the uncertainty. So, we can consider that the CO_2 value does not vary significantly, which means that the office is "well ventilated".

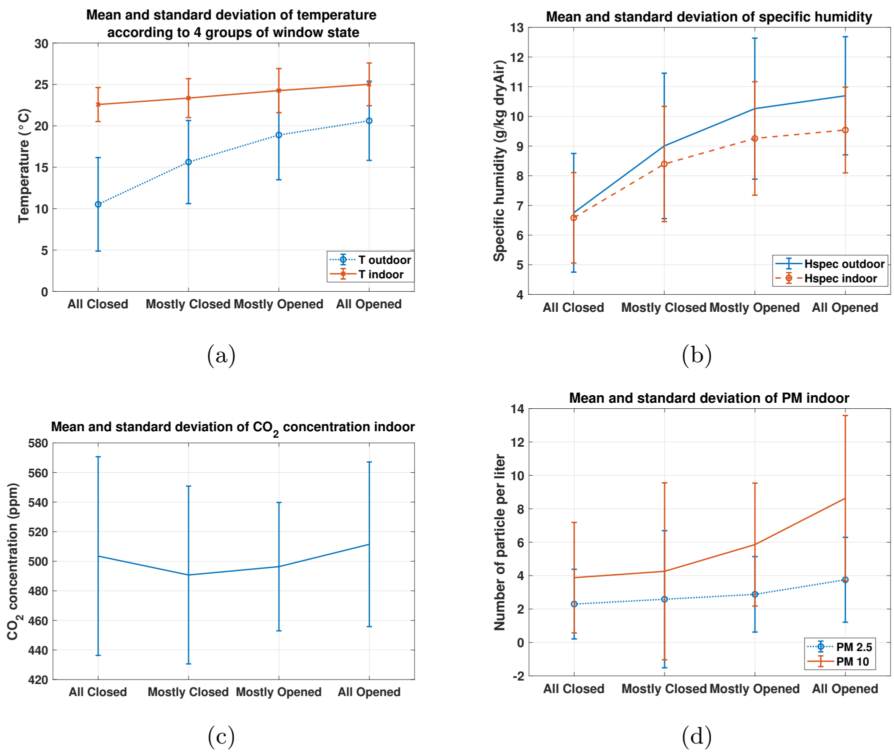


Fig. 5. Statistic profile of 4 groups of window opening during 2014 according to (a) Temperature, (b) Specific humidity (c) CO_2 concentration and (d) PM concentration.

For the model implementation, we need different data sets: training, validation, testing, etc. We decided to divide the time series data into sets of 25 hours and use the 20 first hours for training and validation, and the remaining 5 hours for testing (ratio 80:20 – see Figure 6). The reason why we did not use the day 365th for training is that we need the windows status of this day to evaluate the testing set of the 364th day ('previous 24th hour'). In total, 6980 hours were used for training.

As k-NN method can not deal with numerical and categorical data at the same time, quantitative data had to be recoded to generate qualitative (categorical) data. Numerical data were obtained from environmental parameters monitoring; in order to be transformed into categorical data, the values of each variable were divided into 10 groups (or categories) based on their percentiles in order to equally represent the groups. The first 10 percentiles belong to the first group, the data of percentiles from 11 to 20 belong to the second group, and so on.

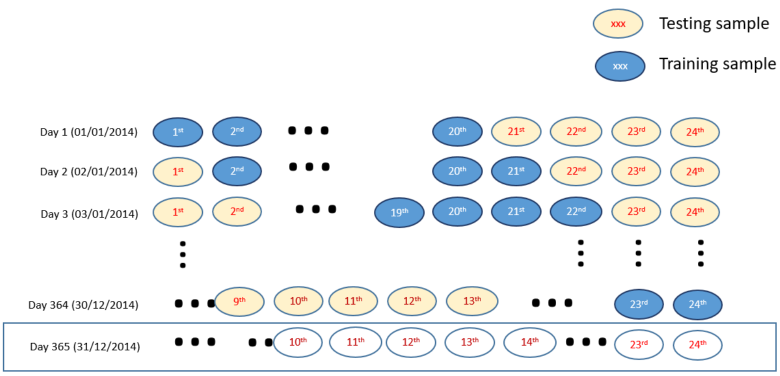


Fig. 6. Figure explaining how we split the data into training and testing sets (sets of every 25 hours).

3.3 Models parameterizations

The Classification Learner application of Matlab® was used for the model development. The 'OptimizeHyperparameters' option for 'all' the input parameters was used to obtain the best values for the hyperparameters of the models and to avoid overfitting. This optimization attempts to minimize the cross-validation loss (error) by varying the parameters. The summary of the obtained values of the different hyperparameters for the three models are presented in Table 3.

The other general parameters of the models are listed below:

- Number of data – training set: 6980 samples (80% data of 2014)
- Number of data – testing set:
  - Testing set of 2014 (which will be called 'test set 2014'): 1745 samples (the rest of 20% data of 2014)
  - Testing set of 2015 (which will be called 'test set 2015'): 4345 samples (data from January to June 2015)

Table 3. Summary of the different hyperparameters for the three models.

Algorithm	Hyperparameter	Value
Decision Tree	Maximum number of Splits	4454
	Split Criterion	deviance
	Minimum leaf size	1
	Tree Depth	16
k Nearest Neighbor	Number of neighbor (k)	3
	Distance metric function	hamming
	Standardize	true
Kernel Approximation	Kernel function	polynomial
	Polynomial Order	3
	Standardize	true

- Data type: hourly averaged data
- Validation method: 10-fold cross validation
- Initial number of input variables: 16 variables as in Table 4:

Table 4. Summary about the input variables for the predicting model.

Idx	Name	Value	Type of data	Moment
1	Month	month	categorical	Current moment
2	DoW	day of the week	categorical	Current moment
3	HoD	hour of the day	categorical	Current moment
4	T_out	outdoor temperature	numerical/categorical[a]	previous 24^{th} hour
5	T_in	indoor temperature	numerical/categorical[a]	previous 24^{th} hour
6	Hs_out	outdoor specific humidity	numerical/categorical[a]	previous 24^{th} hour
7	Hs_in	indoor specific humidity	numerical/categorical[a]	previous 24^{th} hour
8	CO2_in	indoor CO_2 concentration	numerical/categorical[a]	previous 24^{th} hour
9	PM2.5in	indoor PM2.5 concentration	numerical/categorical[a]	previous 24^{th} hour
10	PM10in	indoor PM10 concentration	numerical/categorical[a]	previous 24^{th} hour
11	Prv_Wd	state of group of windows	categorical	previous 24^{th} hour
12	PMA	prevailing mean outdoor air temperature	numerical/categorical[a]	previous 24^{th} hour
13	WindD	wind direction	categorical	previous 24^{th} hour
14	Rain	raining status	categorical	previous 24^{th} hour
15	Occ	occupancy status	categorical	previous 24^{th} hour
16	Door	entrance door status	categorical	previous 24^{th} hour

[a] This variable is coded in 10 categories for the k-NN classification model. For Kernel Approximation and Decision Tree, the monitored numerical data is kept as original.

4 Results and discussion

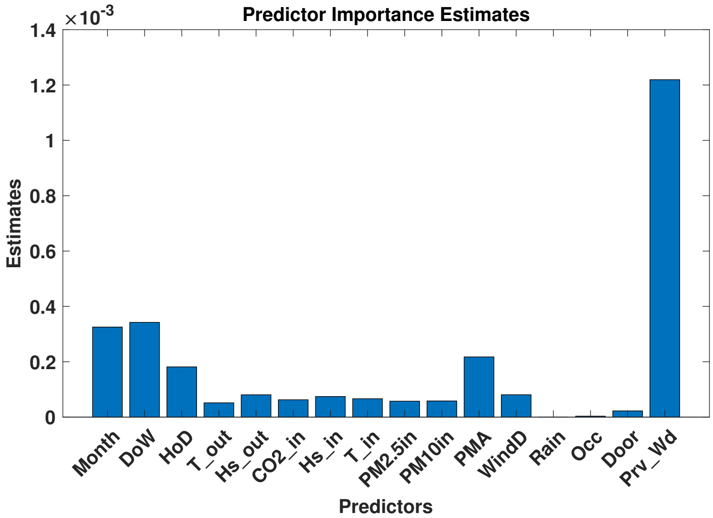
4.1 Rank of the important scores of predictors

Because input variables have a direct influence on the model predictive performance, it is essential to determine which variables are the most important for the model development. The input selection is based on the relevance of the different predictors by evaluating the relative contribution of a given input to the performance of a particular model. This approach is called model-dependent and the advantage of this method is that the input selection is strongly related to the model performance, giving useful information for building predictive models.

Figure 7 shows the relative importance of the factors for window opening status prediction by using the Decision Tree model. Similar results were obtained for the other two methods (k-NN and Kernel Approximation) and will not be presented here. This figure shows the relative significance of the categorical variables (month, day of the week, hour of the day, and the previous 24^{th} hour windows state), as well as the previous 24^{th} hour value of the prevailing mean outdoor temperature outdoors (PMA). According to this observation, these parameters are the most important ones for this modeling. Surprisingly, an important influencing factor - the outdoor temperature, has a small effect on the model's performance. This can be explained by the substantial impact of the specific humidity and PMA, which are calculated using the outside temperature value as in the equations (1) and (9). The rain condition and the status of occupancy show very low importance. Based on this result, we decided to implement the models without these two parameters (Rain and Occupancy). In conclusion, 14 parameters were selected as inputs for our predicting models: Month, DoW, HoD, T_out, T_in, Hs_out, Hs_in, CO2_in, PM2.5in, PM10in, Prv_Wd, PMA, WinD, Door.

4.2 Performance of the window opening state model

Data monitoring starts on the 1^{st} of January 2014 and ends on the 30^{th} of June 2015 (13104 samples-hours). We have decided to use 80% data of the year 2014 for the training and validation set (6980 samples). The remained data was divided into 2 sets for testing: (i) the rest of 20% of the data of the year 2014 (1745 samples) and (ii) data from January 2015 - June 2015 (4345 samples), because we want to observe the different behaviors of the built model when it has to



(a)

Fig. 7. Predictors importance for predicting window opening status for a DT with the input containing all the available parameters. The Month, DoW and HoD correspond to the current moment, all the other variables correspond to the previous 24^{th} hour (see table 4).

deal with data of the same period (the same year 2014) and with data from a completely new period (data of 2015).

Performance of the Decision Tree classifier

Based on the results of the hyperparameters optimization presented in the table 3, a Decision Tree of 541 nodes (Tree Depth = 16) has been obtained after using 80% data of the year 2014 for training and validation, with accuracies of 98.09% and 89.81%, respectively. Using this trained decision tree, we predicted the testing set containing the rest of 20% of the data of 2014 and then we compared it to the monitored values. A value of 86.36% for accuracy (% of well-classified data) was achieved for this test. A confusion matrix of the Decision Tree method for this testing set is displayed in Figure 8a.

As we can see from the figure 8a, the model has a tendency of mislabeling one sample as a 'neighbor label'. The explanation for this could be that the environmental factors change gradually, the 'ALL OPENED' and 'ALL CLOSED' states are easily identifiable, but the 'ALL CLOSED' and 'MOSTLY CLOSED' ones can be ambiguous. The decision tree achieves 910 correct predictions and misses 58 (31+24+3) when the true label is 'ALL CLOSED'; 31 samples

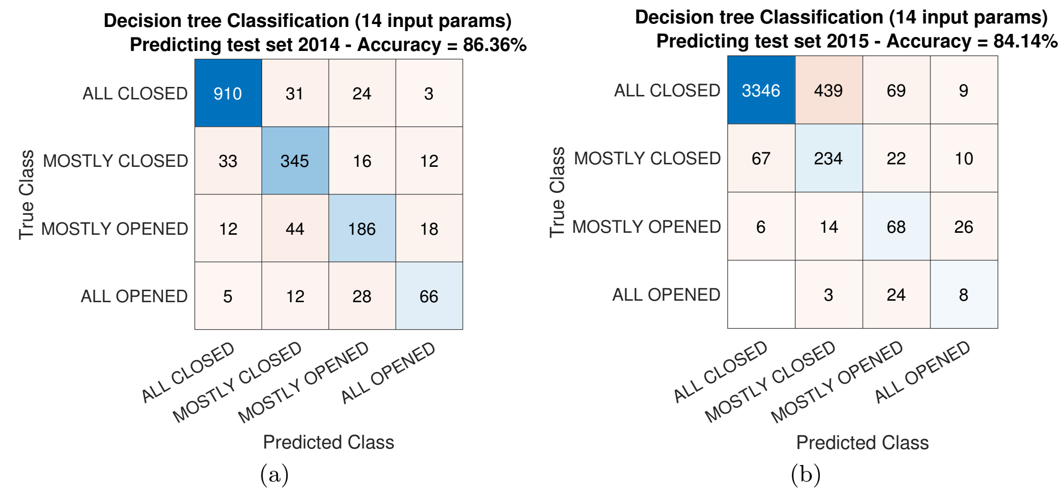


Fig. 8. Confusion matrix of Decision Tree classification (14 input parameters - including information about wind direction and door status) for (a) test set 2014 (1745 samples) and (b) test set 2015 (4345 samples).

were incorrectly predicted to be in the 'MOSTLY CLOSED' state, 24 samples were wrongly labeled as 'MOSTLY OPENED,' and 3 samples were misclassified as 'ALL OPENED.' Similarly, when the true label is 'MOSTLY CLOSED,' 345 samples are properly predicted whereas 61 are incorrectly classified (33+16+12). The labels 'MOSTLY OPENED' and 'ALL OPENED' are accurately predicted in 186 and 66 examples, respectively.

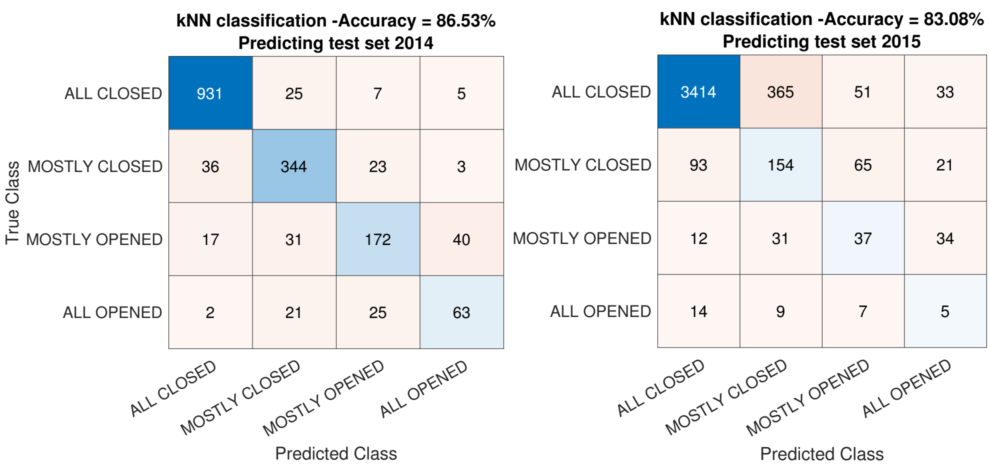
Using the same trained Decision Tree classifier, we predicted the window status of the first 6 months from January to June, of 2015, and compared them to the monitored values. A value of 84.14% for accuracy was achieved.

The confusion matrix for this testing set (data of 2015) is displayed in Figure 8b. Similar to the test set 2014, the true label 'ALL CLOSED' has the highest number of right predictions when the model successfully labeled 3346 samples and mislabeled 517 samples. The label 'MOSTLY CLOSED' also ranks second with 243 accurate samples, and 'MOSTLY OPENED' follows in the third position with 68 correctly classified samples. Specifically, the model can properly identify just 8 samples of the 'ALL OPENED' label while misclassifying up to 24 samples as 'MOSTLY OPENED'. The more detailed evaluation of these confusion matrices will be discussed in the next section.

Performance of the k-NN classifier

Regarding the k-NN classification model, k=3 was obtained after the hyperparameters optimization (see table 3). The achieved accuracies were 99% for training and 92.3% for validation.

The confusion matrix obtained on the test set 2014 is displayed in Figure 9a. This model obtained a value of overall acurracy of 86.53%. From the figure, the highest number of wrong classified belongs to the ”MOSTLY OPENED” label, while 40 samples are wrongly predicted as ”ALL OPENED”. Similar to the Decision Tree model, ‘ALL CLOSED’ label achieved the highest performance, 96.2% sample of this label were correctly predicted (931 corrects from a total of 968 samples). The ‘MOSTLY CLOSED’ label got the second rank with 84.7% correctly predicted samples (344 correct from a total of 406 samples). Finally, the ‘MOSTLY OPENED’ and ‘ALL OPENED’ labels rank the last as they have only 66.2% and 56.8% correct predictions, respectively.



(a) (b)

Fig. 9. Confusion matrix of k-NN classification (14 input parameters) for (a) test set 2014 (1745 samples) and (b) test set 2015 (4345 samples).

Similarly, the confusion matrix for the same trained k-NN model applied on the test set 2015 is represented in Figure 9b.

Same as the Decision Tree model results for the test set 2015, one can observe that a significant number of ”ALL CLOSED” labels are misclassified as ”MOSTLY CLOSED” (365 samples - 9.4%). Eventhough, “ALL CLOSED” label still achieved the highest number of correct classifications (88.4% - 3414 correct predictions out of 3863 total samples). The ”MOSTLY CLOSED” and "MOSTLY OPENED" achieved their ranks as second and third with 46.2% and 32.5%, respectively. The 'ALL OPENED' label, again, got the last position with only 5 correct predictions (14.3%).

Performance of the Kernel Approximation classifier

The polynomial kernel function of order 3 has been obtained after the hyperparameter optimization. In comparison with the two other classification models, when using the Kernel Approximation classifier, the training accuracy results were even lower: only 81.7% for training and 80.6% for validation.

The confusion matrices for Kernel Approximation classifications for the years 2014 and 2015 are displayed in Figure 10. While the accuracy was only 79.3% for the test set 2014, this method achieved up to 92.9% for the test set 2015. Similar to the two other models, this model also has a tendency of mislabeling one sample as a 'neighbor label'. According to the Figure 10, Kernel Approximation misclassified the "MOSTLY CLOSED" as "MOSTLY OPENED" quite a lot (60 samples) and vice versa (58 samples). For the testing set of 2015, the same mistake also was showed when 75 samples were mislabeled as 'MOSTLY CLOSED' and up to 82 samples were wrongly classified as 'MOSTLY CLOSED' instead of 'ALL CLOSED'.

It is interesting to note that the Kernel Approximation method has a different rank of correct predictions among labels in comparison with the two other models for the test set 2015. For test set 2015, the true label 'ALL CLOSED' still has the highest number of right predictions (97.3%), however, the 'MOSTLY OPENED' (42.9%) and 'MOSTLY CLOSED' (36%) labels switched their ranks as second and third, respectively. 'ALL OPENED' label, again, has the last position. Specifically, this method has the highest correct predictions for the label "ALL OPENED" of test set 2015 with up to 15 samples on a total of 33.

4.3 Accuracy statistics for the Decision Tree model

For a deeper analysis of the results, it is necessary to analyse the detailed statistics of the accuracy according to the day of the week, the hour of the day, and the month. We decided to present in this subsection only the results obtained for the Decision Tree model because for the other two models, they keep the same global trend.

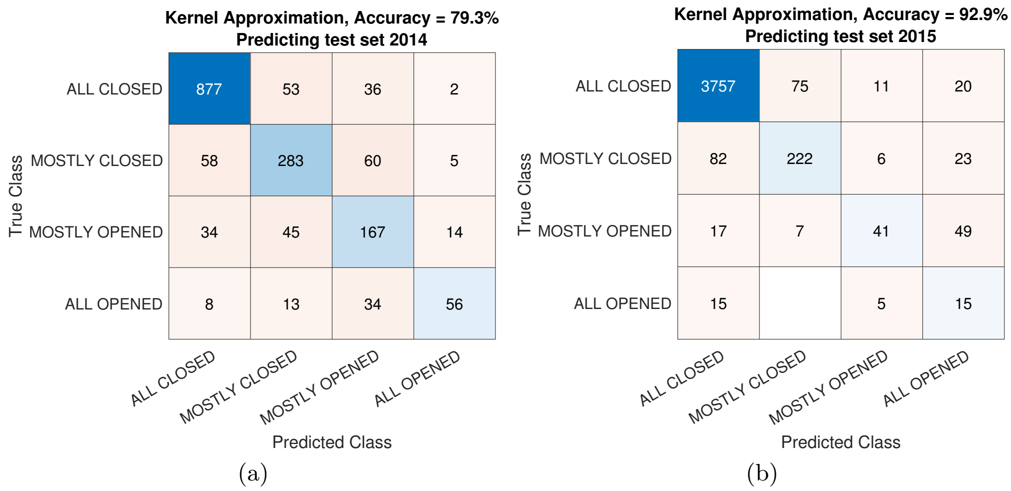


Fig. 10. Confusion matrix of Kernel Approximation classification (14 input parameters) for (a) test set 2014 (1745 samples) and (b) test set 2015 (4345 samples).

The statistics for the test set 2014 are showed in the Figure 11. The highest accuracies were obtained when predicting the windows state for Saturday (100%), winter season (October – February, more than 90%) and night-time periods (8 p.m. – 7 a.m., more than 88% except for the 11 p.m. when maybe the guard round took place). This is expectable because the windows are mainly closed during this time.

The lower accuracy values correspond to the months of the summer season (June – September, around 70%, except for August 79% - the month of vacation), lunch-time periods (12 a.m.– 2 p.m. around 76%) and the 'office leaving' hour (5 p.m. - 73%). In all these periods, there are more changes in the status of the windows and they mostly contain the labels 'ALL OPENED' and 'MOSTLY OPENED'. Interestingly, Tuesday and Sunday have the lowest values of accuracy (around 81%).

Similarly, the statistics for the Decision Tree accuracy for the test set 2015 are presented in Figure 12. The results show that: Saturday (97%), January (98%) and night-time periods (10 p.m. – 7 a.m., around 88%) obtained the highest values of accuracy. In contrast, the lowest accuracy values correspond to the month of June (the only month that has "ALL OPENED" status in 2015), day-time periods (9 a.m.–6 p.m.) and the working days (Monday to Friday). Tuesday, again, has the lowest value of accuracy (only 79%). According to the hour of the day, the prediction accuracy at 5 p.m. is still the lowest (75%), probably because it corresponds to the “office leaving” hours. Some people tend to close the windows before leaving while others leave them opened.

4.4 Evaluation and Discussion

While the Accuracy can be used to evaluate the model’s percentage of well-classified data, Recall and Precision are two other important indicators to evaluate the performance of classification. In addition, the F1 coefficient has been used for evaluating the model’s predictive performance by combining the results from both Recall and Precision. The quality of a classifier can be evaluated by these indicators, which are calculated using the true positive (TP), the true negative (TN), the false positive (FP) and the false negative (FN), based on the equations (17 - 20).

$$Accuracy = (TP + TN)/(TP + FP + TN + FN) \tag{17}$$

$$Sensitivity(Recall) = TP/(TP + FN) \tag{18}$$

$$Precision(F_{rate}) = TP/(TP + FP) \tag{19}$$

$$F1 = 2(Recall)(Precision)/(Recall + Precision) \tag{20}$$

Table 5. Summary about the overall accuracy of the three models

Algorithm	Test set 2014	Test set 2015
Decision Tree	86.36	84.14
k Nearest Neighbor	86.53	83.08
Kernel Approximation	79.30	92.90

Table 5 summarises again the general accuracy values of the three methods: Decision Tree, k-NN and Kernel approximation, when predicting the test set 2014 and the test set 2015. Decision Tree and k-NN obtained quite the same performance achieving similar results for the two testing sets (around 84%). Meanwhile, the Kernel Approximation achieved a significant higher accuracy when predicting the data of 2015. The fact that Kernel Approximation model’s accuracy when predicting the test set 2014 is lower than predicting the test set 2015 can be explained by the particular distribution of labels in 2015 and by the high perfomance of this method for separation in the case of nonlinear problems.

Figure 13 represents the calculated Recall (Sensitivity) values for each state of window opening. For the test set 2014, one can notice that the three models give quite similar results, slightly lower for the Kernel Approximation method. While the highest Recall value is obtained when predicting the 'ALL CLOSED' state of the group of windows ($\approx 90\%$), the lowest value corresponds to the 'ALL OPENED' label ($\approx 60\%$). Similarly for test set 2015, the highest Recall value is still obtained when predicting the 'ALL CLOSED' label (90%) while the lowest belongs to the 'ALL OPENED' label (excepting the Recall value obtained by the Kernel Approximation method for test set 2015, where the lowest value belongs to 'MOSTLY OPENED' label).

Figure 14 and Figure 15 represent the Precision values and F1 scores, respectively. The same situation is obtained for both testing sets. While the highest values are obtained when predicting the 'ALL CLOSED' state, the lowest values correspond to the 'ALL OPENED' label (excepting the Precision value obtained by the Kernel Approximation method for the test set 2014, where the lowest value belongs to 'MOSTLY OPENED' label). For the test set 2015, regarding the Precision values, an even lower value of 5.4% is observed for the 'ALL OPENED' label, by the k-NN model. The reason for which the model's accuracy when predicting the 'ALL OPENED' label was much lower than for the 'ALL CLOSED' label is the particular distribution of labels during the two years. The windows are mainly "ALL CLOSED" and this label is "well learned" by the model. Window opening models are often biased towards the over-represented class where windows remained closed [22].

In general, the Accuracy gave us an overall result without the information about a specific label. Meanwhile, in the case of Recall and Precision indicators we got a detailed accuracy for each label in different perspectives: Precision - How many predicted samples of this label are correct? Recall - How many samples of this label are correctly predicted? From the Figure 13, one can observe the significant differences in Recall values of Kernel Approximation for 'ALL OPENED' label and Decision Tree for 'MOSTLY OPENED' label of test set 2015. Similarly, figure 14 reveals the high differences in Precision values of 'MOSTLY OPENED' and 'MOSTLY CLOSED' label for the test set 2015 when using the Kernel Approximation. However, when we calculated the F1 values, these differences were smaller.

Overall, the Decision Tree method appears to be the best classification model, with the best balance of Recall, Precision and F1 values regarding the four labels. Kernel Approximation occasionally achieved the highest evaluation values (particularly for the test set 2015 for 'ALL CLOSED' and 'ALL OPENED' labels). This can be explained by its high perfomance in separation in the case of nonlinear problems. However, the overall accuracy for the test set 2014 of this method is slightly lower in comparison with the two other methods. In addition, Decision Tree also provides the list of classification rules (export in .txt file), which can easily be used to apply for new data. Regarding the kNN model, the low values of these evaluation indicators could be explained by the fact that this method has been applied on categorical data for all the parameters, by contrast to the other methods, which allow the both types of inputs (numerical and categorical). This decoding operation probably leads to a loss of information.

5 Conclusion and Future work

In conclusion, in this study, we have obtained three ML classification models to predict the opening state for a group of windows in an open-plan office. To select the appropriate set of features, the ACF values and predictor importance estimates were calculated. In our case, the most pertinent inputs were: the previous 24^{th} hour state of the windows (which can be related to the personal preferences of the occupants), the day of the week, the month, the hour of the day (which can be related to the occupancy and the personal preferences) and the previous 24^{th} hour of the prevailing mean outdoor air temperature (outdoor environment condition). The models were then established by using these important parameters completed with the 'previous 24^{th} hour' of the following variables: the wind direction, entrance door status, indoor CO_2 and particle matter (PM2.5 and PM10) concentrations, as well as both indoor and outdoor temperatures and specific humidity. Validation tests have been used to compare the outputs of the models and the measured windows states obtained in the years 2014 and 2015 in the open-plan office. According to the different evaluation indicators, the results show that all the three models perform well with the testing sets.

In the future, we can improve the over-represented 'ALL CLOSED' label by resampling in order to have an unbiased data set or by providing different weights for each label to penalize misclassification. In addition, with an algorithm that combines multiple trees and control for bias or variance, like Random Forests [20] or Gradient boosted trees [24], the Decision Tree model could have a better performance. For the k-NN model, an efficient method to deal with both the numerical and categorical data in order to avoid the loss of information needs to be further investigated. Furthermore, the high performance of Kernel Approximation approach - a good nonlinear separator, is also noteworthy.

We could then use one of the three developed models as a standalone, or as a part of a real-time IAQ monitoring system, in order to optimize the action to be taken to reduce the exposure of the occupants.

References

1. Report to congress on indoor air quality: Assessment and control of indoor air pollution. Tech. rep., U.S. Environmental Protection Agency (1989)
2. Amasyali, K., El-Gohary, N.M.: A review of data-driven building energy consumption prediction studies. Renewable and Sustainable Energy Reviews **81**, 1192–1205 (2018). https://doi.org/10.1016/j.rser.2017.04.095
3. Andersen, C., Bro, R.: Practical aspects of parafac modeling of fluorescence excitation-emission data. Journal of Chemometrics **17**, 200 – 215 (04 2003). https://doi.org/10.1002/cem.790
4. Andersen, R., Fabi, V., Toftum, J., Corgnati, S.P., Olesen, B.W.: Window opening behaviour modelled from measurements in danish dwellings. Building and Environment **69**, 101–113 (2013)
5. Box, G., Jenkins, G.M., Reinsel, G.: Time Series Analysis: Forecasting and Control. 3rd ed. Englewood Cliffs, NJ: Prentice Hall (1994)
6. Calì, D., Wesseling, M.T., Müller, D.: Winprogen: A markov-chain-based stochastic window status profile generator for the simulation of realistic energy performance in buildings. Building and Environment **136**, 240–258 (2018)
7. Chen, S., Mihara, K., Wen, J.: Time series prediction of co2, tvoc and hcho based on machine learning at different sampling points. Building and Environment **146**, 238–246 (2018). https://doi.org/10.1016/j.buildenv.2018.09.054
8. Cheng, Y.H., Lin, Y.L.: Measurement of particle mass concentrations and size distributions in an underground station. Aerosol and Air Quality Research **10**, 22–29 (02 2010). https://doi.org/10.4209/aaqr.2009.05.0037

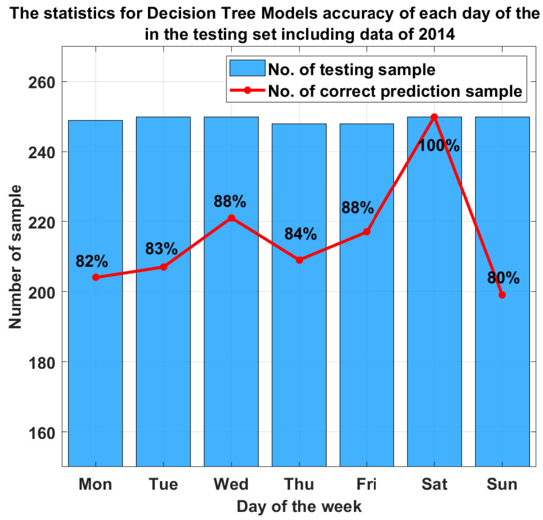
9. Dai, X., Liu, J., Zhang, X.: A review of studies applying machine learning models to predict occupancy and window-opening behaviours in smart buildings. Energy and Buildings **223**, 110–159 (2020)
10. D'Oca, S., Hong, T.: A data-mining approach to discover patterns of window opening and closing behavior in offices. Building and Environment **82**, 726–739 (2014)
11. Dreiseitl, S., Ohno-Machado, L.: Logistic regression and artificial neural network classification models: a methodology review. Journal of Biomedical Informatics **35**(5), 352–359 (2002). https://doi.org/10.1016/S1532-0464(03)00034-0
12. Edwards, R.E., New, J., Parker, L.E.: Predicting future hourly residential electrical consumption: A machine learning case study. Energy and Buildings **49**, 591–603 (2012). https://doi.org/10.1016/j.enbuild.2012.03.010
13. El Naqa, I., Murphy, M.J.: What Is Machine Learning? Springer International Publishing (2015)
14. Fabi, V., Andersen, R., Corgnati, S., Olesen, B.: Occupants' window opening behaviour: A literature review of factors influencing occupant behaviour and models. Building and Environment **58**, 188–198 (2012)
15. Fix, E., Hodges, J.L.: Discriminatory analysis : nonparametric discrimination, consistency properties. USAF School of Aviation Medicine (1951)
16. Godish, T., Spengler, J.D.: Relationships between ventilation and indoor air quality: A review. Indoor Air **6**(2), 135–145 (1996). https://doi.org/10.1111/j.1600-0668.1996.00010.x
17. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics, Springer New York Inc., New York, NY, USA (2001)
18. of Heating Refrigerating, A.S., Engineers, A.C.: Thermal environmental conditions for human occupancy. Tech. rep., American Society of Heating Refrigerating and Air-Conditioning Engineers (2017)
19. Hinds, W.C.: Aerosol Technology: Properties, Behavior, and Measurement of Airborne Particles. Wiley (1999)
20. Ho, T.K.: Random decision forests. p. 278–282 (07 2016)
21. Hosmer, D., Lemeshow, S.: Applied Logistic Regression. Hoboken, vol. 354 (01 2000). https://doi.org/10.1002/0471722146
22. Markovic, R., Grintal, E., Wölki, D., Frisch, J., van Treeck, C.: Window opening model using deep learning methods. Building and Environment **145** (2018), 10.1016/j.buildenv.2018.09.024
23. Martínez-Comesaña, M., Eguía-Oller, P., Martínez-Torres, J., Febrero-Garrido, L., Granada-Álvarez, E.: Optimisation of thermal comfort and indoor air quality estimations applied to in-

use buildings combining nsga-iii and xgboost. Sustainable Cities and Society **80**, 103723 (2022). https://doi.org/10.1016/j.scs.2022.103723

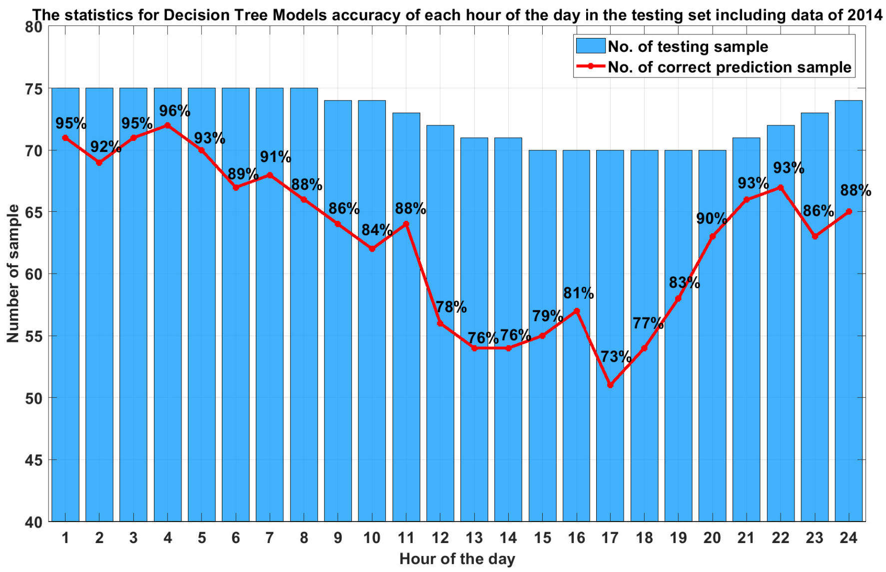
24. Natekin, A., Knoll, A.: Gradient boosting machines, a tutorial. Frontiers in neurorobotics **7**, 21 (12 2013). https://doi.org/10.3389/fnbot.2013.00021
25. Pan, S., Xiong, Y., Han, Y., Zhang, X., Xia, L., Wei, S., Wu, J., Han, M.: A study on influential factors of occupant window-opening behavior in an office building in china. Building and Environment **133**, 41–50 (2018)
26. Park, J.: Long-term field measurement on effects of wind speed and directional fluctuation on wind-driven cross ventilation in a mock-up building. Building and Environment **62**, 1–8 (2013). https://doi.org/10.1016/j.buildenv.2012.12.013
27. Park, J., Choi, C.: Modeling occupant behavior of the manual control of windows in residential buildings. Indoor Air **29** (11 2018). https://doi.org/10.1111/ina.12522
28. Park, J., Jeong, B., Chae, Y.T., Jeong, J.W.: Machine learning algorithms for predicting occupants' behaviour in the manual control of windows for cross-ventilation in homes. Indoor and Built Environment **30**(8), 1106–1123 (2020). https://doi.org/10.1177/1420326X20927070
29. Quinlan, J.R.: Induction of decision trees. Machine Learning **1**, 81–106 (1986). https://doi.org/10.1007/BF00116251
30. Rahimi, A., Recht, B.: Random features for large scale kernel machines. Advances in Neural Information Processing Systems **20**, 1177–1184 (01 2008)
31. Raja, I.A., Nicol, J., McCartney, K.J., Humphreys, M.A.: Thermal comfort: use of controls in naturally ventilated buildings. Energy and Buildings **33**(3), 235–244 (2001). https://doi.org/10.1016/S0378-7788(00)00087-6
32. Ramalho, O., Ouaret, R., Ionescu, A., Le Ponner, E., Candau, Y.: Tribu - suivi dynamique en temps réel de la qualité de l'air intérieur dans un environnement de bureaux. contributions des sources et modèle prévisionnel rapport, primequal apr eiai / projet tribu. Tech. rep., CSTB (2016), https://www.primequal.fr/sites/default/files/tribu_rf.pdf
33. Sarkhosh, M., Najafpoor, A., Alidadi, H., Shamsara, J., Amiri, H., Andrea, T., Kariminejad, F.: Indoor air quality associations with sick building syndrome: An application of decision tree technology. Building and Environment **188** (2021)
34. Tan, Pang-Ning, Steinbach, M., Adeyeye Oshin, M., Kumar, V., Vipin: Introduction to Data Mining (05 2005)
35. Tien, P.W., Wei, S., Liu, T., Calautit, J., Darkwa, J., Wood, C.: A deep learning approach towards the detection and recognition of opening of windows for effective management of building

ventilation heat losses and reducing space heating demand. Renewable Energy **177**, 603–625 (2021). https://doi.org/10.1016/j.renene.2021.05.155

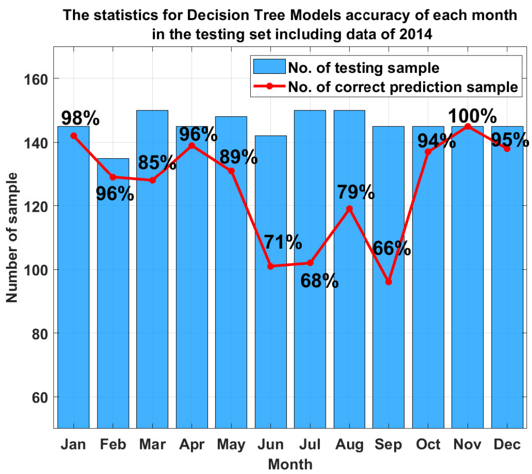
36. Viet, L., Sarlos, T., Smola, A.: Fastfood: Approximate kernel expansions in loglinear time. 30th International Conference on Machine Learning, ICML 2013 **28**, 244–252 (08 2013)
37. Wei, W., Ramalho, O., Malingre, L., Sivanantham, S., Little, J.C., Mandin, C.: Machine learning and statistical models for predicting indoor air quality. Indoor Air **29**(5), 704–726 (2019). https://doi.org/10.1111/ina.12580
38. Yao, M., Zhao, B.: Window opening behavior of occupants in residential buildings in beijing. Building and Environment **124**, 441–449 (2017)
39. Zhao, Y., Qiu, R.C., Zhao, X., Wang, B.: Speech enhancement method based on low-rank approximation in a reproducing kernel hilbert space. Applied Acoustics **112**, 79–83 (2016). https://doi.org/10.1016/j.apacoust.2016.05.008



(a)

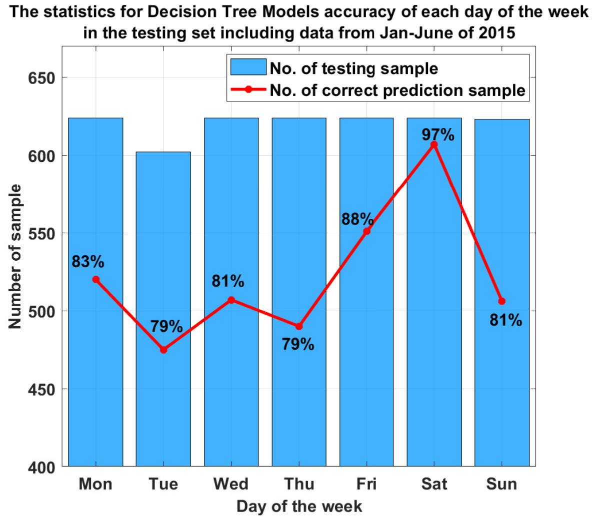


(b)

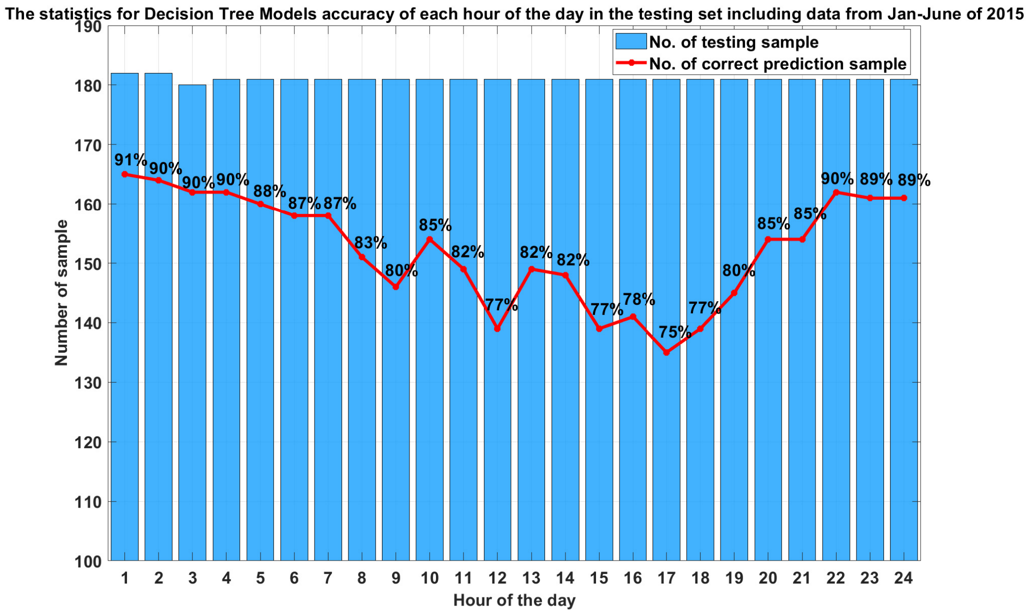


(c)

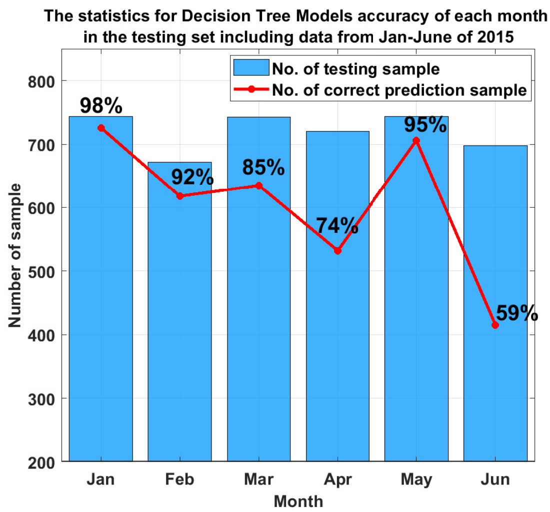
Fig. 11. The statistics for DT Models accuracy according to (a) each day of the week, (b) each hour of the day, and (c) each month for the test set 2014. The corresponding accuracy is displayed above the red curve for each time period.



(a)



(b)



(c)

Fig. 12. The statistics for DT Models accuracy according to (a) each day of the week, (b) each hour of the day, and (c) each month for the test set 2015. The corresponding accuracy is displayed above the red curve for each time period.

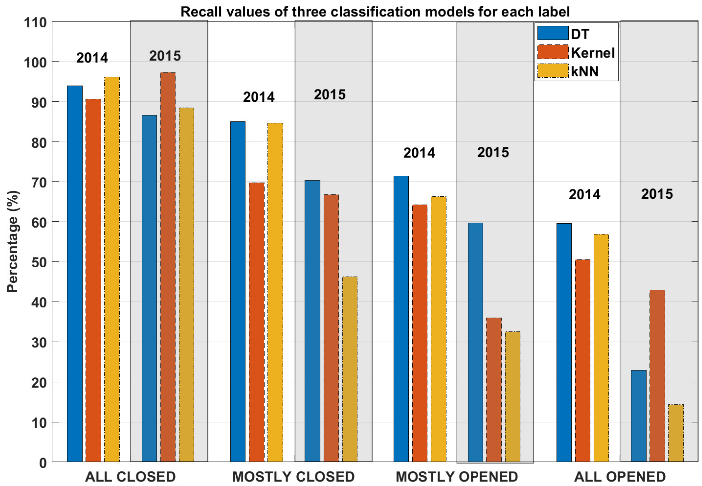


Fig. 13. Recall values of three classification models: Decision Tree, k-NN and Kernel approximation. The obtained values for testing data from January to June of 2015 are displayed in grey background.

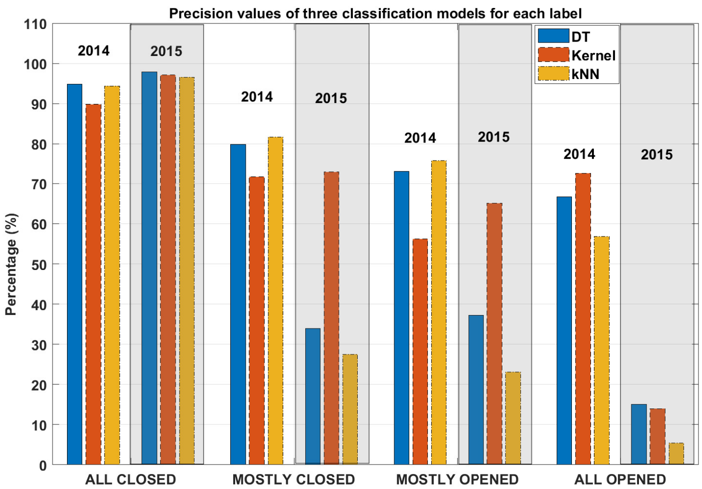


Fig. 14. Precision values of three classification models: Decision Tree, k-NN and Kernel approximation. The obtained values for testing data from January to June of 2015 are displayed in grey background.

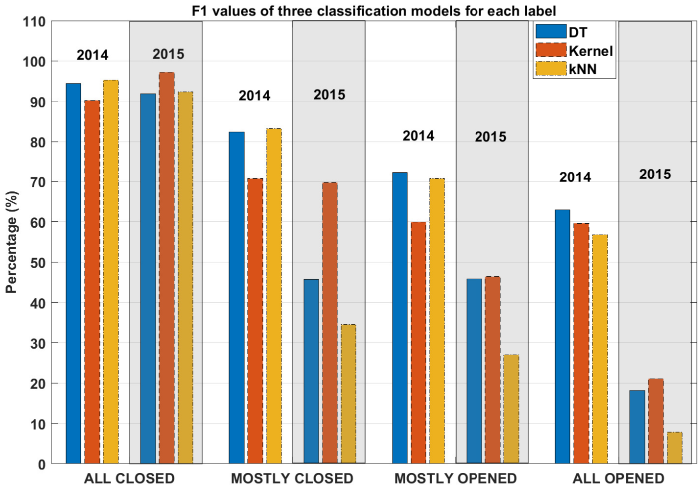


Fig. 15. F-1 values of three classification models: Decision Tree, k-NN and Kernel approximation. The obtained values for testing data from January to June of 2015 are displayed in grey background.