applied sciences

MDPI

Article

Heat Loss Coefficient Estimation Applied to Existing Buildings through Machine Learning Models

Miguel Martínez-Comesaña [1,*], Lara Febrero-Garrido [2], Enrique Granada-Álvarez [1], Javier Martínez-Torres [3] and Sandra Martínez-Mariño [1]

[1] Department of Mechanical Engineering, Heat Engines and Fluids Mechanics, Industrial Engineering School, University of Vigo, Maxwell s/n, 36310 Vigo, Spain; egranada@uvigo.es (E.G.-Á.); samartinez@uvigo.es (S.M.-M.)

[2] Defense University Center, Spanish Naval Academy, Plaza de España, s/n, 36920 Marín, Spain; lfebrero@cud.uvigo.es

[3] Department of Applied Mathematics I, Telecommunications Engineering School, University of Vigo, 36310 Vigo, Spain; javmartinez@uvigo.es

* Correspondence: migmartinez@uvigo.es

Received: 23 November 2020; Accepted: 12 December 2020; Published: 16 December 2020

Abstract: The Heat Loss Coefficient (HLC) characterizes the envelope efficiency of a building under in-use conditions, and it represents one of the main causes of the performance gap between the building design and its real operation. Accurate estimations of the HLC contribute to optimizing the energy consumption of a building. In this context, the application of black-box models in building energy analysis has been consolidated in recent years. The aim of this paper is to estimate the HLC of an existing building through the prediction of building thermal demands using a methodology based on Machine Learning (ML) models. Specifically, three different ML methods are applied to a public library in the northwest of Spain and compared; eXtreme Gradient Boosting (XGBoost), Support Vector Regression (SVR) and Multi-Layer Perceptron (MLP) neural network. Furthermore, the accuracy of the results is measured, on the one hand, using both CV(RMSE) and Normalized Mean Biased Error (NMBE), as advised by AHSRAE, for thermal demand predictions and, on the other, an absolute error for HLC estimations. The main novelty of this paper lies in the estimation of the HLC of a building considering thermal demand predictions reducing the requirement for monitoring. The results show that the most accurate model is capable of estimating the HLC of the building with an absolute error between 4 and 6%.

Keywords: energy efficiency; heat loss coefficient; machine learning; XGBoost; MLP; SVR

1. Introduction

Energy efficiency in buildings is a key roadmap to achieve sustainable development worldwide. Nearly one third of the primary energy consumed in the world comes from non-industrial buildings, and it is similar to all the energy used in the transport sector [1]. Consequently, the residential sector constitutes the largest contributor to global warming by means of carbon dioxide emissions. To achieve the goal of reducing building energy consumption, international cooperation is crucial. The European Union (EU) has created a strict legislative framework implementing two important directives: Energy Performance of Buildings Directive 2010/31/EU (EPBD) [2] and the Energy Efficiency Directive 2012/27/EU [3]. Both of them were updated and amended over the years. Specifically, the Directive amending the Energy Performance of Buildings Directive 2018/844/EU [4] includes different aspects strengthening the commitment of real building stock modernisation through technological improvements [5]. On the other hand, the International Energy Agency (IEA) is also working hard on this issue by means of its Energy in Buildings and Community (EBC) research programme developing activities towards Nearly Zero Energy Buildings (NZEB), the reduction of carbon dioxide emissions and energy savings technologies. One of the high priority research project themes is the building envelope.

The envelope of a building is one of the main sources of heat transfer between the exterior and the interior of the rooms, and it is responsible for the difference between the conditions predicted at the design stage and the actual conditions of the use of the building, including lighting, occupancy and Heating, Ventilation and Air Conditioning (HVAC) systems' operation. This is called the performance gap between building design and operation [6,7]. The envelope efficiency of a building can be characterized under in-use conditions by calculating the Heat Loss Coefficient (HLC) [8]. This coefficient determines the rate of heat flow through the buildings' envelope when a temperature difference exists between the indoor air and the outdoor air under steady state conditions [9]. Therefore, the accurate estimation of the HLC contributes to optimizing the energy consumption of a building. On the other hand, building energy performance simulation tools have been used in recent years to analyse the energy behaviour of buildings [10]. However, achieving an accurate simulation is not trivial. Many sources of error are introduced into the simulation: occupancy and user's behaviour, weather data [11], envelope materials and thickness, the use of electric equipment, etc. The alternative to simulation is the monitoring of the building operation, which leads to the difficulty of the installation of sensors in in-use buildings. Therefore, in this context, the application of advanced mathematical modelling techniques, such as machine learning methods, is becoming more and more common [12].

There are three widely used building energy prediction models: the white-box (physics-based), black-box (data-driven) and grey-box (a combination of physics based and data-driven) modelling approaches [13]. The application of black-box models in building energy analysis has been consolidated in recent years. These models have received considerable attention because of their great success in learning complex patterns and being able to make accurate predictions without specific knowledge of the subject [14,15]. Machine Learning (ML) models, also known as black-box models, are specific mathematical models that are capable of learning a pattern from data and extrapolating it to a new sample. Thus, they were widely applied in studies presenting control strategies to reduce energy costs. Three of the most used are: eXtreme Gradient Boosting (XGBoost), Support Vector Regression (SVR) and Multi-Layer Perceptron (MLP) neural network. Among them, XGBoost is characterized for building regression trees, one-by-one, so that the subsequent models are trained taking into account the residuals of the previous ones [16]. This algorithm has been used in numerous fields such as biology [17,18], econometrics [19], the environment [20,21] or, as in this case, the energy performance of buildings [22,23]. SVR models emerged as a predictive alternative due to the use of a distinctive loss function [24,25], on the one hand, and the dual formulation of the problem [26], on the other. This algorithm focuses on minimising an upper bound of the generalization error instead of minimising the prediction error in the training sample (*empirical risk minimizatio*n) [24]. The usefulness of this algorithm is demonstrated by its expansion into scientific fields such as econometrics [26], health [27,28], electrical efficiency in cities [29], as well as energy consumption in buildings [25,30]. Lastly, MLP neural networks have stood out in past years for their great capacity to model non-linear relationships between certain inputs and outputs, as well as for their massive interconnectivity [31–33]. Furthermore, this type of Artificial Neural Network (ANN) has been applied to different areas of study such as chemistry [34], the environment [35,36], sensors [37] or energy analysis of buildings [38–40].

The aim of this paper is to estimate the HLC of an existing building through the prediction of building thermal demands using a methodology based on ML models. Specifically, three different methods are applied to a public library in the northwest of Spain and compared: XGBoost, SVR and MLP neural network. The dataset consists of hourly observations of two climate variables (outdoor temperatures and solar radiation) and two variables related to the thermal behaviour of the building (heating demand and indoor temperatures). In addition, to improve model training, three temporal variables (hour of the year, day of the week and hour of the day) are taken into account as model inputs. A comparison between the errors obtained by the different machine learning models is carried out on three different validation samples. Moreover, the accuracy of each of the models is quantified with CV(RMSE) and Normalized Mean Biased Error (NMBE), both recommended by the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), and their usefulness was demonstrated in [41,42]. The novelty of this work lies in the application of machine learning models to estimate the HLC of a building through heating demand predictions. In this context, this paper contributes with a methodology that reduces the exigency of building monitoring, and heating demands will be accurately predicted. Additionally, this methodology does not depend on physical simulations, which require huge amounts of input data and specific knowledge and which have uncertainty in the evaluation of energy renovation or the energy certification of buildings.

2. Materials and Methods

2.1. Heat Loss Coefficient Calculation

The HLC is the most used Key Performance Indicator (KPI) to describe the building envelope energy efficiency. This coefficient reflects transmission heat losses through the envelope per degree difference between indoor and outdoor temperatures from walls, roofs and floors (UA(kW/K)) and ventilation and/or infiltration heat losses per degree difference between indoor and outdoor temperatures (Cv(kW/K))) [43]. Following the method developed by Uriarte et al. [8], the HLC can be described as Equation (1):

$$HLC = (UA + C_v) \qquad (\mathrm{kW/K}) \tag{1}$$

where UA stands for the building envelope transmission heat transfer coefficient (kW/K) and Cv stands for the infiltration and/or ventilation heat loss coefficient (kW/K). Moreover, the energy balance used to calculate the HLC, assuming stationary conditions, was developed by Uriate et al. [8,44], and it is presented in Equation (2):

$$\sum_{k=1}^{N} Q_k + \sum_{k=1}^{N} K_k = HLC \sum_{k=1}^{N} (T_{in,k} - T_{out,k}) - \sum_{k=1}^{N} (S_a V_{sol})_k \tag{2}$$

where Q_k represents the heating gains (kW), K_k represents the internal gains (kW) caused by occupants and electricity consumption ($K_k = K_{electricity,k} + K_{occupancy,k}$), $(T_{in,k} - T_{out,k})$ summarizes the gap between temperatures inside and outside the building in degrees Kelvin and $S_a V_{sol}$ stands for the solar gains (kW).

Several terms of this heat exchange are difficult to measure for in-use buildings. For example, the solar gains are complicated to know, and therefore, they introduce an important source of uncertainty into the equation. Thus, in this study, only cold and cloudy periods are considered where solar radiation was very low; direct solar radiation can be considered null; and all the radiation can be perceived as purely diffuse [45]. This is why in this study, direct solar radiation is not taken into account, and only diffuse solar radiation is considered. Another parameter that is difficult to estimate is the occupancy. Therefore, with the aim of reducing the sources of uncertainty, the study is developed during weekends when there is no occupancy. Then, the result of Equation (2) will give an idea of how well built the building is in terms of the enclosure of its envelope.

In this way, the HLC formula that is used in this analysis (average method), taking into account that the periods analysed in this paper were weekends with no occupation and in which there was only purely diffuse radiation, is shown in Equation (3):

$$HLC = \frac{\sum_{k=1}^{N} Q_k}{\sum_{k=1}^{N} (T_{in,k} - T_{out,k})} \tag{3}$$

2.2. Machine Learning Models

The different mathematical models (or black-box models) built to estimate the HLC of the analysed building are presented in this section. In this case, the three machine learning algorithms studied are XGBoost, SVR and MLP neural network.

2.2.1. Extreme Gradient Boosting

Gradient boosting is a meta-algorithm built on the basis of weak learners, like decision trees [19], with the aim of obtaining a strong ensemble learner [46,47]. Specifically, XGBoost is a scalable and efficient way to carry out the implementation of a gradient boosting algorithm. It is remarkable for its ease of implementation and its high accuracy in predictions [20]. This algorithm focuses on the idea of combining, in a final step, all the predictions made by a set of weak learners (*additive training strategy* [48]). Moreover, XGBoost is capable of simplifying the objective function with different combinations of predictive and regularization terms without loss in the optimal computational speed [19,20]. The specific learning process is summarized as follows [18,20]:

- An initial learner is fitted to the whole sample of inputs.
- A second model is fitted to the residuals of the first model to reduce its deficiencies.
- These two learning steps are repeated until a particular stop criterion is reached.
- The final predictions are obtained from the sum of the individual predictions of the learners used. The general function to obtain a prediction at step t, with additive training, is presented in Equation (4).

$$G_i^{(t)} = \sum_{j=1}^{t} G_j(\mathbf{x}_i) = G_i^{(t-1)} + G_t(\mathbf{x}_i) \tag{4}$$

where $G_t(\mathbf{x}_i)$ is the learner at step t, $G_i^{(t-1)}$ the prediction at step $t-1$ and $\mathbf{x}_i$ the input variable.

On the other hand, to prevent the problem of overfitting while maintaining an optimal computational speed, XGBoost takes into account Equation (5) to evaluate the suitability of the model [17,20]:

$$O^{(t)} = \sum_{j=1}^{n} L(\hat{\mathbf{y}}_i, \mathbf{y}_i) + \sum_{j=1}^{t} \Psi(G_i) \tag{5}$$

n being the number of observations, L a differentiable loss function and Ψ a regularization term that penalizes the complexity of the model [18].

XGBoost expands the loss function into the second order to be able to optimize the problem quickly. Furthermore, several specific techniques can reduce the possible overfitting of the algorithm [18]. One of them consists of, after each step of boosting, the weights recently added being scaled by a factor η (*shrinkage*). This process reduces the influence of the individual trees built, and therefore, the training will be slower and more efficient.

Lastly, XGBoost optimization requires the control of multiple parameters [19], and finding the best combination of them is important. In this analysis, the optimal combinations of parameters (*max_depth,* *min_child_weight*, *subsample*, *colsample_bytree* and *learning_rate*) are found through a k-fold cross-validation method ($k = 10$) [49] carried out on the different validation samples studied.

2.2.2. Support Vector Regression

SVR is a nonlinear kernel based regression model that focuses on finding the best regression hyperplane with the least structural risk in a high-dimensional feature space [25–27]. The SVR function is represented in Equation (6):

$$g(\mathbf{x}) = \mathbf{w}^T \delta(\mathbf{x}) + \mathbf{b} \tag{6}$$

$\delta(x)$ being a nonlinear mapping that connects the input space to the feature space, b a bias term and w the weight coefficient vector. In this case, both w and b are estimated by resolving the following optimization problem [26,28]:

$$\begin{aligned} \underset{\mathbf{w},\mathbf{b}}{\text{minimize}} \quad & \frac{1}{2} \parallel \mathbf{w} \parallel^2 + C \sum_{j=1}^{l} (\tau_j + \tau_j^*) \\ \text{subject to} \quad & \mathbf{y}_j - g(\mathbf{x}_j) \leq \epsilon + \tau_j \\ & g(\mathbf{x}_j) - \mathbf{y}_j \leq \epsilon + \tau_j^* \\ & \tau_j, \tau_j^* \geq 0 \end{aligned} \tag{7}$$

where the constant $C > 0$ represents the trade-off between training error and model complexity, ϵ corresponds to a threshold value and l is the number of training patterns. As Huang et al. [28] and Vrablecová et al. [29] showed, once having resolved the optimization problem (Equation (7)) and taking the Lagrangian, the model solution can be reached with its dual representation (see Equation (8)):

$$g(\mathbf{x}) = \sum_{j=1}^{l} (\alpha_j^* - \alpha_j) K(\mathbf{x}, \mathbf{x}_j) + \mathbf{b} \tag{8}$$

where α_j, α_j^* are the Lagrangian multipliers ($\neq$0) and the solution for the dual problem, b the bias term and $K(x_j, x)$ the kernel function based on the inner product $\langle \delta(x_j), \delta(x) \rangle$.

Specifically, the model optimization for each of the validation samples studied is carried out through the tuning of certain parameters. In this case, the selected parameters were penalty C and term ϵ [24,29]. With a cross-validation process (k-fold = 10) [49], different values of these parameters were evaluated, and the best adjustment was obtained depending on the validation sample considered.

2.2.3. Multi-Layer Perceptron Neural Network

MLP is an ANN characterized by having several layers [32,40,50]:

- An input layer (first layer), where the inputs are introduced.
- An output layer (last layer), where the results obtained by the trained model are given.
- Hidden layers (intermediate layers) positioned between the previous ones. They can be zero, one or more.

The specific neural network built in this study is composed of five layers (one input layer, one output layer and three hidden layers). The internal structure of neurons in each layer of the network is as follows: 100-100-100-50-1. This architecture is selected after a k-fold cross-validation (k = 10) study [51] in which the grid of different hidden layer options was between zero and four [52,53]. On the other hand, in relation to the number of neurons in the network layers, the different options, as recommended by Vujicic et al. [52] and Doukim et al. [54], take into account the size of the sample and the number of inputs and outputs.

Moreover, due to the complexity of the problem, this grid of values was completed with more complex options such as: 50, 100, 200 and 400.

The neural network training was developed via backward propagation; errors were propagated and corrected through the network [55]. In this way, the real outputs must be known. In addition, the training process was based on updating the weights with an average update (batch-learning), which was obtained by introducing all the patterns in the input file (an epoch) and accumulating the weight updates [56,57]. The stop criterion used to avoid overfitting problems was the cross-validation due to its effectiveness in stopping with the best model generalization [58,59]. The training will stop when the performance of the neural network, measured by the Mean Squared Error (MSE) and evaluated with a small part of the whole sample (test sample), stagnates or starts to decrease. Each MLP trained in this study used the Rectified Linear Unit (ReLU) as the activation function [60], a normal kernel initializer and the Adaptive Moment Estimation (Adam) optimizer [61]. Lastly, the batch size with which the neural networks were trained was equal to 64, and the limit to the epochs in which the model fit did not improve was 100. Further information about the options and variations of the MLP training process can be found in [62,63].

2.3. Case Study Data Acquisition

2.3.1. Description of the Building

The methodology proposed was tested on a public library of the Faculty of Marine Sciences at the University of Vigo (see Figure 1) that is located in the northwest of Spain. The building has three floors that are interconnected, and it is completely monitored and has been used for research purposes many times. Therefore, the building, its HVAC system and its data acquisition system were further and more deeply described in other articles such as Cacabelos et al. [64,65], Fernández et al. [66] and Martínez et al. [40]. The building envelope has a large, south-facing window (see Figure 1a), and its enclosures are made of different concrete and insulation material. Table 1 shows the properties and composition of each material layer of the walls, floor and roof.

(**a**) (**b**)

Figure 1. Appearance of the analysed building: (**a**) exterior and (**b**) interior.

Table 1. Composition and properties of the enclosures of the analysed building.

EXTERIOR WALLS						
Layer (Indoor-Outdoor)	Material	Thickness (cm)	Properties			
			λ (W/m · K)	c (kJ/kg · K)	ρ (kg/m^3)	R (h · m^2 · K/kJ)
1	Plasterboard	2	0.11	1	900	-
2	Extruded polystyrene	4	0.03	1	31	-
3	Mineral wool	6	0.04	1	-	-
4	Air	4	-	-	-	0.05
5	Concrete	25	1.15	1	1800	-
INTERIOR FLOOR						
Layer (Indoor-Outdoor)	Material	Thickness (cm)	Properties			
			λ (W/m · K)	c (kJ/kg · K)	ρ (kg/m^3)	R (h · m^2 · K/kJ)
1	Extruded polystyrene	4	0.03	1	31	-
2	Greenket	2	0.10	1.6	300	-
3	Common concrete	8	1.3	1	2000	-
4	Lightweight concrete	6	0.34	1.1	600	-
5	Concrete block	2	1.32	1	1330	-
SLAB						
Layer (Indoor-Outdoor)	Material	Thickness (cm)	Properties			
			λ (W/m · K)	c (kJ/kg · K)	ρ (kg/m^3)	R (h · m^2 · K/kJ)
1	Extruded polystyrene	4	0.03	1	31	-
2	Greenket	2	0.10	1.6	300	-
3	Common concrete	14	1.3	1	2000	-
4	Lightweight concrete	6	0.34	1.1	600	-
5	Concrete block	2	1.32	1	1330	-
ROOF						
Layer (Indoor-Outdoor)	Material	Thickness (cm)	Properties			
			λ (W/m · K)	c (kJ/kg · K)	ρ (kg/m^3)	R (h · m^2 · K/kJ)
1	Plasterboard	2	0.11	1	900	-
2	Extruded polystyrene	3	0.03	1	31	-
3	Com on concrete	10	1.3	1	2000	-
4	Reinforced concrete	36	2.30	1	2400	-
5	Air	5	-	-	-	0.044

2.3.2. Pre-Processing Data

This analysis is focused on the estimation of the HLC of a building. The data available in this study to train the machine learning models were hourly observations between March 2016 and December 2017 of four variables. The variables considered were two describing the thermal behaviour of the building (thermal demand and indoor temperatures) and two describing the climate conditions (outdoor temperature and solar radiation). In addition, to find significant hours when the heating worked normally, only the hours with a thermal demand of more than 5 kW were selected (n = 5727). In this case, it was not necessary to have a continuous sample because the training did not take into account time lags in the explanatory variables. Therefore, three time variables (hour of the year, day of the week, and hour of the day) were introduced into the models to provide more information and thus to improve the training. The variables that were used as model inputs are presented in Figure 2.

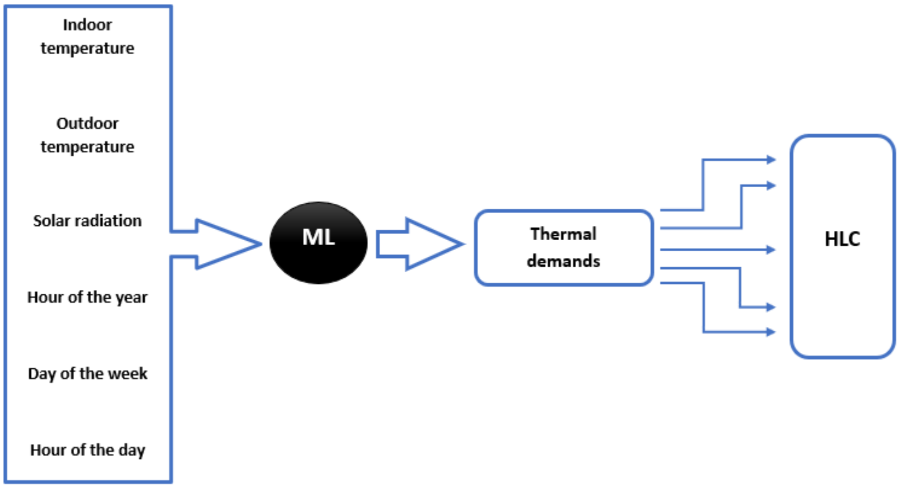


Figure 2. Summary of the training and prediction process. On the left, the input variables of the models. On the right, the objective estimations of the Heat Loss Coefficient (HLC).

The model fit was carried out through a cross-validation process in which the whole sample was divided consecutively into two individual samples: training and testing. In the search for the optimal model and the selection of hyperparameters, the models were tested with different partitions (or subsamples) of the whole training sample (k-fold = 10) [49]. In addition, the models, after being trained and tested with 10 different test samples, were validated with three independent samples to prove their efficiency in estimating the HLC of the building:

- Sample 1: 15/12/2018 07:00 – 17/12/2018 07:00
- Sample 2: 19/01/2019 07:00 – 21/01/2019 07:00
- Sample 3: 09/03/2019 07:00 – 11/03/2019 07:00

These three samples had to comply with certain restrictions due to the specification of the HLC formula (see Section 2.1) and because of the singularities of the monitoring of the building studied (see Section 2.3.1). The restrictions for considering a period suitable for the calculation of the HLC are [8,44]:

- **Weekend**: Due to the non-availability of occupation data, the period must be on the weekend when there was no occupancy in the building.
- **Cool period**: The average difference between indoor and outdoor temperature must be 10 K or more.
- **Cloudy period**: Solar radiation must be low: gains from solar radiation equal to or below 10% of the thermal demand.
- **Temperature stability**: The average between indoor and outdoor temperature must be similar at the beginning and end of the period to ensure steady state conditions.

2.3.3. Validation and Error Assessment

The Coefficient of Variation of the Root Mean Squared Error (CV(RMSE)) and the Normalized Mean Biased Error (NMBE) were the error measures calculated to evaluate the accuracy of the models presented:

$$\mathrm{CV(RMSE)} = 100 \times \frac{\sqrt{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2/N}}{\bar{y}} \tag{9}$$

$$\mathrm{NMBE} = 100 \times \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)}{\sum_{i=1}^{N}(y_i)} \tag{10}$$

Both measurements were used to compare the performance of the different models in the three validation samples. In addition, the CV(RMSE) was also taken into account to select the best trials of each model (with the lowest error) and, then, represented graphically in the Results Section. These measures have been used and their efficiency has been proven in similar studies [42,67,68].

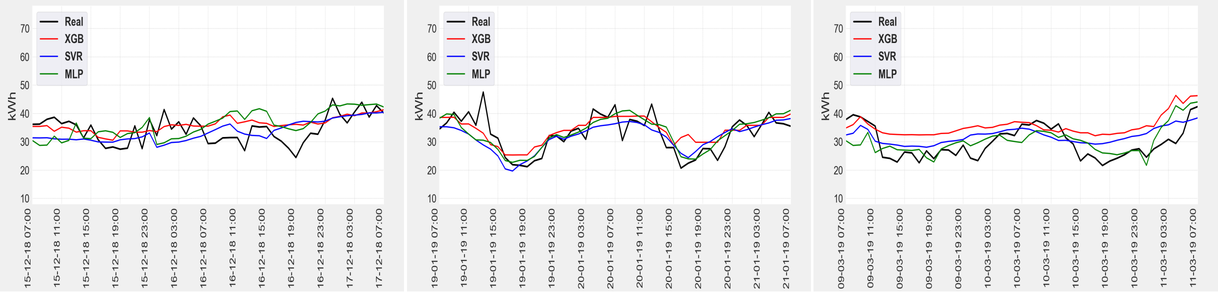
3. Results and Discussion

A methodology, based on thermal demands' predictions made with black-box models, to estimate the HLC of a building is developed in this paper. In particular, the building under study was the public library of the Faculty of Marine Sciences at the University of Vigo. The available data were hourly observations of the variables presented in Figure 2 from March 2016 to December 2017; only taking into account the hours with a significative thermal demand ($\geq$ 5 kW). In this analysis, three specific weekends were considered to study the performance of the models estimating the HLC through heating demand predictions. A comparison between the accuracy of each ML model analysed is presented in the following sections.

Section 3.1 presents the models performance in heating demand predictions and Section 3.2 a similar analysis for the HLC estimations. While in the thermal demand analysis, CV(RMSE) (Equation (9)) and NMBE (Equation (10)) were calculated for each of the models, in the HLC section, the errors were calculated based on an absolute variation rate. Furthermore, to represent the average performance of the different models, the predictions of the three validation samples were repeated 10 times (varying the subsamples where the model was tested). Thus, the numerical results shown are average errors obtained through the 10 trials, as well as their standard deviations. Finally, all figures shown in the following sections were created with the Python programming language [69].

3.1. Thermal Demand Analysis

The results of the heating demand predictions for the analysed building in the three validation samples are presented in this section in Figure 3 and Table 2. These predictions are important as they will be used to obtain the subsequent HLC estimations (see Section 3.2). Furthermore, in Figure 3, each of the algorithms is represented by the prediction that obtains the lowest CV(RMSE) among the 10 trials (best scenario).



(**a**) (**b**) (**c**)

Figure 3. Results of thermal demand predictions in all samples studied: (**a**) Sample 1; (**b**) Sample 2; (**c**) Sample 3. Each machine learning model is represented by the prediction curve with the lowest CV(RMSE) among the 10 experiment trials.

In the case of Sample 1 (Figure 3a) and taking into account the CV(RMSE) results, the XGBoost model presents the lowest average error (18.57%). The MLP neural network and SVR model show a worse performance with an average error of 19.84% and 21.31%, respectively (see Table 2). Moreover, Table 2 shows that in all models, the variability of errors in the 10 trials was relatively low (below ± 4), but MLP is the one that shows the lowest dispersion (±2.41). On the other hand, regarding the NMBE results, the lowest average error was obtained by the MLP model (−5.49%), while the other algorithms only managed to obtain negative average errors higher than 10% (−12.52% for XGBoost and −11.63% for SVR). In addition, in this case, while the variability of the NMBE results of XGBoost and MLP are similar and around ± 5 (see Table 2), the SVR results show a greater dispersion (above ± 8). In general, all the models, in the best scenario, efficiently reproduced the real behaviour of the thermal demand of the building analysed (see Figure 3a). Although the XGBoost model obtained better results in relation to the CV(RMSE), the MLP predictions showed a better overall adjustment to reality by obtaining a similar average CV(RMSE) and an average NMBE much lower than XGBoost.

In Sample 2 (Figure 3b), considering the CV(RMSE) results, the XGBoost algorithm and MLP neural network show a similar average performance. MLP obtained an average error of 17.38% and XGBoost 17.43%. In this situation, the SVR model shows better results than the other models with an average error of 16.61% (see Table 2). Additionally, while SVR and MLP present a similar error dispersion around ± 2, the XGBoost model is the one that presents the highest variability (above ± 4). With respect to the NMBE results, the SVR model again obtained the lowest average error (2.99%). The MLP neural network and XGBoost algorithm, on the other hand, present higher and negative average error (−4.01% and −8.40%, respectively). However, as shown in Table 2, the SVR model is the one that showed the greatest dispersion in the NMBE results (almost ± 7), even though the other models also showed an important variability. In Figure 3b, it is demonstrated that each of the algorithms, taking into account the best scenario, is capable of replicating the reality except for certain peaks. However, more specifically, the model that yielded the best average performance was SVR.

Regarding Sample 3 (see Figure 3c), the three models studied obtained average CV(RMSE) values above 20%. While the SVR algorithm and MLP neural network showed an average CV(RMSE) of 21.60% and 21.54%, respectively, XGBoost model obtained an average error around 29% (see Table 2). Furthermore, taking into account that the variabilities presented were not high, the MLP neural network obtained the least variable results (±2.33) and XGBoost the most variable (±3.81). In the case of the NMBE results, the MLP model stands out from the rest (see Table 2). It shows the lowest average of NMBE by far (0.25%). The average results of the other algorithms were higher than 10% and negative (−13.60% for SVR and −23.05% for XGBoost). On the other hand, the dispersion of the NMBE values among the 10 trials of this sample is high (above 4.5) in all the models, and the MLP neural network was the model with the greatest error dispersion (±6.19). In addition, as in the other validation samples, Figure 3c shows that the built models, in the best case scenario, are very close to the real values.

Table 2. Numerical results of the thermal demand predictions for all validation samples. The mean of the CV(RMSE) and the Normalized Mean Biased Error (NMBE) obtained through the 10 trials, besides the standard deviation (SD) of each of them, are presented.

Model	Sample 1				Sample 2				Sample 3			
	CV(RMSE) (%)	SD	NMBE (%)	SD	CV(RMSE) (%)	SD	NMBE (%)	SD	CV(RMSE) (%)	SD	NMBE (%)	SD
XGBoost	18.570	3.940	−12.517	5.224	17.434	4.203	−8.404	4.759	29.143	3.808	−23.052	4.473
SVR	21.312	3.710	−11.627	8.460	16.607	2.417	2.987	6.819	21.596	3.581	−13.600	5.049
MLP	19.839	2.409	−5.491	5.693	17.381	2.301	−4.012	3.061	21.543	2.331	0.253	6.189

Lastly, it is demonstrated that all the models presented, in general, are capable of recreating reality (see Figure 3), obtaining optimal errors in all the samples studied (see Table 2). Concretely, they normally predict above the real values of the heating demand of the building analysed (see the negative NMBE results in Table 2). Furthermore, encompassing all the results, the model that performed best was the MLP neural network. In terms of CV(RMSE), it always yielded one of the best results, and in terms of NBME, it obtained much better results than the other algorithms in two of the three validation samples.

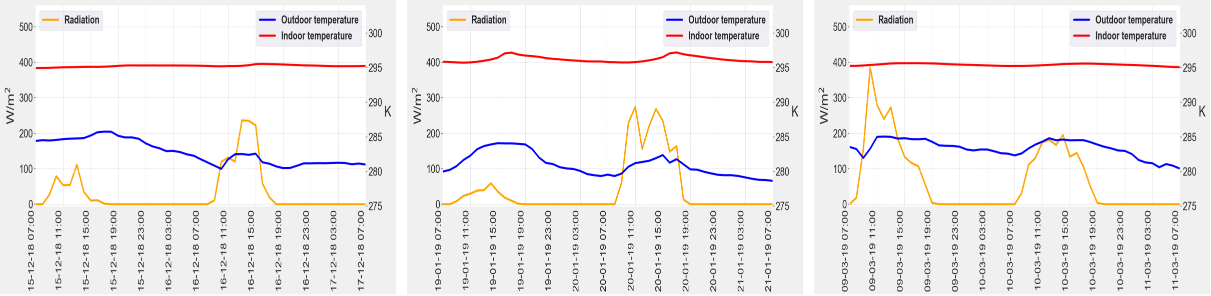
3.2. HLC Estimation Analysis

The specific characteristics of each of the validation samples studied in this work are analysed in Table 3 and presented in Figure 4. Moreover, the results of the HLC estimations, based on the previous heating demand predictions, for each of the three validation samples are shown in this section in Table 4. Although estimated from the heating demand predictions during a weekend, the HLC is represented as a single number (see Equation (3)). As in the preceding section, the average performance of the different algorithms among the 10 trials is summarized in Table 4.

Table 3. Thermal conditions of the three validation samples. These represent the mean difference between indoor and outdoor temperatures, the average heating demand ($\overline{Q}$), the solar radiation gain (Rad) over the thermal demand and the average temperatures (indoor and outdoor) at the start ($\overline{T^{\circ}}_{initial}$) and at the end of the period ($\overline{T^{\circ}}_{final}$). HLC, Heat Loss Coefficient.

	$\mathbf{HLC_{calculated}}$ (kW/K)	$\overline{T_{int} - T_{out}}$ (K)	$\overline{Q}$ (kW)	Rad/Q (%)	$\overline{T^{\circ}}_{initial}$ (K)	$\overline{T^{\circ}}_{final}$ (K)
Sample 1	2.750	12.476	34.313	0.091	289.671	288.127
Sample 2	2.146	15.246	32.725	0.127	287.917	287.220
Sample 3	2.464	12.053	29.704	0.232	289.402	287.741

First, Table 3 shows that the three validation samples fulfilled the conditions necessary to efficiently measure the HLC of the analysed building (see Section 2.3.2). Moreover, each of the samples has different HLC values. The main different thermal conditions of the samples, which caused slightly different HLC values, are also summarized in Table 3 and presented in Figure 4. The highest HLC value was obtained in Sample 1 (2.75 kW/K) due to the fact that the weight of the radiation gains, in relation to the heating demands, was the lowest among the samples (0.09%). This is related to the fact that the average thermal demand throughout this period was the highest and is a numerator value in the HLC formula (see Equation (3)). In addition, the average difference between indoor and outdoor temperatures, as shown in Figure 4a, was one of the smallest (12.48 K). These were the main reasons for obtaining a greater HLC (see Table 3). On the other hand, Sample 2 is where the HLC value is the lowest (2.15 kW/K) because, as shown in Figure 4b, the average difference between indoor and outdoor temperatures was the largest (15.25 K). This value, which is in the denominator of the HLC formula, together with an average heating demand significantly lower than in Sample 1 (32.72 kW) reduces the calculated HLC value (see Table 3). Lastly, in Sample 3 (see Figure 4c), an intermediate value of the HLC (2.46 kW/K) that came from the lowest average difference between indoor and outdoor temperatures (12.05 K) and the lowest average thermal demand (29.70 kW) was obtained.



(**a**) (**b**) (**c**)

Figure 4. Summary of the thermal conditions of each of the validation samples: (**a**) Sample 1; (**b**) Sample 2; (**c**) Sample 3. Solar radiation and indoor and outdoor temperatures affecting the analysed building during the specific time periods are represented.

On the other hand, in the case of Sample 1, where the calculated HLC was 2.75 kW/K, the MLP neural network was the most accurate model (see Table 4). While this algorithm obtained an average absolute variation rate of 6.50%, the XGBoost and SVR models were only able to obtain an average absolute variation greater than 10% (12.52.% and 12.16%, respectively). Thus, the average HLC value estimated by MLP (2.90 kW/K) is the closest to the measured HLC value. Table 4 shows that all models obtained a higher average estimation than the calculated one (the same situation as in Section 3.1). Additionally, in relation to the variation rate dispersion, the MLP neural network was the one with the lowest variability among the errors (±4.5). However, in general, all models showed a significant high standard deviation.

Table 4. Numerical results of the HLC estimations for all validation samples. The values shown summarize the performance of each of the models through 10 repetitions of the experiment. The mean of the absolute variation rates (together with the standard deviation), the mean of the estimated HLC values and the calculated HLC are presented.

Model	**Sample 1—HLC$_{\text{calculated}}$ = 2.750**			**Sample 2—HLC$_{\text{calculated}}$ = 2.146**			**Sample 3—HLC$_{\text{calculated}}$ = 2.464**		
	HLC$_{\text{estimated}}$	**% Variation**	**SD**	**HLC$_{\text{estimated}}$**	**% Variation**	**SD**	**HLC$_{\text{estimated}}$**	**% Variation**	**SD**
XGBoost	3.094	12.517	5.224	2.327	8.404	4.759	3.032	23.052	4.473
SVR	3.070	12.158	7.677	2.082	5.680	4.813	2.800	13.600	5.049
MLP	2.901	6.505	4.500	2.232	4.083	2.966	2.458	4.966	3.702

In Sample 2 the calculated HLC was 2.15 kW/K, and in relation to the average absolute variation values, the most accurate model was the MLP neural network (4.08%), but close to the SVR model (5.68%). As in the thermal demand section, the XGBoost model performed worse with an average absolute variation rate of 8.40% (see Table 4). Therefore, the HLC values estimated by the SVR and MLP models are very close to the calculated HLC value; while the average value obtained by SVR is 2.08 kW/K, the average value estimated for MLP is 2.23 kW/K. With respect to the dispersion of variation rate data and, as in Sample 1, Table 4 shows that the MLP neural network was the most stable model with a standard deviation below ±3. The other models presented values above ±4.

Regarding Sample 3, in which the measured HLC was 2.46 kW/K, the MLP neural network was again the model with the best average performance. While the XGBoost and SVR algorithms presented average absolute variation of 23.05% and 13.60%, respectively, the MLP model showed an average absolute variation of 4.97% (see Table 4). Therefore, the average HLC value from the MLP estimations (2.46 kW/K) was much closer to the calculated HLC value than those obtained by the other models. In addition, in this sample, all models obtained a high variability in their results: all standard deviations were higher than ±3.5. In this situation, the MLP neural network was the model with the lowest dispersion among its errors (±3.70).

Definitely, the HLC value that characterizes the studied building, calculated as the average of the presented results, was 2.45 ± 0.30 kW/K. Observing the results presented in Table 4 and taking into account the whole analysis (the results of the three validation samples), the model that presents the best average performance was the MLP neural network. It was the model that obtained the most stable results and had the lowest average absolute variation rate for all the samples analysed. In this way, it is demonstrated that it is possible to estimate the HLC of the analysed building with an absolute error around 4 or 6% if an MLP model is used to make the necessary thermal demand predictions. On the other hand, if an SVR algorithm is used, the error increases to 5-13%, and if the model is XGBoost, the errors vary between 8 and 23%.

4. Conclusions

A new methodology for estimating the HLC of a building is presented in this paper. It is based on the introduction of thermal demand predictions obtained with machine learning models in the HLC formula. This study focuses on the analysis, on the one hand, of monitored data on heating demands and indoor temperatures belonging to the Science Library of the University of Vigo. On the other hand, two meteorological variables (outdoor temperature and solar radiation) and three temporal variables (hour of the year, day of the week and hour of the day) are also taken into account. The aim of this paper is to show a methodology that allows the efficient estimation of the HLC value of a building without the need to control its heating demands (nor indoor temperatures if they are assumed to be constant). The search for the optimal methodology considers and compares three different machine learning models (XGBoost, SVR and MLP neural network). In addition, the performance of each one is evaluated and analysed both through its average accuracy in thermal demand predictions and its average accuracy in HLC estimations.

The research contribution of this work is the application of mathematical models to estimate the HLC of a building with low errors. In addition to reducing the necessity for monitoring, these models can be useful for detecting errors in measurements from sensors installed in the building. Moreover, the black-box models presented contributes with advantages compared to traditional research in building simulation. The use and application of the traditional building thermal simulation models are conditioned by the need for significant knowledge on a subject. In addition, these models need to control many different parameters related to the energy performance of a building. Nevertheless, machine learning models, which need less time for development than dynamic simulation methods, do not require specific prior knowledge and can be applied in numerous fields. The only important need for these models is the availability of a significant amount of data from which a behaviour pattern is extracted. Furthermore, the inputs introduced in the built models are variables typically monitored in buildings. Therefore, the methodology presented here is extractable to other studies and buildings.

The results obtained show that it is possible to efficiently estimate the HLC value of a building over specific time periods with black-box models. To this end, it is important to obtain thermal demand predictions, which are used as inputs in HLC estimations, with average errors lower than the values proposed for the calibrated models. The results also demonstrate that the MLP neural network is the algorithm with the best average performance in HLC estimation (more stability and higher average accuracy in all samples studied). The SVR model shows a close average behaviour, but XGBoost, except in Sample 1, presents a much worse performance than the other two. In the first validation sample, in which the measured HLC is 2.750 kW/K, only the MPL neural network is capable of obtaining an average absolute variation rate below 10% (6.50%). The SVR and XGBoost models obtain an average variation of 12.16% and 12.52%, respectively. On the other hand, in the second validation sample, where the calculated HLC was 2.146 kW/K, both the MLP and SVR models performed better than XGBoost. While the first two obtain an average absolute variation rate of 4.08% and 5.68%, respectively, XGBoost shows an average variation of 8.40%. Lastly, in the case of the third validation sample, with a measured HLC value of 2.464 kW/K, the MLP neural network yields an average absolute variation rate far from the other algorithms (4.97%). The SVR and XGBoost models are only able to yield an average absolute variation of 13.60% and 23.05%, respectively. Furthermore, regarding the dispersion in error data, MLP model is the one that shows the most stable results in all validation samples. The other algorithms present similar variability between them, but higher than the one obtained by the MLP model. Definitely, taking into account all the results presented, the HLC value of the analysed building is 2.45 ± 0.30 kW/K.

From an energy point of view, the conclusion is that efficient predictions related to the thermal conditions of a building and, in addition, made by machine learning models can be used to efficiently estimate its HLC. This is demonstrated in this paper with three different validation samples that fulfil the necessary conditions to calculate the HLC. The most accurate model, which in this case is the MLP neural network, is able to estimate the HLC of the analysed building with an average absolute variation rate of around 5% and with a standard deviation of around ± 3. On the other hand, the main limitation of this research is the many restrictions on finding suitable time periods for calculating the HLC. In this case, the unavailability of occupation data means that only weekends are studied. For this reason, some possible future lines of research are similar analyses considering more data such as occupation or extending the study to more general situations.

Author Contributions: Conceptualization, L.F.-G. and M.M.-C.; methodology, M.M.-C.; software, M.M.-C.; validation, L.F.-G., E.G.-Á. and S.M.-M.; formal analysis, L.F.-G., J.M.-T. and S.M.-M.; investigation, M.M.-C., L.F.-G. and S.M.-M.; resources, E.G.-Á.; data curation, M.M.-C. and J.M.-T.; writing, original draft preparation, M.M.-C. and L.F.-G.; writing, review and editing, L.F.-G., J.M.-T., E.G.-Á. and S.M.-M.; visualization, M.M.-C.; supervision, E.G.-Á., L.F.-G. and J.M.-T.; project administration, E.G.-Á.; funding acquisition, E.G.-Á. All authors read and agreed to the published version of the manuscript.

Funding: This research was funded by the Spanish Government (Science, Innovation and Universities Ministry) under the project RTI2018-096296-B-C21.

Acknowledgments: This research was supported by the Spanish Government (Science, Innovation and Universities Ministry) under the project RTI2018-096296-B-C21.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. *IEA EBC Annex 75: 5th Expert Meeting*; Energy in Buildings and Communities Programme (EBC): Venice, Italy, 2020.
2. *Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the Energy Performance of Buildings*; Official Journal of the European Union: Brussels, Belgium, 2010; Volume 153, pp. 13–35.
3. *Directive 2012/27/EU of the European Parliament and of the Council of 25 October 2012 on Energy Efficiency, Amending Directives 2009/125/EC and 2010/30/EU and Repealing Directives 2004/8/EC and 2006/32/EC Text with EEA Relevance*; European Commission: Brussels, Belgium, 2012; Volume 315, pp. 1–56.

4. *Directive 2018/844/EU of the European Parliament and of the Council of 30 May 2018 Amending Directive 2010/31/EU on the Energy Performance of Buildings and Directive 2012/27/EU on Energy Efficiency*; European Commission: Luxembourg, 2018; Volume 156, pp. 75–91.
5. Gatt, D.; Yousif, C.; Cellura, M.; Camilleri, L.; Guarino, F. Assessment of building energy modelling studies to meet the requirements of the new Energy Performance of Buildings Directive. *Renew. Sustain. Energy Rev.* **2020**, *127*, 109886. [CrossRef]
6. Yan, D.; Hong, T.; Dong, B.; Mahdavi, A.; D'Oca, S.; Gaetani, I.; Feng, X. IEA EBC Annex 66: Definition and simulation of occupant behavior in buildings. *Energy Build.* **2017**, *156*, 258–270. [CrossRef]
7. De Wilde, P. The gap between predicted and measured energy performance of buildings: A framework for investigation. *Autom. Constr.* **2014**, *41*, 40–49. [CrossRef]
8. Uriarte, I.; Erkoreka, A.; Giraldo-Soto, C.; Martin, K.; Uriarte, A.; Eguia, P. Mathematical development of an average method for estimating the reduction of the Heat Loss Coefficient of an energetically retrofitted occupied office building. *Energy Build.* **2019**, *192*, 101–122. [CrossRef]
9. Givoni, B. Well Tempered and Illuminated Interiors. In *Passive and Low Energy Ecotechniques*; Bowen, A., Ed.; Pergamon: Oxford, UK, 1985; pp. 210–225. [CrossRef]
10. Maile, T.; Fischer, M.; Bazjanac, V. Building Energy Performance Simulation Tools-a Life-Cycle and Interoperable Perspective. In *Center for Integrated Facility Engineering (CIFE) Working Paper*; CIFE: Stanford, CA, USA, 2007; Volume 107.
11. Eguía Oller, P.; Alonso Rodríguez, J.; Saavedra González, A.; Arce Fariña, E.; Granada Álvarez, E. Improving transient thermal simulations of single dwellings using interpolated weather data. *Energy Build.* **2017**, *135*, 212–224. [CrossRef]
12. Lü, X.; Lu, T.; Kibert, C.J.; Viljanen, M. Modeling and forecasting energy consumption for heterogeneous buildings using a physical–statistical approach. *Appl. Energy* **2015**, *144*, 261–275. [CrossRef]
13. Li, X.; Wen, J. Review of building energy modelling for control and operation. *Renew. Sustain. Energy Rev.* **2014**, *37*, 517–537. [CrossRef]
14. Helm, M.; Swiergosz, A.; Haeberle, H.; Karnuta, J.; Schaffer, J.; Krebs, V.; Spitzer, A.; Ramkumar, P. Machine Learning and Artificial Intelligence: Definitions, Applications, and Future Directions. *Curr. Rev. Musculoskelet. Med.* **2020**, *13*. [CrossRef] [PubMed]
15. Murdoch, W.J.; Singh, C.; Kumbier, K.; Abbasi-Asl, R.; Yu, B. Interpretable machine learning: definitions, methods, and applications. *arXiv* **2019**, arxiv:abs/1901.04592.
16. Pesantez-Narvaez, J.; Guillen, M.; Alcañiz, M. Predicting Motor Insurance Claims Using Telematics Data—XGBoost versus Logistic Regression. *Risks* **2019**, *7*, 10. [CrossRef]
17. Babajide Mustapha, I.; Saeed, F. Bioactive Molecule Prediction Using Extreme Gradient Boosting. *Molecules* **2016**, *21*, 983. [CrossRef] [PubMed]
18. Wang, H.; Liu, C.; Deng, L. Enhanced Prediction of Hot Spots at Protein-Protein Interfaces Using Extreme Gradient Boosting. *Sci. Rep.* **2018**, *8*. [CrossRef] [PubMed]
19. Carmona, P.; Climent, F.; Momparler, A. Predicting failure in the U.S. banking sector: An extreme gradient boosting approach. *Int. Rev. Econ. Financ.* **2019**, *61*, 304–323. [CrossRef]
20. Fan, J.; Wang, X.; Wu, L.; Zhou, H.; Zhang, F.; Yu, X.; Lu, X.; Xiang, Y. Comparison of Support Vector Machine and Extreme Gradient Boosting for predicting daily global solar radiation using temperature and precipitation in humid subtropical climates: A case study in China. *Energy Convers. Manag.* **2018**, *164*, 102–111. [CrossRef]
21. Guo, R.; Zhao, Z.; Wang, T.; Liu, G.; Zhao, J.; Gao, D. Degradation State Recognition of Piston Pump Based on ICEEMDAN and XGBoost. *Appl. Sci.* **2020**, *10*, 6593. [CrossRef]
22. Mo, H.; Sun, H.; Liu, J.; Wei, S. Developing window behavior models for residential buildings using XGBoost algorithm. *Energy Build.* **2019**, *205*, 109564. [CrossRef]
23. Fan, C.; Xiao, F.; Zhao, Y. A short-term building cooling load prediction method using deep learning algorithms. *Appl. Energy* **2017**, *195*, 222–233. [CrossRef]
24. Chen, K.Y.; Wang, C.H. Support vector regression with genetic algorithms in forecasting tourism demand. *Tour. Manag.* **2007**, *28*, 215–226. [CrossRef]

25. Zhong, H.; Wang, J.; Jia, H.; Mu, Y.; Lv, S. Vector field-based support vector regression for building energy consumption prediction. *Appl. Energy* **2019**, *242*, 403–414. [CrossRef]
26. Kazem, A.; Sharifi, E.; Hussain, F.K.; Saberi, M.; Hussain, O.K. Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Appl. Soft Comput.* **2013**, *13*, 947–958. [CrossRef]
27. Khelif, R.; Chebel-Morello, B.; Malinowski, S.; Laajili, E.; Fnaiech, F.; Zerhouni, N. Direct Remaining Useful Life Estimation Based on Support Vector Regression. *IEEE Trans. Ind. Electron.* **2017**, *64*, 2276–2285. [CrossRef]
28. Huang, K.; Guo, Y.F.; Tseng, M.L.; Wu, K.J.; Li, Z.G. A Novel Health Factor to Predict the Battery's State-of-Health Using a Support Vector Machine Approach. *Appl. Sci.* **2018**, *8*, 1803. [CrossRef]
29. Vrablecová, P.; Bou Ezzeddine, A.; Rozinajová, V.; Šárik, S.; Sangaiah, A.K. Smart grid load forecasting using online support vector regression. *Comput. Electr. Eng.* **2018**, *65*, 102–117. [CrossRef]
30. Paudel, S.; Nguyen, P.; Kling, W.; Elmitri, M.; Lacarrière, B.; Corre, O. Support Vector Machine in Prediction of Building Energy Demand Using Pseudo Dynamic Approach. *arXiv* **2015**, arxiv:abs/1507.05019.
31. Jiang, W.; He, G.; Long, T.; Ni, Y.; Liu, H.; Peng, Y.; Lv, K.; Wang, G. Multilayer Perceptron Neural Network for Surface Water Extraction in Landsat 8 OLI Satellite Images. *Remote Sens.* **2018**, *10*, 755. [CrossRef]
32. Azorin-Molina, C.; Ali, Z.; Hussain, I.; Faisal, M.; Nazir, H.M.; Hussain, T.; Shad, M.Y.; Mohamd Shoukry, A.; Hussain Gani, S. Forecasting Drought Using Multilayer Perceptron Artificial Neural Network Model. *Adv. Meteorol.* **2017**, *2017*, 5681308. [CrossRef]
33. Taki, M.; Ajabshirchi, Y.; Ranjbar, S.F.; Rohani, A.; Matloobi, M. Heat transfer and MLP neural network models to predict inside environment variables and energy lost in a semi-solar greenhouse. *Energy Build.* **2016**, *110*, 314–329. [CrossRef]
34. Iglesias, C.; Anjos, O.; Martínez, J.; Pereira, H.; Taboada, J. Prediction of tension properties of cork from its physical properties using neural networks. *Eur. J. Wood Wood Prod.* **2015**, *73*, 347–356. [CrossRef]
35. Anjos, O.; Iglesias, C.; Peres, F.; Martínez, J.; García, A.; Taboada, J. Neural networks applied to discriminate botanical origin of honeys. *Food Chem.* **2015**, *175*, 128–136. [CrossRef]
36. Chen, Y.; Song, L.; Liu, Y.; Yang, L.; Li, D. A Review of the Artificial Neural Network Models for Water Quality Prediction. *Appl. Sci.* **2020**, *10*, 5776. [CrossRef]
37. Kang, Y.; Lv, W.; He, J.; Ding, X. Remote Sensing of Time-Varying Tidal Flat Topography, Jiangsu Coast, China, Based on the Waterline Method and an Artificial Neural Network Model. *Appl. Sci.* **2020**, *10*, 3645. [CrossRef]
38. Chae, Y.T.; Horesh, R.; Hwang, Y.; Lee, Y. Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings. *Energy Build.* **2016**, *111*, 184–194. [CrossRef]
39. Kusiak, A.; Li, M.; Zhang, Z. A data-driven approach for steam load prediction in buildings. *Appl. Energy* **2010**, *87*, 925–933. [CrossRef]
40. Martínez Comesaña, M.; Febrero-Garrido, L.; Troncoso-Pastoriza, F.; Martínez-Torres, J. Prediction of Building's Thermal Performance Using LSTM and MLP Neural Networks. *Appl. Sci.* **2020**, *10*, 7439. [CrossRef]
41. Ruiz, G.R.; Bandera, C.F. Validation of Calibrated Energy Models: Common Errors. *Energies* **2017**, *10*, 1587. [CrossRef]
42. Hong, T.; Kim, J.; Jeong, J.; Lee, M.; Ji, C. Automatic calibration model of a building energy simulation using optimization algorithm. *Energy Procedia* **2017**, *105*, 3698–3704. [CrossRef]
43. Butler, D.; Dengel, A. *Review of Co-Heating Test Methodologies: Primary Research*; NHBC Foundation: Milton Keynes, UK, 2013.
44. Uriarte, I.; Erkoreka, A.; Eguia, P.; Granada, E.; Martin-Escudero, K. Estimation of the Heat Loss Coefficient of Two Occupied Residential Buildings through an Average Method. *Energies* **2020**, *13*, 5724. [CrossRef]
45. Duffie, J.; Beckman, W. *Solar Engineering of Thermal Processes*, 4th ed.; John Wiley and Sons: Hoboken, NJ, USA, 2013. [CrossRef]
46. Friedman, J.H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [CrossRef]
47. Touzani, S.; Granderson, J.; Fernandes, S. Gradient boosting machine for modelling the energy consumption of commercial buildings. *Energy Build.* **2018**, *158*, 1533–1543. [CrossRef]

48. Priscilla, C.V.; Prabha, D.P. Influence of Optimizing XGBoost to handle Class Imbalance in Credit Card Fraud Detection. In Proceedings of the 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2020; pp. 1309–1315. [CrossRef]
49. Xiong, Z.; Cui, Y.; Liu, Z.; Zhao, Y.; Hu, M.; Hu, J. Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation. *Comput. Mater. Sci.* **2020**, *171*, 109203. [CrossRef]
50. Pham, B.T.; Nguyen, M.D.; Bui, K.T.T.; Prakash, I.; Chapi, K.; Bui, D.T. A novel artificial intelligence approach based on Multi-layer Perceptron Neural Network and Biogeography-based Optimization for predicting coefficient of consolidation of soil. *CATENA* **2019**, *173*, 302–311. [CrossRef]
51. Sheela, K.; Deepa, S.N. Review on Methods to Fix Number of Hidden Neurons in Neural Networks. *Math. Probl. Eng.* **2013**, *2013*. [CrossRef]
52. Vujicic, T.; Matijević, T.; Ljucovic, J.; Balota, A.; Sevarac, Z. Comparative Analysis of Methods for Determining Number of Hidden Neurons in Artificial Neural Network. In *Central European Conference on Information and Intelligent Systems*; Faculty of Organization and Informatics Varazdin: Varaždin, Croatia, 2016.
53. Panchal, G.; Ganatra, A.; Kosta, Y.; Panchal, D. Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers. *Int. J. Comput. Theory Eng.* **2011**, *3*, 332–337. [CrossRef]
54. Doukim, C.; Dargham, J.; Chekima, A. Finding the number of hidden neurons for an MLP neural network using coarse to fine search technique. In Proceedings of the 10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010), Kuala Lumpur, Malaysia, 10–13 May 2010; pp. 606–609.
55. Liu, Y.; Liu, S.; Wang, Y.; Lombardi, F.; Han, J. A Stochastic Computational Multi-Layer Perceptron with Backward Propagation. *IEEE Trans. Comput.* **2018**, *67*, 1273–1286. [CrossRef]
56. Guresen, E.; Kayakutlu, G.; Daim, T.U. Using artificial neural network models in stock market index prediction. *Expert Syst. Appl.* **2011**, *38*, 10389–10397. [CrossRef]
57. Smith, L.N. A disciplined approach to neural network hyper-parameters: Part 1-learning rate, batch size, momentum, and weight decay. *arXiv* **2018**, arxiv:abs/1803.09820.
58. Li, M.; Soltanolkotabi, M.; Oymak, S. Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks. In Proceedings of the Machine Learning Research (PMLR), Palermo, Italy, 2020; Volume 108, pp. 4313–4324.
59. Barrow, D.K.; Crone, S.F. Cross-validation aggregation for combining autoregressive neural network forecasts. *Int. J. Forecast.* **2016**, *32*, 1120–1137. [CrossRef]
60. Eckle, K.; Schmidt-Hieber, J. A comparison of deep networks with ReLU activation function and linear spline-type methods. *Neural Netw.* **2019**, *110*, 232–242. [CrossRef]
61. Bock, S.; Weiß, M. A Proof of Local Convergence for the Adam Optimizer. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8. [CrossRef]
62. Nakama, T. Theoretical analysis of batch and on-line training for gradient descent learning in neural networks. *Neurocomputing* **2009**, *73*, 151–159. [CrossRef]
63. Devarakonda, A.; Naumov, M.; Garland, M. AdaBatch: Adaptive Batch Sizes for Training Deep Neural Networks. *arXiv* **2017**, arXiv:1712.02029.
64. Cacabelos, A.; Eguía, P.; Míguez, J.L.; Granada, E.; Arce, M.E. Calibrated simulation of a public library HVAC system with a ground-source heat pump and a radiant floor using TRNSYS and GenOpt. *Energy Build.* **2015**, *108*, 114–126. [CrossRef]
65. Cacabelos, A.; Eguía, P.; Febrero, L.; Granada, E. Development of a new multi-stage building energy model calibration methodology and validation in a public library. *Energy Build.* **2017**, *146*, 182–199. [CrossRef]
66. Fernandez Rodríguez, M.; Eguía, P.; Granada, E.; Febrero Garrido, L. Sensitivity analysis of a vertical geothermal heat exchanger dynamic simulation: Calibration and error determination. *Geothermics* **2017**, *70*, 249–259. [CrossRef]
67. Kuo, P.H.; Huang, C.J. A High Precision Artificial Neural Networks Model for Short-Term Energy Load Forecasting. *Energies* **2018**, *11*, 213. [CrossRef]

68. Martínez, S.; Eguía, P.; Granada, E.; Moazami, A.; Hamdy, M. A performance comparison of multi-objective optimization-based approaches for calibrating white-box building energy models. *Energy Build.* **2020**, *216*, 109942. [CrossRef]
69. Pilgrim, M.; Willison, S. *Dive Into Python 3*; Springer: Berlin, Germany, 2009; Volume 2.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.