energies

MDPI

Article

The Data-Driven Multi-Step Approach for Dynamic Estimation of Buildings' Interior Temperature

Stefano Villa [1] and Claudio Sassanelli [2,*]

[1] Evogy srl, Via Pastrengo 9, 24068 Seriate, Italy; stefano.villa@evogy.it
[2] Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy
* Correspondence: claudio.sassanelli@polimi.it

Received: 28 October 2020; Accepted: 11 December 2020; Published: 17 December 2020

Abstract: Buildings are among the main protagonists of the world's growing energy consumption, employing up to 45%. Wide efforts have been directed to improve energy saving and reduce environmental impacts to attempt to address the objectives fixed by policymakers in the past years. Meanwhile, new approaches using Machine Learning regression models surged in the modeling and simulation research context. This research develops and proposes an innovative data-driven black box predictive model for estimating in a dynamic way the interior temperature of a building. Therefore, the rationale behind the approach has been chosen based on two steps. First, an investigation of the extant literature on the methods to be considered for tests has been conducted, shrinking the field of investigation to non-recursive multi-step approaches. Second, the results obtained on a pilot case using various Machine Learning regression models in the multi-step approach have been assessed, leading to the choice of the Support Vector Regression model. The prediction mean absolute error on the pilot case is 0.1 ± 0.2 °C when the offset from the prediction instant is 15 min and grows slowly for further future instants, up to 0.3 ± 0.8 °C for a prediction horizon of 8 h. In the end, the advantages and limitations of the new data-driven multi-step approach based on the Support Vector Regression model are provided. Relying only on data related to external weather, interior temperature and calendar, the proposed approach is promising to be applicable to any type of building without needing as input specific geometrical/physical characteristics.

Keywords: Support Vector Regression; Machine Learning; energy and comfort management system; artificial intelligence; multi-step model; data-driven model; simulation; temperature estimation; Industry 4.0; cyber–physical system

1. Introduction

As part of the European Green Deal, the Commission proposed in September 2020 to raise the 2030's greenhouse gas emission reduction target, including removals, to at least 55% compared to 1990 [1]. This will enable the EU to move towards a climate-neutral economy and implement its commitments under the Paris Agreement by updating its Nationally Determined Contribution. This climate and energy framework is subsequent to the package adopted in 2009 [2], having the intent of setting energy and climate targets to be achieved by 2020. Despite that appreciable actions have been implemented, the European Commission assessed that EU was 50% toward the target in 2018, with reference to the buildings and appliances markets [3]. It has to be underlined that buildings are among the main protagonists of the world's growing energy consumption, employing between 40 and 45% of the total European energy expenditure [4]. In addition, inhabitant's requirements (in terms of thermal and visual comfort and indoor air quality) are supposed to augment with the need to preserve their health and productivity [5]. Flanking this, people are progressively increasing the

time spent inside buildings [6]. Thus, this sector deserves and asks for major heed to try to address energy needs through an efficient use of electricity and gas, leading to optimized consumption and related costs. Attempting to support these actions, in the last decades energy and comfort management systems (ECMS) have been developed and used to actively control systems for thermal and air quality regulation of the buildings, that can be grouped as following:

- heating, ventilation and air-conditioning (HVAC) system,
- lighting system,
- others, e.g., electrical appliances and devices.

The importance of ECMS has been highlighted by a survey made in the U.S. stating that HVAC systems are responsible for about 50% of the site energy consumption [7]. Therefore, ECMS can strongly affect the overall energy consumption, justifying the efforts of building and proposing new automated control systems to optimize HVAC performances. Notwithstanding, the lack of real-time input of dynamic elements in the buildings (e.g., the number of the occupants in the room, etc.) are hard to be managed through control systems. Indeed, users can have some habits not easily structurable as data to be used during the development of ECMSs. Moreover, required comfort levels may change, being related to both the perceptions of the occupants and the type of activity they are addressing. Thus, ECMSs are called to solve extremely complex problems. The classical model-based approach [8] consists of gathering all the data related to the buildings' physical and geometric characteristics and of modeling the dynamic processes with a non-linear, input/output, time invariant model (and simulation software are frequently used to aid this). Instead, in the last decades, data-driven approaches are surging [9,10]. This paper proposes a data-driven multi-step approach for estimating the interior temperature of a building. To address this scope, single-step or multi-step models can be used. When models are integrated in ECMS to plan the control strategy of buildings' equipment, multi-step models are preferred over single-step ones, being able to exploit the thermal inertia of the building. The multi-step approach can be applied using in a recursive way a single regression model, i.e., feeding the prediction back to the regression model. This method can result in a low long-term precision, because a model is not trained to manage its own output. Therefore, this research develops a multi-step approach training a different Machine Learning regression model for each considered future instant. The rationale behind the approach has been chosen based on two steps. First, an investigation of the extant literature on the methods to be considered for tests has been conducted, shrinking the field of investigation to non-recursive multi-step approaches. Second, the results obtained on a pilot case using the multi-step approach with various Machine Learning (ML) regression models have been assessed, leading to the choice of the Support Vector Regression (SVR) model.

The paper is structured as follows. Section 2 starts presenting a literature review concerning data-driven ECMSs approaches and about data-driven modeling for the thermal building dynamics, leading to shrink the focus on multi-step ones. Moreover, the theory behind the SVR model is introduced. Then, the research methodology adopted is described, explaining its phases, the evolution of the model proposed and the case chosen as pilot for the model development. Section 3 presents the main results of the research. First, the new SVR data-driven multi-step approach is proposed. Then, an application case is presented to assess and validate the developed model. Section 4 discusses results, evaluating the performances of the model, its absolute error and analyzing the predicted temperature profile. Finally, Section 5 concludes the paper, also unveiling its limitations and providing an overview of the future directions.

2. Materials and Methods

2.1. Research Context

2.1.1. The Data-Driven ECMS Model

The standard architecture of ECMS is depicted in Figure 1 [11]. An ECMS is an intelligent control system for buildings. It includes the use of embedded devices and communications to drive the HVAC in order to satisfy the comfort expectations of the occupants while optimizing the energy consumption [12]. The comfort expectations include parameters like humidity, temperature, CO_2 concentration and illumination levels. A series of indoor and outdoor sensors are used to get signals from the environment, send information and store them into a Smart Decision Unit (SMU) to which comfort levels (and the building usage profiles) are set and given as input references. Indeed, in the SMU happens all the decisions concerning both the management of HVAC and the lighting systems.

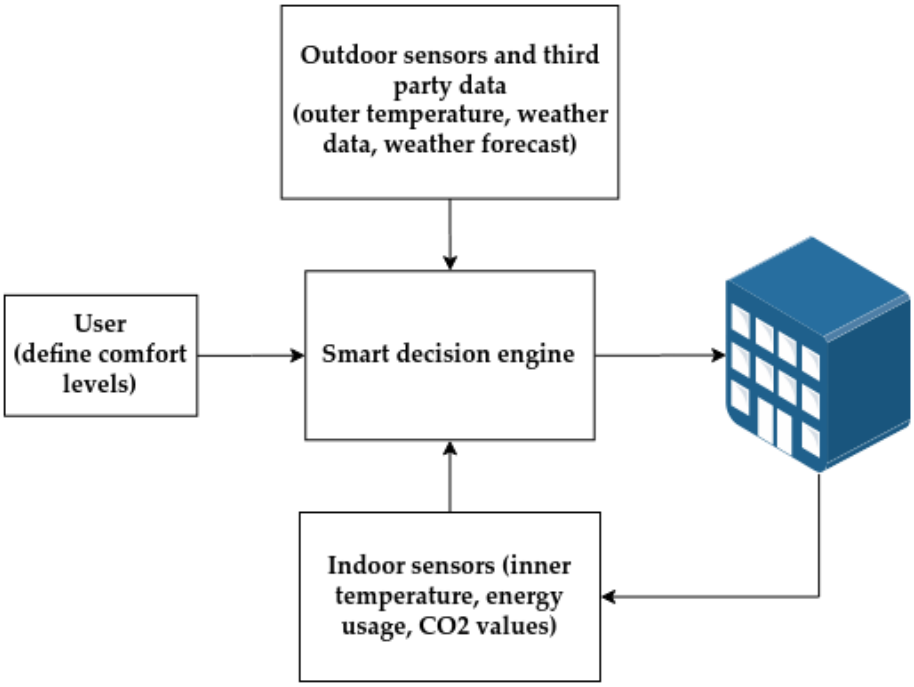


Figure 1. Standard energy and comfort management system (ECMS) architecture.

In a data-driven approach, data feed the black box predictive model, without requiring any building knowledge. The black box data-driven approach is applied in a Model Predictive Control (MPC) [13] system, that is constituted by two different blocks:

- the control block, the core of the smart decision engine, takes as inputs the comfort levels and compares them to the temperature, humidity and CO_2 concentration room feedbacks, also catering the control variables (indicating the status of the HVAC);
- the model, receiving as inputs by the control block the control variables and the external disturbances. Based on them, it provides as outputs through a simulation the environmental signals (room temperature, humidity and CO_2 concentration), also enabling the estimation of the gas or energy consumption (basing on the status of the control variables).

Data-driven approaches are typically grounded on time series forecasting via ML models. Time series forecasting is always more recurrent and strategic in engineering and applied mathematics contexts [14]. Some years ago, both mathematical model-based methods, grounded on differential equations, and the methods based on statistical mathematical models (belonging to the ARIMA

class) were considered the referring ones to solve problems of time series forecasting. Then, with the advent of digital technologies [15], bringing sensors and data storage devices, but also thanks to the performance improvements of computing processors, there was the birth of Internet of Things (IoT) [16,17]. This combined technological enhancement brought to a surge in the amount of historical data available for both academics and practitioners [18–20]. Thus, ML methods performing supervised regression, have been successfully adopted in forecasting historical time series [11] to fully exploit this huge set of historical data. ML algorithms are perceived as black boxes that are difficult to interpret. However, in reality there are techniques, called feature or variable importance, that allow to make a ranking of the input variables of the model based on their importance on the prediction of the quantity of interest [21].

Premised that, to fully exploit their potentiality, all the methodological approaches require to model the thermal dynamics of the building. Reference [22] shows that a data-driven approach based on a Random Forest regression model performs better than a standard model-based approach on the winter period for the same building that has been used in this research. Hence, this paper focuses on the data-driven approach. In particular, first it provides some related extant literature to investigate which are the available methods to be considered for tests, shrinking the field of investigation on multi-step ones. Then, based on the results obtained on a pilot case using the regression models selected from the literature, proposes a new approach based on the SVR model. In Figure 2, the possible approaches that can be adopted for modeling the thermal dynamics of the building are shown.

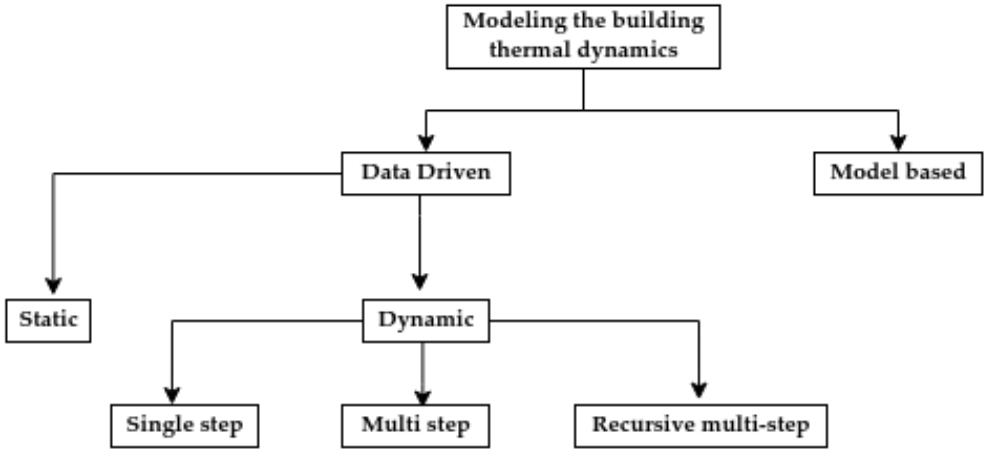


Figure 2. Extant predictive modeling approaches.

2.1.2. Machine Learning Models for Predicting the Inner Temperature

Even if [22] unveiled that data-driven approaches are more accurate than model-based approaches, it has to be raised that a static, time invariant approach was adopted, not exploiting the advantage that derives from updating the prediction with real time data. Indeed, in recent years the interest on the prediction of the inner temperature of a building through dynamic data-driven approaches is constantly growing.

Starting with dynamic single-step prediction approaches, References [23,24] show that the dynamic of the inner temperature and the humidity of buildings can be successfully captured applying recurrent neural networks models, respectively a non-linear autoregressive network with exogenous inputs (NARX) and a long short-term memory [25] (LSTM) neural network regressor. However, in an MPC context, in order to exploit the inertia of thermal dynamics when planning the control strategy, a multi-step lookahead is preferred over a single step prediction [26].

Moving the attention to multi-step approaches, in [27] the authors investigate the problem of predicting the cooling load for a building on a 24 h horizon with various ML models, both shallow and deep (i.e., respectively with few and many rounds of transformations of the input data), among which was the SVR model. In [26,27], the multi-step prediction is obtained feeding the model output to the regression model to obtain the next step prediction. However, concerning this, References [28,29] observed that a regressor is not trained to manage its own output. Hence, when the model output is given back as input of the model itself, the estimation error can result in a low long-term precision (see [30]). A solution to this problem is to train a different regressor for each future instant that is estimated. With this approach, the n-th regressor parameters are optimized for the estimation of the dynamic of the n-th instant, knowing the current system status. Due to their inner nature, recurrent models like the LSTM neural network regressor cannot be applied in this way. As a result, it is needed to understand which regression algorithm has to be applied in the multi-step approach. Since, this paper shows that the SVR model can be successfully used to predict the inner temperature for a multi-step lookahead, the model is introduced in the next sub-section.

2.1.3. The Support Vector Regression Model

Support Vector Machines (SVM) are supervised learning models that can be used for both regression and classification. This work focuses on regression for time series forecasting. For a formal introduction on SVM, see [31,32]. The idea behind SVM can be introduced considering a set $X \subseteq \mathbb{R}^d$ of n points belonging to two classes A and B. Assume that the points in X are linearly separable, i.e., there is a $(d-1)$-dimensional hyper-plane Π such that all the points of X on one side of Π belongs to class A and all points on the other side of Π belongs to class B; see Figure 3.

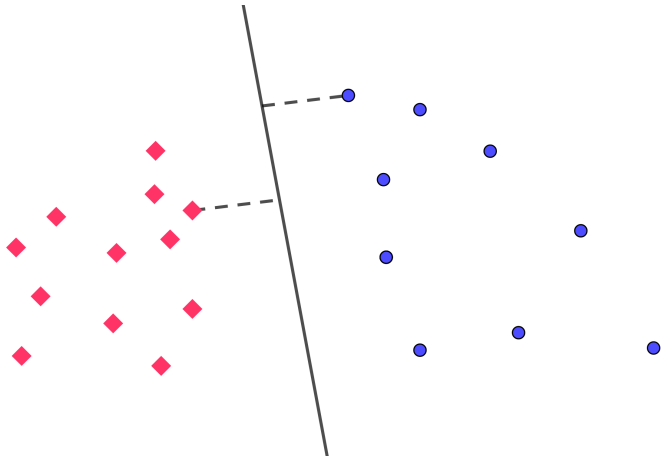


Figure 3. When the data are linearly separable, the Support Vector Machine (SVM) algorithm finds the maximum-margin hyper-plane.

The task for a classification model is to decide if a new given point belongs to class A or class B. Without loss of generality, we can assume that if x_i is a point of class A, it has label $y_i = 1$, else $y_i = -1$. The SVM learning algorithm finds the maximum-margin hyper-plane, i.e., the plane that has the maximum distance from the closest point of each class in X. The plane can be represented by the equation

$$\langle \omega, x \rangle + b = 0, \tag{1}$$

where $\langle \_, \_ \rangle$ is the standard dot product in $\mathbb{R}^d$, $\omega \in \mathbb{R}^d$ and $b \in \mathbb{R}$. When the data are linearly separable, the problem of finding the maximum-margin hyper-plane can be stated as:

$$\underset{\omega,b}{\text{minimize}} \frac{1}{2}\|\omega\|^2 \text{ subject to } y_i(\langle \omega, x_i \rangle + b) \geq 1 \; \forall \; i \in \{1, ..., n\}, \tag{2}$$

because this condition is equivalent to maximizing the margin of separation between the two classes of points. The maximum-margin hyper-plane is completely determined by the points of X that are the closest to it, that are called support vectors.

In a more general scenario, the points are not separable by a hyper-plane. Then, the constraints of the optimization problem are weakened imposing a linear penalty when the margin constraint is violated, so that Equation (2) becomes:

$$\begin{array}{l} \text{minimize}_{\omega,b,\xi} \dfrac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{subject to } y_i(\langle\omega, x_i\rangle + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0, \ \forall i \in \{1, ..., n\}. \end{array} \tag{3}$$

The value C is a positive parameter that can be used to balance the constraints violations versus the maximization of the margin.

The algorithm for regression is an extension of the one for classification, where instead of requiring the condition $y_i(\langle\omega, x_i\rangle + b) \geq 1 - \xi_i$ for all i, the soft-constraints

$$y_i - \langle\omega, x_i\rangle + b \leq \epsilon - \xi_i \text{ and } \langle\omega, x_i\rangle + b - y_i \leq \epsilon - \xi_i^* \tag{4}$$

are imposed for all $i \in \{1, ..., n\}$. The values ξ_i and ξ_i^* are positive variables.

Even if we can now deal with non-linearly separable sets, the model is still linear. To generalize it, a non-linear function Φ can be found to map the set X into a possibly higher dimensional space $\mathbb{R}^D$ where the points are linearly separable. Instead of requiring that the function Φ is known explicitly, the SVM learning algorithm uses another function

$$k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}, \tag{5}$$

such that

$$k(x_i, x_j) = \left\langle \Phi(x_i), \Phi(x_j) \right\rangle_{\mathbb{R}^D} \text{ for all } x_i, x_j \in X, \tag{6}$$

where $\langle \_, \_ \rangle_{\mathbb{R}^D}$ is the standard dot product in $\mathbb{R}^D$. The function k is called kernel function and it allows to compute the separating plane in the transformed space without knowing explicitly the function Φ. Among all the possible kernel functions, one of the most widely used is the Gaussian radial basis function

$$k(x_i, x_j) = \exp(\gamma\|x_i - x_j\|^2) \text{ with } \gamma > 0. \tag{7}$$

2.2. Research Methodology

To address the gap raised in Section 1 dealing with ECMS and provide a consistent data-driven approach, a research methodology was designed (taking as a reference [33,34]). The main research tradition considered, being tested in real-world settings, is system development research [35]. Indeed, observation, theory building and experimentation represent different steps to develop a prototype that represents, along the research conduction, both a proof of concept and a basis for iteratively improving the research in a qualitative process [36]. The research methodology, shown in Figure 4, is composed by three phases:

1. conceptualization: a literature review has been conducted to find the best approach for the predictive model and define its first conceptualization;
2. development: a pilot plant has been chosen to practically test the data-driven multi-step approach [22]. Iterative testing led to improve the data-driven model until its final design was obtained;
3. validation: a final validation has been conducted in the same pilot case but in a different season and using the multi-step approach with several regression models to verify its flexibility with different boundary conditions. Based on the results obtained, further improvements have been implemented on the model final design in order to obtain the full prototype.

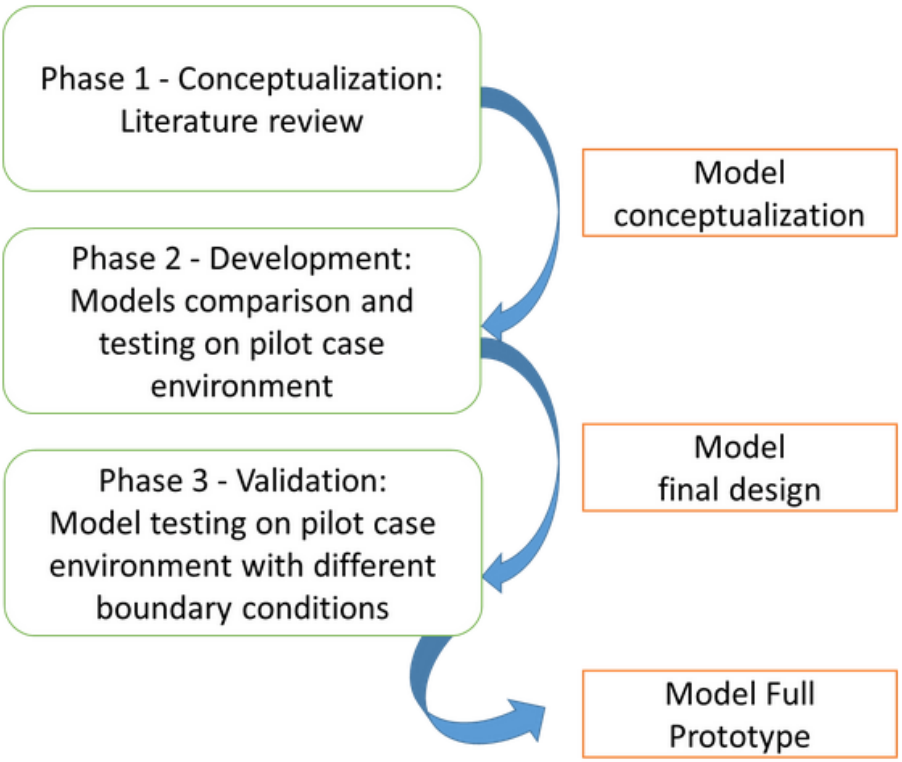


Figure 4. Research methodology: main phases.

2.3. Model Evolution

The development of the new approach involved a team composed by a business expert, a data scientist and a software developer.

Along the research time-lapse, the model has been improved several times. Once the data-driven model has been chosen to the detriment of the model-based one, the focus shifted on the improvement of the data-driven approach. In particular, in the step between development and validation, the model has been evaluated comparing its results with its previous version from [22]. Here, the application of the model on different boundary conditions, also considering other regression models, and the consequent introduction of real time data to progressively improve the solution while the time advances, makes the two versions qualitatively different. For this reason, Table 1 is provided. It explains the details related to the improvements implemented on the final design of the SVR data-driven approach to obtain the full prototype (presented in Section 3.1).

As shown in Table 1, it can be seen that the new model uses real-time data collected from the sensors installed on the field and stored on SimonLab, the ECMS platform of the Italian SME Evogy srl [14]. The real-time data are used with a moving horizon approach [26]: at every iteration step the real data for the input variables are read from the field and are used as model inputs to predict a future window of values for the inner temperature, correcting progressively the prediction as new data become available. This approach will pave the way for the implementation of an MPC algorithm, which is the final target.

Table 1. Data-driven approach evolution.

	Random Forest Data-Driven Approach [22]	**SVR Data-Driven Multi-Step Approach**
Team involved	• 1 business intelligence expert • 1 data scientist	• 1 business intelligence expert • 1 data scientist • 1 software developer
Pilot application	*The Bridge* office (Evogy's headquarter) (winter data)	*The Bridge* office (Evogy's headquarter) (summer data)
Origin of the dataset	The building data are acquired from IoT sensors and collected on Evogy's proprietary platform	The building data are acquired from IoT sensors and collected on Evogy's proprietary platform
Model main features	• Static approach • Random Forest model used for regression	• Moving horizon approach • Improved features engineering [37], e.g., using the mean of the previous values of the inner temperature as model input • SVR model used for regression • Multi-step prediction
Computational resources	Local computer for model development and evaluation	• Local computer for model development and evaluation • Cloud virtual machine for model continuous real time testing
Driving reasons for the development	To compare model-based and data-driven approaches	• Evolve the model, introducing real-time data and continuous predictions • Implementation of a prototype software for MPC of the system

2.4. The Application Pilot Case

In this sub-section, the application case, used as pilot for the development of the proposed approach, is introduced, describing its main characteristics.

The planimetry of the building space assessed, *The Bridge* office (Evogy's headquarter), is shown in Figure 5 taken from Evogy's ECMS. The office is on a single floor and has a main open space, two meeting rooms and a bathroom. The heating and cooling systems have only two possible states (on or off). The inner temperature is read by two sensors located in the open space close to the two entrances of the office (respectively on the east and on the west sides). The datum read by the two sensors is averaged to obtain a value less dependent on local conditions and refers to the temperature of the open space. The outer temperature and the other weather features are retrieved through a third party service provided by REST apis.

The cooling system relies on electricity and it just performs air re-circulation, cooling the air inside the building. The windows of the office can be opened for free cooling, but it is not possible to know directly from the data if the windows are open or closed in a certain instant. The set of data used to evaluate the model spans the period from 18 August, to 1 September 2020; see also Figure 6. Note the following facts.

- During the first two days of the set, the inner temperature follows very closely the outer temperature profile. This is caused by the fact that the windows are open.
- During the first weekend (22 and 23 August) the air cooling system was off and the windows are closed, hence the temperature grows and remains well above 30 degrees Celsius on both days.
- On the second weekend (29 and 30 August) the outer temperature is cooler than the inner temperature which, consequently, remains substantially stable varying between 26 and 27 degrees.
- In all other days, the inner temperature decreases when the cooling system is on, while it grows or remains stable when the cooling system is off.

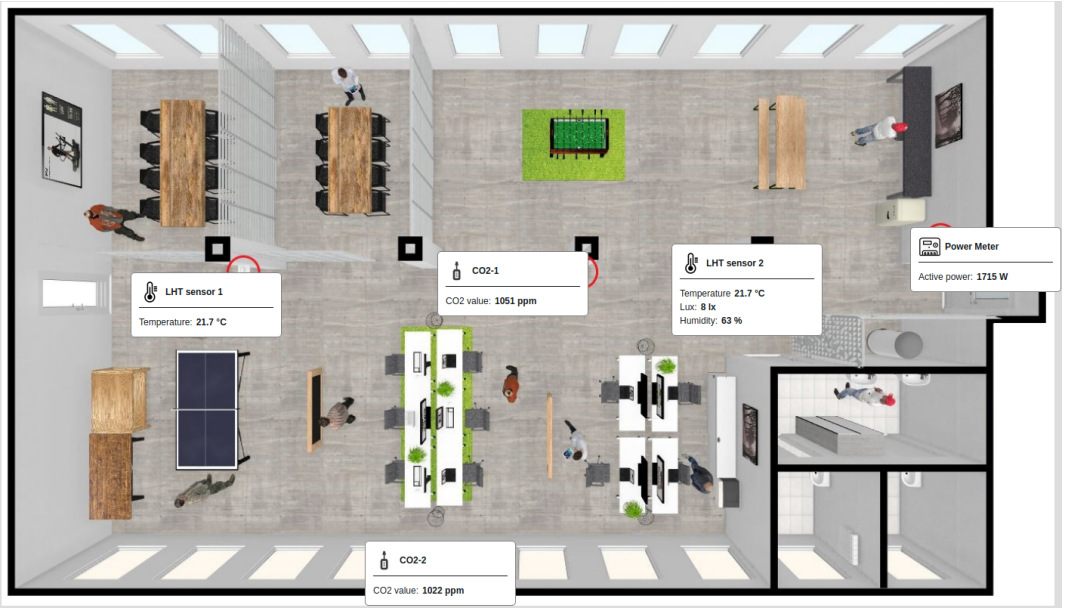


Figure 5. Planimetry of *The Bridge* office, Evogy's headquarter, used as case study.

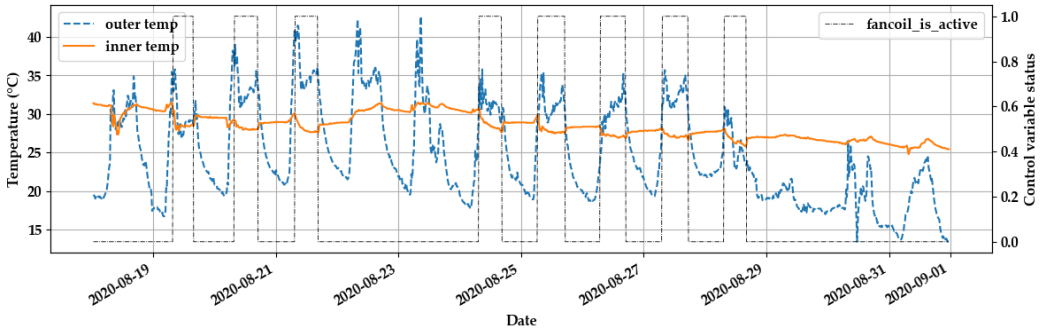


Figure 6. The set of data used as the model test set.

In Figure 6, the continuous orange line is the inner temperature, the dashed blue line the outer temperature and the black dashed-dotted line the status of the air cooling system (1 for "on" and 0 for "off" on the right y-axis). When the inner temperature follows exactly the outer temperature, the windows are open. The target of the present approach is the prediction of the building inner temperature but similar reasonings can be applied to other variables like the electricity consumption.

3. Results

This section presents the results obtained through the adoption of the research methodology presented above. Based on the research context proposed in Section 2.1, the SVR data-driven approach characterized by a multi-step structure is presented, introducing its input and output variables in Section 3.1, while in Section 3.2 the results of the application conducted on the pilot case *The Bridge* office are provided to give practical evidence to the benefits deriving from the model adoption.

3.1. The New Data-Driven Multi-Step Approach Based on the SVR Model

Based on the research context provided in Section 2.1, the new approach is proposed.
The data-driven predictive model on which the approach is based uses the following input variables:

- datetime variables: minute within the day, day within the year, if the day is a weekday or in the weekend;

- status of the cooling system: for how many instants the cooling system has been consecutively active and not active, respectively, in the last 30, 60 and 120 min;
- quantity of rain fell in the last 30 and 60 min;
- cloud cover percentage in the last 30 and 60 min;
- mean outer temperature in the last 30, 60 and 120 min;
- difference between the current value of the outer temperature and the value, respectively, 15, 30 and 60 min ago;
- difference between the inner and outer temperature, respectively, 15, 30 and 60 min ago;
- difference between the current value of the outer temperature and its value in the prediction instant.

In order to capture the cyclic nature of the minute within the day and of the day within the year, the sine and cosine of their values are used.

Let t_0 be the current instant, let N be an integer and n a number between 1 and N. Assuming that the target is the estimation of the system dynamic for N future instants $t_1, \ldots, t_N$, the predictive model is the composition of N sub-regressors, denoted F^1, ..., F^N: the n-th regressor takes as inputs the most recent observations of the input data collected by the sensors and, possibly, all the available information for the future instants from t_1 to t_{n-1}, like the control strategy for the HVAC system and the weather forecast, and gives as output the system status at the instant t_n. Hence, each F^n is a function from $\mathbb{R}^d$ to $\mathbb{R}$ where $d \in \mathbb{N}$ is the dimension of the input data.

Following [26], the model does not directly estimate the inner temperature dynamic, because this function is not easily learned if the effect of a certain action on a short interval is small. Instead, the model's target variable is the difference between the current observation and the target future observation of the inner temperature, i.e., denoting t_0 the current instant, $T_{\text{in}}(t)$ the inner temperature at time t, the n-th regressor predicts the value:

$$\Delta T_{\text{in}}(t_n|t_0) = T_{\text{in}}(t_n) - T_{\text{in}}(t_0) \text{ with } t_n > t_0. \tag{8}$$

Let us denote:

- t_0 the current instant,
- $T_{\text{in}}(t_N)$ the inner temperature measured by the sensors at time $t_N > t_0$,
- $d(t_0)$ the data collected from the field described above for the instant t_0,
- $\{a(t_1), \ldots, a(t_{N-1})\}$ the sequence of control actions for the instants $t_1, \ldots, t_{N-1}$,
- $\{T_{\text{out}}(t_1), \ldots, T_{\text{out}}(t_{N-1})\}$ the outer temperature forecast for the instants $t_1, \ldots, t_{N-1}$,
- $o(t_n|t_0)$ the prediction of the temperature delta between time t_n and t_0.

For the application case, the control action values are 0 or 1 respectively for off and on.
For each instant t_n with $n = 1, \ldots, N$, there is a trained SVR model F^n which estimates the inner temperature delta for the instant t_n, taking in input the current datum $d(t_0)$, the sequence of future actions $a(t_1), ..., a(t_{n-1})$ and the future outer temperatures $T_{\text{out}}(t_1), \ldots, T_{\text{out}}(t_{N-1})$. Hence, the global model composed by the N sub-regressors can be described by the equations:

$$o(t_n|t_0) = F^n\left(d(t_0), T_{\text{out}}(t_1), \ldots, T_{\text{out}}(t_{N-1}), a(t_1), \ldots, a(t_{N-1})\right) \text{ for } n = 1, \ldots, N \tag{9}$$

and the estimate $T_{\text{in}}^{\text{pred}}(t_n|t_0)$ of the inner temperature $T_{\text{in}}(t_n)$ for the instant t_n obtained at t_0 is

$$T_{\text{in}}^{\text{pred}}(t_n|t_0) = T_{\text{in}}(t_0) + o(t_n|t_0). \tag{10}$$

The performance comparison metrics adopted to evaluate the predictive model are the mean absolute error (MAE) and the mean absolute percentage error (MAPE), defined respectively as:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - p_i|\,, \tag{11}$$

and

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - p_i}{p_i}\right| \times 100, \tag{12}$$

where y_i is the actual value and p_i is the predicted value.

3.2. Application Case

The data-driven approach described in Section 3.1 has been implemented and tested on the application pilot case introduced in Section 2.4. In addition, the results coming from the application of the SVR model have been compared with the ones coming from other commonly used regression algorithms, i.e., Random Forest [38], Extreme Gradient Boosting [39] and Multilayer Perceptron neural network [40]. The results of the algorithms' application are shown in Appendix A.

The SVR model implementation used in this work comes from the Python open-source library Scikit-Learn [41]. The selection of the best parameters of the model is done splitting the available dataset in three disjoint and time consecutive sets: the first is used to train the algorithm, the second to select and validate the best set of hyper-parameters and the last to test the trained model. Each SVR sub-regressor is treated as a stand-alone model and the best parameters are chosen by minimizing the sum of the absolute errors made by each regressor on the validation set. The applied parameters selection technique explores a grid varying the parameters γ and C (see Section 2.1.3).

For the training, testing and validation of the model, a dataset has been collected. The dataset refers to the summer period 2020. The 14 days from 18 August, to September 1st are used as test set (see Section 2.4), and the 15 days from 3 August, to 17 August as validation set. The number of training days is one of the model parameters that are let vary during the parameters fine-tuning process: the last day is always 2 August and different sizes were tried from 15 to 210 days. To guarantee that the data are independent, the three sets do not overlap. Then, for the training process of the SVR model a grid search has been employed. The possible values are:

- γ in 0.001, 0.01, 0.05, 0.1, 0.25 and 0.5;
- C in 0.001, 0.01, 0.1, 0.5, 1 and 5.

The kernel function is the Gaussian kernel introduced in Equation (7). The data are sampled every 15 min and a multi-step model composed by 32 sub-regressors is considered, so that the prediction horizon is 8 h (480 min).

Figures 7 and 8 show the surfaces of the MAE and MAPE values with respect to the prediction horizon and the length of the training set. The results refer to the test set. As one may reasonably expect, both metrics grow with the minutes offset. The MAE is almost equal to zero (0.1 ± 0.2 °C) when the minutes offset from the prediction instant is 15 min, for all the values of the number of training days. The best values are found for 45 training days. In this case the error starts from 0.1 ± 0.2 °C when the offset from the prediction instant is 15 min and reaches 0.3 ± 0.8 °C for a prediction horizon of 8 h. For less than 45 training days, the error is higher, probably because the regression model has less data for learning the system dynamics. Up to 120 training days, the values of the MAE and of the standard deviation of the absolute error are very similar to the ones observed for 45 training days, while for more than 120 training days the error grows. This situation can be explained by the fact that data referring to many months before the test set, not only do not add information to the regression algorithm, but they add noise in the training process resulting in a higher error. The same observations are valid for the absolute percentage error.

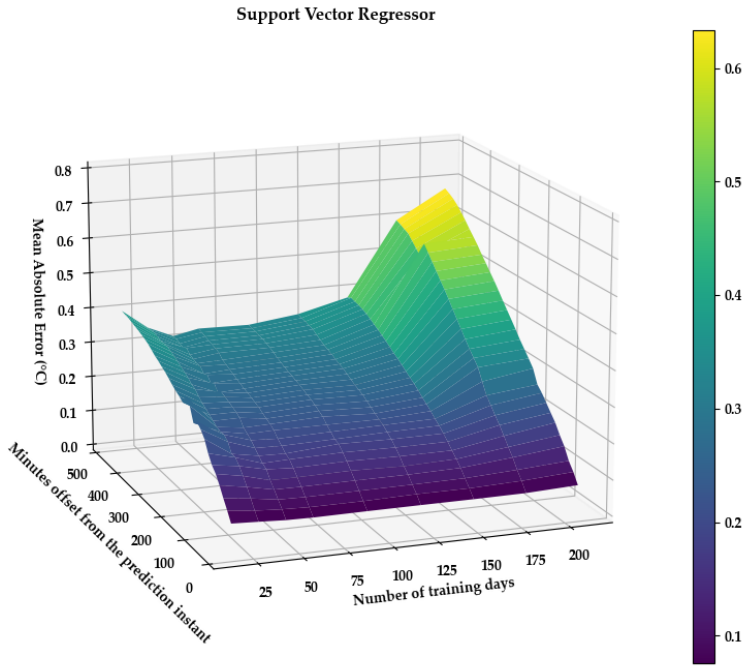


Figure 7. Surface describing the variation of the MAE with respect to the minutes horizon from the prediction instant (x-axis) and the number of days used to train the model (y-axis). The results refer to the test set.

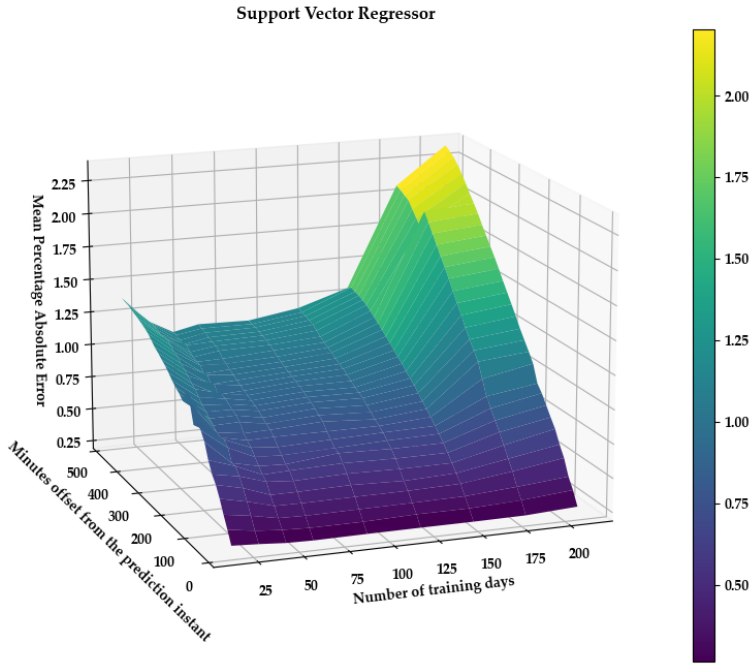


Figure 8. Surface describing the variation of the MAPE with respect to the minutes horizon from the prediction instant (x-axis) and the number of days used to train the model (y-axis). The results refer to the test set.

4. Discussion

In this section the absolute error is analyzed to fully discuss the results introduced in Section 3.2. Considering the average of the MAE and of the standard deviation of the absolute error on the test set with respect to the number of training days, the minimum is reached for 45 training days with a value

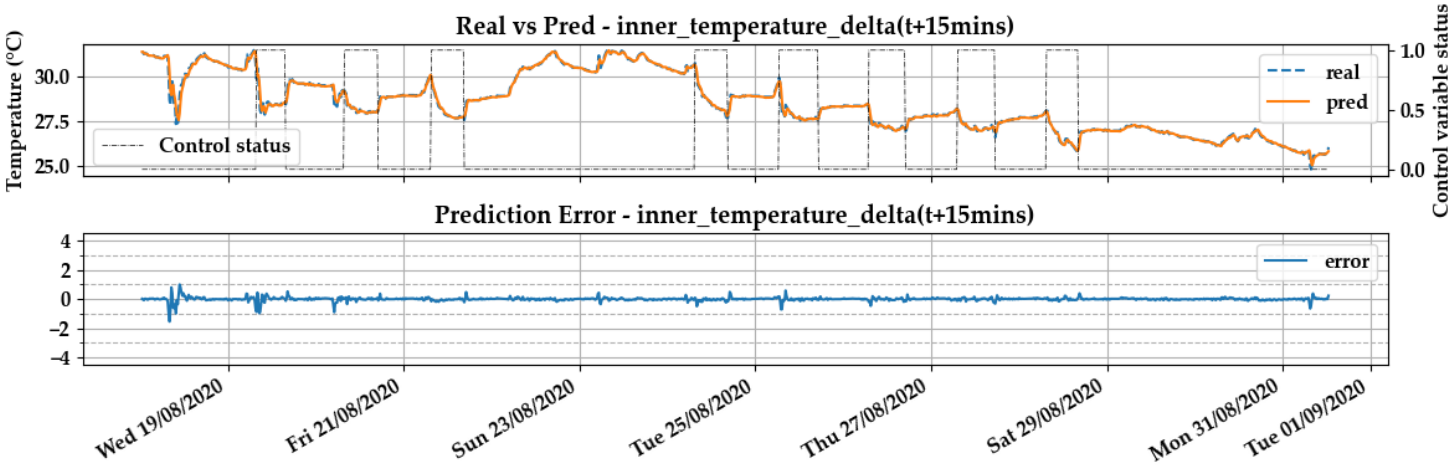
of 0.23 ± 0.69 °C. The values for 30, 60, 90 and 120 training days are very similar. The MAPE of the test set has its minimum for 45 training days with a value of 0.81 ± 2.31. The time necessary to train the model depends on the processing computer characteristics and on the dimension of hyper-parameters grid used for the model fine-tuning. On a computer with CPU processor having 1.80 GHz frequency, the time grows with the number of training days, starting from 7 min for 15 days and up to 320 min for 210 training days. See Table 2 for more details.

Table 2. Average of the absolute error mean (MAE) and standard deviation (STDAE), and of the absolute percentage mean (MAPE) and standard deviation (STDAPE) of the predictions on the test set grouped by the number of training days. The last column shows the total training time for tuning the model (the hyper-parameters tuning with the grid-search procedure is included).

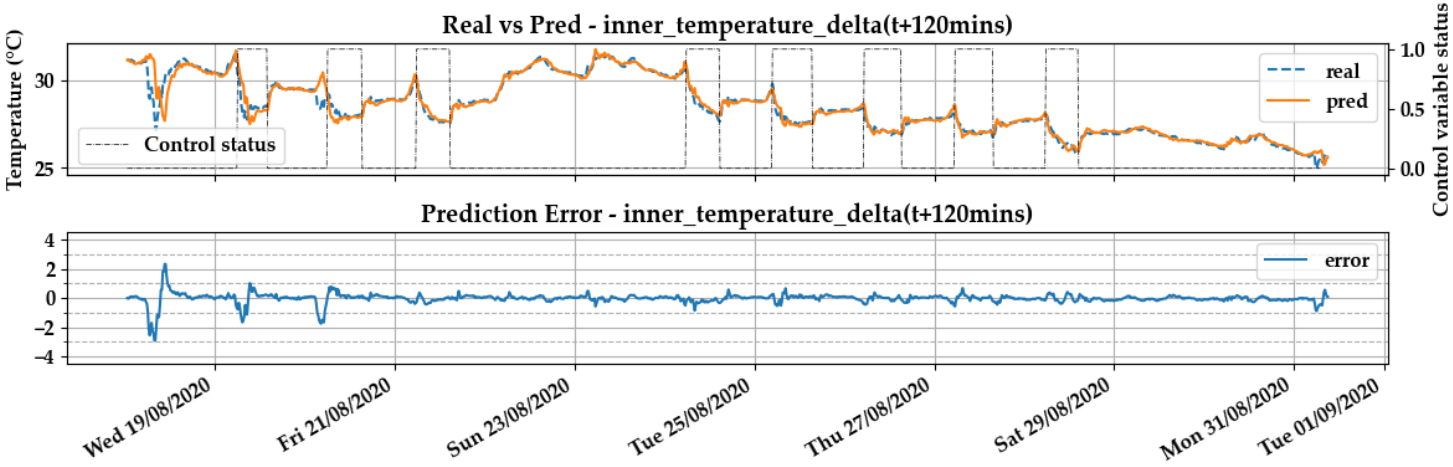
Training Days	Average MAE	Average STDAE	Average MAPE	Average STDAPE	Training Time
15	0.29	0.40	1.02	1.35	416 s
30	0.26	0.36	0.88	1.20	1126 s
45	0.23	0.34	0.81	1.15	2281 s
60	0.23	0.37	0.81	1.23	3640 s
90	0.24	0.37	0.82	1.23	7424 s
120	0.24	0.38	0.84	1.26	8049 s
150	0.27	0.40	0.91	1.32	8531 s
180	0.38	0.49	1.33	1.65	13,418 s
210	0.41	0.50	1.42	1.68	19,605 s

An exhaustive analysis on the output prediction needs to be catered. The predictive model used in the simulations is trained on 45 days. Overall, the model shows very good performances and satisfies the expectations, except when unpredictable events (e.g., windows opening) happen. These situations result in a higher prediction error; see also Figure 9 which shows the results for the sub-regressors F^1, F^8, F^{16}, F^{24} and F^{32}. In Figure 9a–e, in the upper plot the blue dashed line is the real value of the temperature measured by the sensors, the orange continuous line is the predicted temperature profile and the thin black dashed-dotted line is the status of the control variable—value is 1 for "on", 0 for "off"; the lower plot shows the prediction error. In the prediction error plots the horizontal lines grid are one-degree steps. The plots refer to the test set.

The model makes the most significant errors during the first three days of the test set. On the first day of the considered period (18 August) the temperature inside the building decreases strongly even if the air conditioning system is off. In this situation, the absolute error increases together with the minutes offset, reaching a maximum of 4 °C when the prediction horizon is 6 h and 8 h (respectively Figure 9d,e). This can be explained by the fact that the building windows were open. A similar situation can be seen on 20 August, where the inner temperature drops before the air conditioning system is turned on. In these two situations two different phenomena occur: on 18 August a horizontal shift between the real and the predicted temperature profiles is observed. This is caused by the fact that the predictive model expects that the inner temperature grows slowly during the day, because the air conditioning is off. Hence it "projects" the current temperature into the future. On the other hand, the temperature generally grows before the air conditioning is turned on, so that on 20 August the predictive model expects a similar behavior. This is the cause of the high difference between the predicted and the real profiles, especially for the furthest horizons. Finally, on 19 August, the inner temperature falls steeply from more than 31 °C to 28 °C. In this case, the air conditioning system was on, but the sudden and strong decrease in the temperature is due again to the open windows. In the rest of the dataset, where the thermal dynamic is controlled by the air conditioning system and the windows are closed, the prediction absolute error is generally below 1 °C even for the furthest prediction horizon.

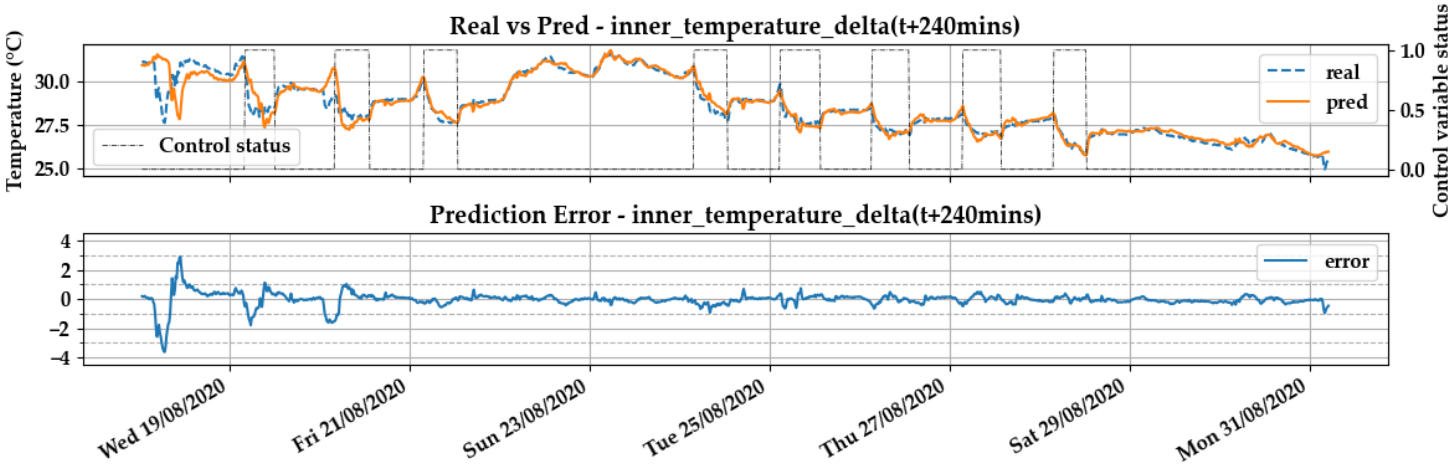


(**a**) Prediction horizon is 15 min from the prediction instant.

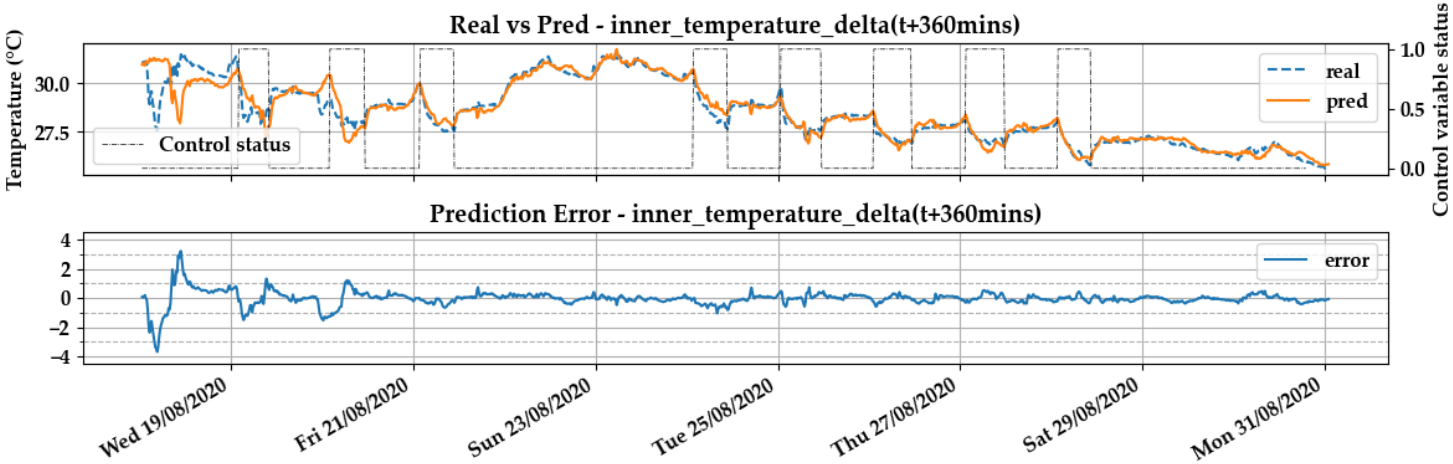


(**b**) Prediction horizon is 120 min from the prediction instant.

Figure 9. *Cont.*

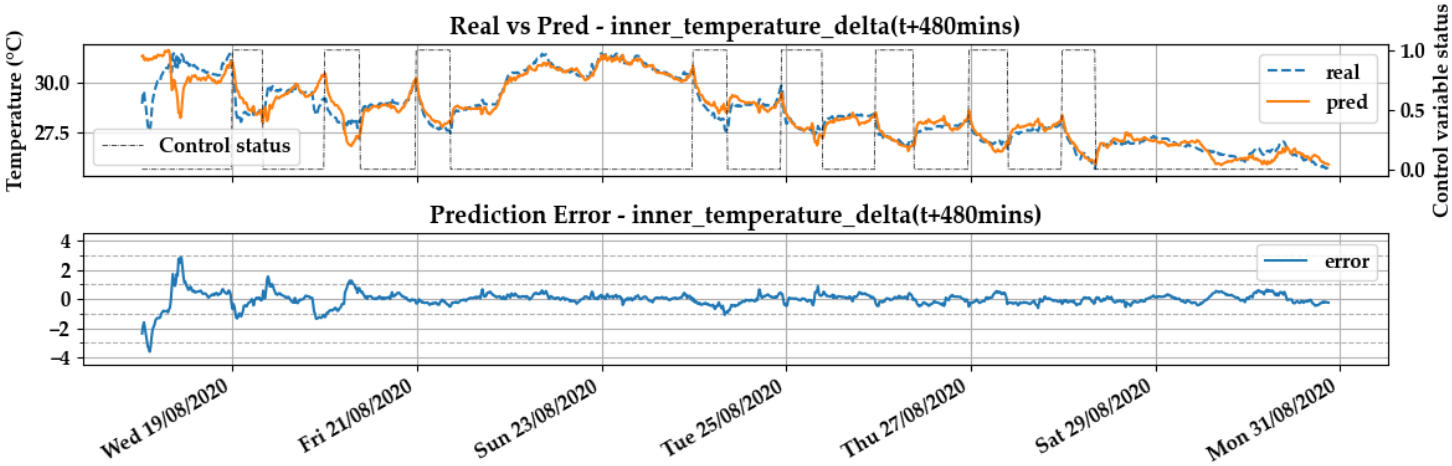


(**c**) Prediction horizon is 240 min from the prediction instant.



(**d**) Prediction horizon is 360 min from the prediction instant.

Figure 9. *Cont.*



(**e**) Prediction horizon is 480 min from the prediction instant.

Figure 9. In (**a**–**e**) the upper plot shows the comparison between the real and the predicted values of the inner temperature. The results refer to the test set.

5. Conclusions

In this research a data-driven multi-step approach for dynamically estimating the interior temperature of a building has been proposed. The predictive model can be integrated in an ECMS and is based on the SVR model. The input data are the weather conditions and the current inner temperature of the building. The novelty of the model proposed stands in its foundation on the multi-step approach to estimate the interior temperature of a building. Indeed, single-step or multi-step models can be used to address this scope. When models are integrated in ECMS to plan the control strategy of buildings' equipment, multi-step models are preferred over single-step ones, being able to exploit the thermal inertia of the building. The multi-step approach can be applied using in a recursive way a single regression model, i.e., feeding the prediction back to the regression model. This method can result in a low long-term precision, because a model is not trained to manage its own output. Therefore, this research develops a multi-step approach training a different Machine Learning regression model for each considered future instant. The SVR regression model has been chosen based on two steps. First, an investigation of the extant literature on the methods to be considered for tests has been conducted, shrinking the field of investigation to non-recursive multi-step approaches. Second, the results obtained with the regression model on a pilot case have been assessed and compared with other ones using as well a multi-step approach. The model has been applied in Evogy's headquarter, *The Bridge* office, and integrated in its ECMS. This building space represents the pilot case where the model has been applied to verify that it can effectively reproduce the thermal dynamic of a given building, even for a future horizon of 8 h from the prediction instant. The error analysis shows that the predictive model accuracy depends on the number of days that are used to train the SVR multi-step model. For the optimal hyper-parameters retrieved through a grid search method, the best results are found for 45 training days. In this case, the error starts from 0.1 ± 0.2 °C when the offset from the prediction instant is 15 min and reaches 0.3 ± 0.8 °C for a prediction horizon of 8 h. Moreover, the results highlight that the predictions are reliable, given that the thermal dynamic is not influenced by unpredictable variables like the windows opening by the building occupants.

As in most ML models, the main limitation of the approach derives from the fact that it is not robust to changing conditions and it requires periodical re-training or enrichment of the training dataset to perform well.

This work contributes to knowledge providing a new model that has been developed based on both the gaps raised by the extant literature and the data from the field of the pilot application case conducted. In addition, the model proposed contributes to practice providing an affordable tool to foster the energy management of buildings, either civil or tertiary. Indeed, the model can be applied to different types of buildings (with heterogeneous characteristics), provided that data collected from the field are available. Hence, the present solution has the practical advantage of being generalizable and scalable more easily than a standard model-based approach (typically proposed in literature as the referring one to support ECMS) requiring a deep knowledge of the building features. Indeed, the proposed approach is promising to be applicable to any type of building without needing as input specific geometrical/physical characteristics. Moreover, the data-driven predictive model proposed can be easily implemented with open source libraries, so that it can be integrated as part of larger software architectures.

From a managerial perspective, the model can help building managers to better plan and schedule the use of HVAC systems based on the temperature forecast provided. In addition, it would help to reduce energy consumption, its environmental impact and also the related costs, not neglecting the enhanced social impact (with a higher degree of users' comfort level).

From a policymakers' perspective, the model proposed can be a powerful tool to help governments in reducing energy consumption and its related environmental impact.

Finally, some limitations have to be reported, opening at the same time new rooms for further investigations. So far, the model has been tested on a single case, *The Bridge* office, limiting its adoption to a single market (civil buildings) and using data coming from the summer period. The model can be

further tested with more application cases and applied in an MPC problem. In addition, the model proposed is based on tests conducted with summer data. Further tests with data belonging to different seasons of the year will lead the authors to strengthen their new position in favor of the multi-step approach using the SVR model in place of other ML regression models and to choose the most reliable overarching approach. Supporting this, the adoption and use of Evogy's ECMS [14], flanked with the model proposed, would support its diffusion on different cases.

Author Contributions: Conceptualization, S.V. and C.S.; formal analysis, S.V. and C.S.; investigation, S.V.; methodology, C.S. and S.V.; software, S.V.; Supervision, C.S; data curation, S.V.; writing—original draft preparation, S.V. and C.S.; writing—review and editing, C.S.; visualization, S.V. and C.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work has received funding from both Evogy srl and the European Union's Horizon 2020 research and innovation program under grant agreement No 872548. In any case, the present work cannot be considered as an official position of the supporting organization, but it reports just the point of view of the authors.

Acknowledgments: The authors want to thank both Evogy srl and the European Commission research and innovation program under the project DIH4CPS.

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

Abbreviations

The following abbreviations are used in this manuscript:

HVAC	Heating, ventilation and air-conditioning
ECMS	energy and comfort management system
IoT	Internet of Things
MPC	model predictive control
ML	Machine Learning
MAE	mean absolute error
SVR	Support Vector Regressor

Appendix A. Comparison with Other Regression Algorithms

In this section, a comparison between the results obtained with the SVR model and three other models is given. The considered models are:

- Random Forest (implementation of Scikit-Learn library);
- Extreme Gradient Boosting (implementation of the xgboost library [39]);
- Multilayer Perceptron (implementation of Scikit-Learn library).

For the models' hyper-parameter optimization, the grid search procedure described in Section 3 has been used with the following parameters:

- Random Forest:
  - tree maximum depth: 5, 10, 15, 20 and 25;
  - minimum number of samples to be in a leaf node: 1, 3, 5, 7, 10, 12, 15, 17 and 20;
  - number of estimators: 100.
- Extreme Gradient Boosting:
  - learning rate: 0.005, 0.01, 0.05 and 0.1;
  - number of estimators: 100, 250 and 500;
  - tree maximum depth: 5 and 10.
- Multilayer Perceptron:
  - number of hidden layers: from 1 to 5;
  - activation function: tanh and ReLU function;

– initial learning rate: 0.1, 0.01 and 0.001.

For the Multilayer Perceptron, the number of neurons in each hidden layer is proportional to the dimension of the input data. Denoted with N, the dimension of the input data, the closest layer to the input layer has $0.66 \times N$ neurons, the second $0.5 \times N$ neurons, the third $0.33 \times N$ neurons, the fourth $0.2 \times N$ neurons and the last $0.1 \times N$ neurons. In the grid search procedure, the right layers (i.e., the closest to the output layer) are progressively removed.

Figures A1 and A2 shows respectively the surfaces describing the MAE and the MAPE obtained on the test set presented in Section 3.2 for each regression model. Our analysis shows that the SVR is the best performing model on the test set data. See also Tables A1–A4, that show the average of the mean and of the standard deviation of the absolute error and of the absolute percentage error for each algorithm.

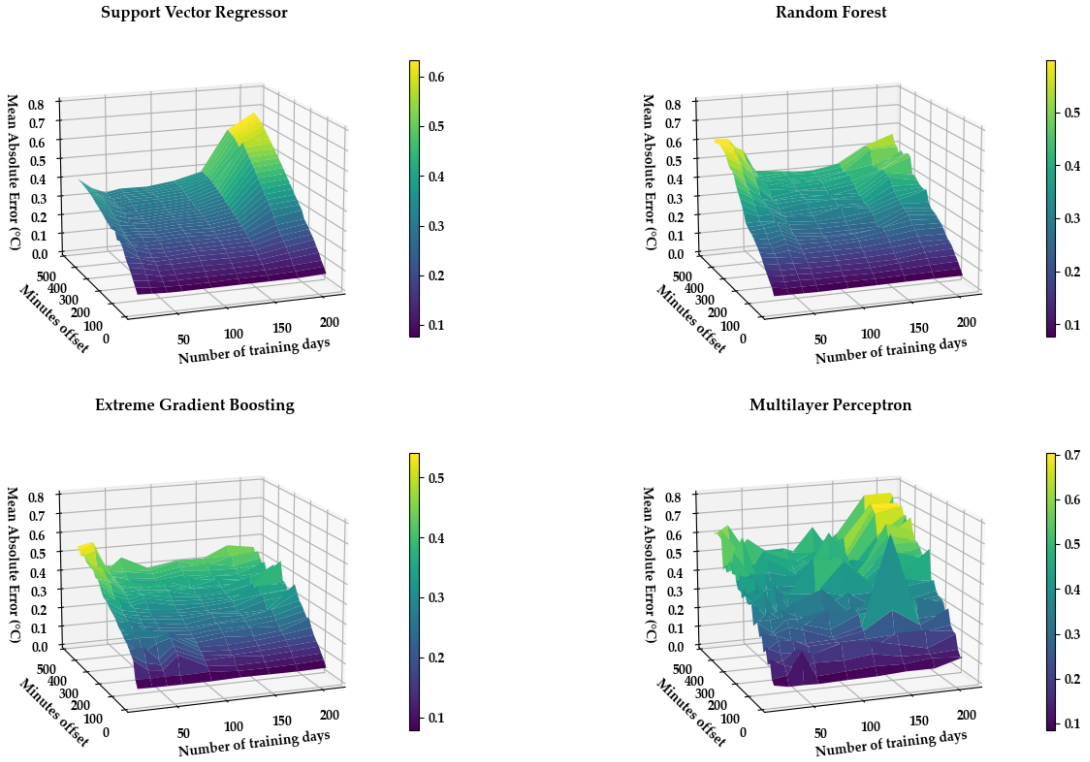


Figure A1. Surface describing the variation of the MAE with respect to the minutes horizon from the prediction instant (x-axis) and the number of days used to train the model (y-axis) for the various regression models tested. The results refer to the test set.

Table A1. Average of the absolute error mean (MAE) and standard deviation (STDAE), and of the absolute percentage mean (MAPE) and standard deviation (STDAPE) of the predictions obtained with the SVR model on the test set grouped by the number of training days.

Training Days	Average MAE	Average STDAE	Average MAPE	Average STDAPE
15	0.29	0.40	1.02	1.35
30	0.26	0.36	0.88	1.20
45	0.23	0.34	0.81	1.15
60	0.23	0.37	0.81	1.23
90	0.24	0.37	0.82	1.23
120	0.24	0.38	0.84	1.26
150	0.27	0.40	0.91	1.32
180	0.38	0.49	1.33	1.65
210	0.41	0.50	1.42	1.68

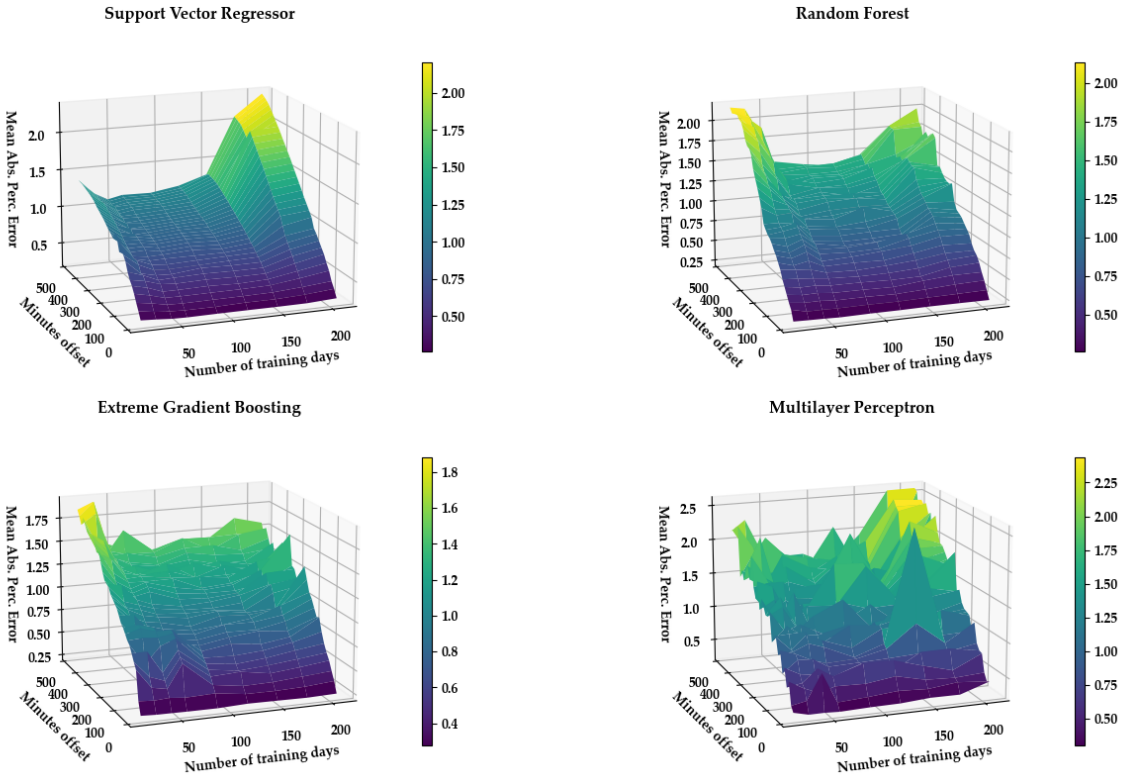


Figure A2. Surface describing the variation of the MAPE with respect to the minutes horizon from the prediction instant (x-axis) and the number of days used to train the model (y-axis) for the various regression models tested. The results refer to the test set.

Table A2. Average of the absolute error mean (MAE) and standard deviation (STDAE), and of the absolute percentage mean (MAPE) and standard deviation (STDAPE) of the predictions obtained with the Random Forest model on the test set grouped by the number of training days.

Training Days	Average MAE	Average STDAE	Average MAPE	Average STDAPE
15	0.37	0.44	1.33	1.58
30	0.35	0.40	1.23	1.42
45	0.32	0.39	1.13	1.35
60	0.27	0.37	0.94	1.25
90	0.27	0.39	0.94	1.28
120	0.27	0.39	0.94	1.29
150	0.28	0.40	0.98	1.33
180	0.33	0.41	1.16	1.41
210	0.35	0.40	1.21	1.38

Table A3. Average of the absolute error mean (MAE) and standard deviation (STDAE), and of the absolute percentage mean (MAPE) and standard deviation (STDAPE) of the predictions obtained with the Extreme Gradient Boosting model on the test set grouped by the number of training days.

Training Days	Average MAE	Average STDAE	Average MAPE	Average STDAPE
15	0.36	0.38	1.27	1.28
30	0.33	0.37	1.16	1.26
45	0.30	0.36	1.03	1.23
60	0.30	0.39	1.04	1.30
90	0.27	0.38	0.95	1.27
120	0.28	0.40	0.96	1.33
150	0.28	0.39	0.96	1.31
180	0.29	0.40	1.01	1.34
210	0.32	0.41	1.11	1.37

Table A4. Average of the absolute error mean (MAE) and standard deviation (STDAE), and of the absolute percentage mean (MAPE) and standard deviation (STDAPE) of the predictions obtained with the Multilayer Perceptron model on the test set grouped by the number of training days.

Training Days	Average MAE	Average STDAE	Average MAPE	Average STDAPE
15	0.40	0.47	1.41	1.66
30	0.38	0.45	1.34	1.59
45	0.35	0.44	1.23	1.48
60	0.34	0.42	1.18	1.44
90	0.24	0.41	1.19	1.39
120	0.27	0.41	1.30	1.40
150	0.27	0.43	1.28	1.44
180	0.42	0.45	1.47	1.54
210	0.47	0.48	1.62	1.63

References

1. European Commission. *2030 Climate Energy Framework*. Available online: https://ec.europa.eu/clima/policies/strategies/2030_en (accessed on 20 October 2020).
2. European Commission. *Limiting Global Climate Change to 2 Degrees Celsius The Way Ahead for 2020 and Beyond: Communication from the Commission to the Council, the European Parliament, the European Economic and and Social Committee and the Committee of the Regions*; Commission of the European Communities (COM), Office for Official Publications of the European Communities: Brussels, Belgium, 2007.
3. Eurostat. *Energy Balance Sheets—2016 Data—2018 Edition*; Publications Office of the European Union: Luxembourg, 2018.
4. Doukas, H.; Patlitzianas, K.D.; Iatropoulos, K.; Psarras, J. Intelligent building energy management system using rule sets. *Build. Environ.* **2007**, *42*, 3562–3569. [CrossRef]
5. Sensharma, N.P.; Woods, J.E.; Goodwin, A.K. Relationships between the indoor environment and productivity: A literature review. In Proceedings of the ASHRAE Winter Meeting, San Francisco, CA, USA, 17–21 January 1998; pp. 17–21.
6. Pérez-Lombard, L.; Ortiz, J.; Pout, C. A review on buildings energy consumption information. *Energy Build.* **2008**, *40*, 394–398. [CrossRef]
7. U.S. Department of Energy. *2011 Buildings Energy Data Book*. Available online: http://web.archive.org/web/20130214012609/http:/buildingsdatabook.eren.doe.gov/default.aspx (accessed on 20 October 2020).
8. Castaldo, V.L.; Pisello, A.L. Uses of dynamic simulation to predict thermal-energy performance of buildings and districts: A review. *Wiley Interdiscip. Rev. Energy Environ.* **2018**, *7*, e269. [CrossRef]
9. Wicaksono, H.; Rogalski, S.; Kusnady, E. Knowledge-based intelligent energy management using building automation system. In Proceedings of the 2010 Conference Proceedings IPEC, Singapore, 27–29 October 2010; pp. 1140–1145. [CrossRef]
10. Zhou, K.; Fu, C.; Yang, S. Big data driven smart energy management: From big data to big insights. *Renew. Sustain. Energy Rev.* **2016**, *56*, 215–225. [CrossRef]
11. Levermore, G.J. *Building Energy Management Systems: Applications to Low-Energy HVAC and Natural Ventilation Control*; E & FN Spon: London, UK, 2000.
12. Shaikh, P.H.; Nor, N.B.M.; Nallagownden, P.; Elamvazuthi, I.; Ibrahim, T. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renew. Sustain. Energy Rev.* **2014**, *34*, 409–429. [CrossRef]
13. Nikolaou, M. Model predictive controllers: A critical synthesis of theory and industrial needs. *Adv. Chem. Eng. (ACES)* **2001**, *26*, 131–204. [CrossRef]
14. Sassanelli, C.; Arriga, T.; Terzi, S. The Evogy Case: Enabling Result-Oriented PSS in the Energy Management of B2B Smart Building Industry through Cyber-Physical Systems. In *Proceedings of the Spring Servitization Conference 2020—Advanced services for Sustainability and Growth*; Copenhagen Business School: Frederiksberg, Denmark, 2020; pp. 258–267.
15. Porter, M.E.; Heppelmann, J.E. How Smart, Connected Products Are Transforming Competition. *Harv. Bus. Rev.* **2014**, 64–89. Available online: https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition (accessed on 20 October 2020).

16. Ashton, K. That 'Internet of Things' Thing. 1999. Available online: https://www.rfidjournal.com/that-internet-of-things-thing (accessed on 20 October 2020).
17. Want, R.; Schilit, B.N.; Jenson, S. Enabling the Internet of Things. *Computer* **2015**, *48*, 28–35. [CrossRef]
18. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
19. Bandyopadhyay, D.; Sen, J. Internet of Things: Applications and Challenges in Technology and Standardization. *Wirel. Pers. Commun.* **2011**, *58*, 49–69. [CrossRef]
20. Goto, S.; Yoshie, O.; Fujimura, S. Internet of things value for mechanical engineers and evolving commercial product lifecycle management system. In Proceedings of the 2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Bali, Indonesia, 4–7 December 2016; pp. 1021–1024. [CrossRef]
21. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
22. Maccarana, Y.; Panza, A.; Maroni, G.; Sarto, L.; Carta, M.F.; Reggiani, S. Comparison of model-based and data-driven approaches for modeling energy and comfort management systems, with a case study. In Proceedings of the 2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), Genova, Italy, 11–14 June 2019; pp. 1–6. [CrossRef]
23. Lou, R.; Hallinan, K.P.; Huang, K.; Reissman, T. Smart Wifi Thermostat-Enabled Thermal Comfort Control in Residences. *Sustainability* **2020**, *12*, 1919. [CrossRef]
24. Huang, K.; Hallinan, K.P.; Lou, R.; Alanezi, A.; Alshatshati, S.; Sun, Q. Self-Learning Algorithm to Predict Indoor Temperature and Cooling Demand from Smart WiFi Thermostat in a Residential Building. *Sustainability* **2020**, *12*, 7110. [CrossRef]
25. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to Forget: Continual Prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef] [PubMed]
26. Zhang, C.; Kuppannagari, S.R.; Kannan, R.; Prasanna, V.K. Building HVAC scheduling using reinforcement learning via neural network based model approximation. In Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys 2019, New York, NY, USA, 13–14 November 2019; pp. 287–296. [CrossRef]
27. Fan, C.; Xiao, F.; Zhao, Y. A short-term building cooling load prediction method using deep learning algorithms. *Appl. Energy* **2017**, *195*. [CrossRef]
28. Asadi, K.; Cater, E.; Misra, D.; Littman, M.L. Towards a Simple Approach to Multi-step Model-based Reinforcement Learning. *arXiv* **2018**, arXiv:1811.00128.
29. Yi, F.; Fu, W.; Liang, H. Model-based reinforcement learning: A survey. In Proceedings of the International Conference on Electronic Business (ICEB), Guilin, China, 2–6 December 2018; pp. 421–429.
30. Asadi, K.; Misra, D.; Littman, M. *{L}ipschitz Continuity in Model-based Reinforcement Learning*; Proceedings of Machine Learning Research (PMLR): Stockholm, Sweden, 2018; Volume 80, pp. 264–273.
31. Schölkopf, B.; Smola, A. *Learning With Kernels*; Schölkopf, B., Mika, S., Burges, C.J.P., Knirsch, K.-R.M., Rätsch, G., Smola, A.J., Eds.; MIT Press: Cambridge, MA, USA, 2001; p. 2000.
32. Hofmann, T.; Schölkopf, B.; Smola, A.J. Kernel Methods in Machine Learning. *Ann. Statist.* **2008**, 36. [CrossRef]
33. Pezzotta, G.; Sassanelli, C.; Pirola, F.; Sala, R.; Rossi, M.; Fotia, S.; Koutoupes, A.; Terzi, S.; Mourtzis, D. The Product Service System Lean Design Methodology (PSSLDM). *J. Manuf. Technol. Manag.* **2018**, *29*, 1270–1295. [CrossRef]
34. Sassanelli, C.; Pezzotta, G.; Pirola, F.; Rossi, M.; Terzi, S. The PSS design GuRu methodology: Guidelines and rules generation to enhance PSS detailed design. *J. Des. Res.* **2019**, *17*, 125. [CrossRef]
35. Nunamaker, J.F., Jr.; Chen, M.; Purdin, T.D. Systems development in information systems research. *J. Manag. Inf. Syst.* **1990**, *7*, 89–106. [CrossRef]
36. Williamson, K. Research Methods for Students, Academics and Professionals: Information Management and Systems. *Libr. Rev.* **2004**, *53*, 193–193. [CrossRef]
37. Zheng, A.; Casari, A. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*; O'Reilly Media: Sebastopol, CA, USA, 2018.

38. Segal, M. *Machine Learning Benchmarks and Random Forest Regression*; Technical Report; Center for Bioinformatics & Molecular Biostatistics, University of California: San Francisco, CA, USA, 2003.
39. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016*. ACM: New York, NY, USA, 2016; [CrossRef]
40. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*. [CrossRef] [PubMed]
41. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.