Expert Systems with Applications 40 (2013) 1061–1068

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Machine learning methods to forecast temperature in buildings

Fernando Mateo[a,*], Juan José Carrasco[a], Abderrahim Sellami[a], Mónica Millán-Giraldo[a,b], Manuel Domínguez[c], Emilio Soria-Olivas[a]

[a] Intelligent Data Analysis Laboratory, E.T.S.E, University of Valencia, Avda Universitat S/N, 46100 Burjassot, Valencia, Spain
[b] Institute of New Imaging Technologies, Department of Computer Languages and Systems, University Jaume I, Av. Sos Baynat S/N, 12071 Castelló de la Plana, Spain
[c] SUPPRESS Research Group, University of León, University Campus de Vegazana, 24007 León, Spain

ARTICLE INFO

Keywords:
Forecasting
Energy efficiency
Machine learning
Time series

ABSTRACT

Efficient management of energy in buildings saves a very important amount of resources (both economic and technological). As a consequence, there is a very active research in this field. One of the keys of energy management is the prediction of the variables that directly affect building energy consumption and personal comfort. Among these variables, one can highlight the temperature in each room of a building. In this work we apply different machine learning techniques along with other classical ones for predicting the temperatures in different rooms. The obtained results demonstrate the validity of these techniques for predicting temperatures and, therefore, for the establishment of optimal policies of energy consumption.

1. Introduction

In recent times, there is an evident trend towards energy efficiency awareness; there are several reasons that motivate this trend. The first one is related to achieving environmental sustainability, due to the need for preserving the limited energy resources on the planet (oil, gas, electricity, etc). In addition, there are economic reasons; every possible step towards saving money is well considered both by private consumers and by companies. Finally, political reasons can be argued; energy efficiency can be seen as a way to mitigate the currency outflow from the national economy, optimize the production and achieve stability. With this goal, governments worldwide have fostered plans, ordinances, resolutions, and strategies that try to raise the concern about energy saving. Moreover, the European Commission has adopted an action plan to reduce energy consumption by 20% by 2020, which would mean saving about 60 billion euro annually (European Commission, 2010). The plan is to improve the energy efficiency of products, buildings and services by promoting efficiency in the production and distribution of energy, reducing the impact of transport on energy consumption, providing the funding and implementation investment in Europe, encouraging and reinforcing rational behaviour regarding energy consumption and also strengthening international action on energy efficiency (European Commission, 2010).

In the recent years, governments are also trying to boost the construction and refurbishing of buildings to make them "energy efficient". Energy consumption in residential and commercial buildings accounts for about 40% of the final energy consumption and 36% of total CO_2 emissions of the European Union according to the International Energy Agency (IEA) (IEA International Energy Agency, 2010). This is the reason why important efforts in applying energy management processes are focused on the building sector, which demonstrates increasing energy intensity and energy consumption indexes (European Commission, 2003, 2005).

In order to keep the comfort in a building while minimising the power consumption, and hence to save energy, it is very important to predict the temperature to try to bring indoor air temperature into a comfortable range.

In the literature it is possible to find comparisons of linear models with ANNs (Artificial Neural Networks) applied to daily temperature profile forecast (Hippert et al., 2004) and indoor temperature forecast in buildings (Mechaqrane & Zouak, 2004; Thomas & Soleimani-Mohseni, 2007), which conclude that neural networks are more advantageous than linear models. In the present work, some emerging methods like Extreme Learning Machines (ELMs), or the combination of Multilayer Perceptron (MLP) ANNs with non-linear autoregressive techniques (NARX) will be introduced and compared with other more traditional models based on linear approximations like multiple linear regression, robust multiple linear regression and autoregressive exogenous models (ARX). Finally, clustering of the data has been taken into account as a possible way to improve the results of the aforementioned methods.

This paper is organized as follows. Section 2 describes the data used in the experiments in detail. Section 3 briefly introduces the machine learning methods that have been evaluated. In Section 4 the results are then compared and interpreted. In the final section, the main conclusions of this work are summarized.

* Corresponding author.
E-mail address: fernando.mateo@uv.es (F. Mateo).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.08.030

2. Data used

The data used for the predictions in this paper has been obtained by simulating a building in TeKton 3D (TeKton 3D, 2011). The building is located in the province of Málaga (Spain), which corresponds to the B3 climatic zone according to Spanish technical code of buildings (CTE) (Pons González & Arco Torres, 2007). Data corresponding to four rooms in the complex have been acquired. The purpose of the four selected rooms is the following:

- Room 1: Thermal zone.
- Room 2: Physiotherapy.
- Room 3: Pottery workshop.
- Room 4: Management.

For the construction of the models, 18 variables were acquired hourly for a period of one year. The variables include indoor/outdoor climatic information and simulated thermal charges. The variables to predict are the temperatures of the four rooms in the next hour (current temperature arrays shifted one hour), all the variables are listed in Table 1.

Table 1
Variables of the used dataset, where $k = \{1,2,3,4\}$ indicates the room number.

Variable	Description
Month	Month
DayW	Day of the week
DayM	Day of the month
Time	Official time
RH	Relative humidity (%)
OT	Outside temperature (°C)
ST_k	Set point temperature of room k (°C)
TP_k	Total thermal power of room k (W)
CT_k	Current temperature of room k (°C)

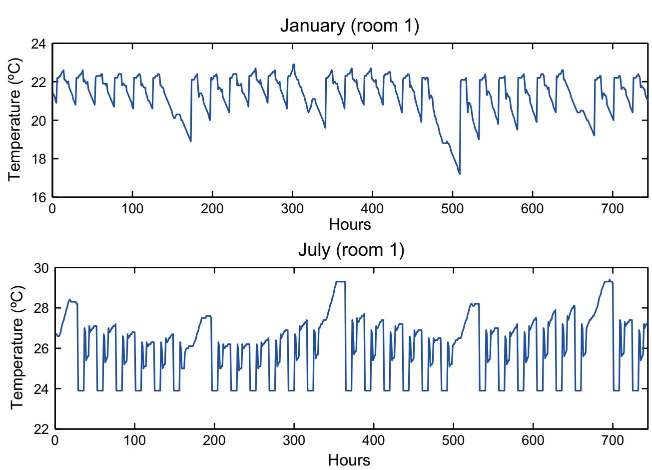

Fig. 1. Example of the temperature record for a winter month (January) and for a summer month (July) for room 1.

As an example, the temperatures obtained by the simulator for room 1 during a winter month (January) and during a summer month (July) have been plotted in Fig. 1. From the plots, it is possible to observe the change of behaviour between different times of the year; and we can even notice daily and weekly periodicities due, respectively, to the interchange between night and day, and between workdays and weekends.

To gain knowledge about the data, the basic statistics of the most relevant variables are shown in Table 2. The negative and positive values of thermal power correspond to heating and cooling modes, respectively. The thermal power due to the cooling mode is greater than the thermal power due to the heating mode as shown in the table. This is frequent in Mediterranean climate, as well as the values of relative humidity and outdoor temperature. Finally, the statistics show that the current indoor temperatures are very similar in the four rooms.

2.1. Data processing

The acquired variables require preprocessing before continuing to the stage of model construction and validation (Makridakis, Wheelwright, & Hyndman, 1998). First, the months and the days of the week were coded as integer numbers (1–12 and 1–7, respectively). Secondly, the official time was converted to a cyclic variable by applying a cosine transformation:

$$\text{Time}' = \cos\left(\frac{2 \cdot \pi \cdot \text{Time}}{24}\right). \tag{1}$$

The setpoint temperatures had always a constant value when set and a null value when not. Therefore, they were coded as 1 when set and 0 when not specified. Finally, the input variables were standardized.

3. Methods used

3.1. Autoregressive model

The formula that defines an AR (p) process, where p is the order, or number of past consecutive quantitative states of the time series to take into account for the prediction, is:

$$y(t) = c + \sum_{i=1}^{p} \varphi_i y(t-i) + \epsilon(t), \tag{2}$$

where c is a constant, φ_i are the parameters of the model and $\epsilon(t)$ is white noise (Makridakis et al., 1998). The AR parameters can be calculated using different methods like least-squares, Yule-Walker or Burg (Kay, 1988).

AR models can be complemented with exogenous inputs $u(t)$ to help the prediction (ARX models) (Makridakis et al., 1998):

$$y(t) = -\sum_{i=1}^{n_a} \varphi_i y(t-i) + \sum_{i=0}^{n_b-1} \psi_i u(t-i-n_k+1) + \epsilon(t), \tag{3}$$

where n_a is the order of the model for the output signal, n_b the order for the input signal and n_k the delay of the input signal.

Table 2
Annual mean, maximum, minimum and standard deviation of relative humidity (RH), outdoor temperature (OT), thermal power (TP) and current temperature (CT) of the 4 rooms.

Statistic	RH	OT	TP_1	TP_2	TP_3	TP_4	CT_1	CT_2	CT_3	CT_4
Mean	64.64	16.77	3829.17	341.69	419.53	158.07	23.64	24.12	23.86	24.10
Min.	24.50	2.40	−48615.00	−1747.00	−5288.00	−1421.00	17.20	17.20	17.20	18.00
Max.	100.00	33.90	65359.00	3080.00	8347.00	2762.00	29.90	30.10	30.70	30.50
Std	15.05	6.35	12209.17	699.87	1499.44	477.81	2.02	1.85	2.45	2.26

AR models can also be modified with moving average terms θ of order q, MA (q), producing autoregressive moving-average ARMA (p,q) models:

$$y(t) = c + \epsilon(t) + \sum_{i=1}^{p} \varphi_i y(t-i) + \sum_{i=1}^{q} \theta_i \epsilon(t-i). \tag{4}$$

3.2. Robust multiple linear regression

Multiple Linear Regression (MLR) fits a data model that is linear in the model coefficients, which are typically estimated by least-squares, and can approximate both lines and polynomials, among other linear models (Weisberg, 1985). A polynomial fit for an input of dimensionality p and n observations is given by the equation

$$y_i = b_0 + \sum_{k=1}^{p} b_k X_{(i,k)} + \epsilon_i, i = 1, \ldots, n, \tag{5}$$

where b_k are the regression coefficients, $X_{n\times p}$ are the input data, y_i is i-th sample of the output variable and ϵ is the error vector, which is unknown but can be estimated by the residuals (differences between the predicted and observed values of the output variables):

$$\hat{\epsilon}_i = \hat{y}_i - y_i. \tag{6}$$

Producing a fit using a linear model requires minimizing the sum of the squares of the residuals. The robust version of the regression function (RMLR) uses an iteratively reweighted least-squares algorithm, with the weights at each iteration calculated by applying the bisquare function to the residuals from the previous iteration (Holland & Welsch, 1977). This algorithm gives lower weight to points that do not fit well. The results are less sensitive to outliers in the data as compared with ordinary least-squares regression.

3.3. Multilayer Perceptron

Multilayer Perceptron (MLP) is the most widely used neural network model and has been applied to almost all scientific fields, thanks to a robust non-linear approximation behaviour (Haykin, 2008). An MLP produces a non-linear mapping of a set of input data to one or more outputs in an adaptive way, which is accomplished by learning from examples. Typically, the learning is done by the *Backpropagation* algorithm or one of its variants.

3.4. Extreme Learning Machine

Let us consider a set of N points $(\mathbf{x}_i, \mathbf{t}_i) \in \mathbb{R}^n \times \mathbb{R}^m$, where $i = 1,\ldots,N$. A standard single-hidden-layer feedforward network (SLFN) with L hidden neurons and activation function g(x) can be mathematically modeled as

$$\sum_{i=1}^{L} \beta_i g(\mathbf{w}_i \mathbf{x}_i + b_i) = \mathbf{d}_j, \quad j = 1, \ldots, N, \tag{7}$$

where $\mathbf{w}_i$ is the weight vector connecting the inputs to the i-th hidden neuron, β_i is the weight vector connecting the i-th hidden neuron and output neurons, b_i is the threshold of the i-th hidden neuron, and $\mathbf{d}_j$ is the output given by the ELM for data point j.

Defining

$$\mathbf{H} = \begin{pmatrix} g(\mathbf{w}_1\mathbf{x}_1 + b_1) & \ldots & g(\mathbf{w}_L\mathbf{x}_1 + b_L) \\ \vdots & \ldots & \vdots \\ g(\mathbf{w}_1\mathbf{x}_N + b_1) & \ldots & g(\mathbf{w}_L\mathbf{x}_N + b_L) \end{pmatrix}_{N\times L}$$

and

$$\beta = \begin{pmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{pmatrix}_{L\times m} \quad \text{and} \quad \mathbf{T} = \begin{pmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{pmatrix}_{N\times m},$$

the solution of the ELM is

$$\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T}, \tag{8}$$

where $\mathbf{H}^{\dagger} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}$ is called Moore–Penrose generalized inverse of $\mathbf{H}$ (Huang, Zhu, & Siew, 2006).

The ELM algorithm for SLFNs can be summarized in these steps (Huang et al., 2006):

1. Assign arbitrary input weights $\mathbf{w}_i$ and biases b_i, $i = 1,\ldots,L$.
2. Calculate the hidden layer output matrix $\mathbf{H}$.
3. Calculate the output weights β:
   $\beta = \mathbf{H}^{\dagger}\mathbf{T}$ where $\mathbf{H}$, β and $\mathbf{T}$ are as defined before.

3.5. Non-linear Autoregressive Exogenous Multilayer Perceptron

Non-linear Autoregressive Exogenous (NARX) models are commonly used for prediction of time series by approximating non-linear relationships between exogenous variables and the variable to predict, according to the Equation (Lentaritidis & Billings, 1985a, 1985b):

$$\begin{aligned} y(t) = f(&x(t-1), x(t-2)\ldots x(t-d); \\ &y(t-1), y(t-2), \ldots y(t-d)) \end{aligned} \tag{9}$$

where $y(t)$ is the variable to predict at and $x(t)$ is an external or exogenous time series.

Often, the non-linear function f is simply a polynomial. It is also possible to create a dynamic MLP network to model the NARX function f assuming that, at instant t, the d past values of the variable to predict and the exogenous variable are available. This can be implemented in practice by applying a delay of d samples to both exogenous variables, as it can be observed in Fig. 2. This configuration based on delays without feedback (open loop) is useful to make one-step-ahead predictions, because the estimations are based on knowledge about the actual past values of the target series, and not on predictions, which would introduce errors in the result.

3.6. Clustering

The use of clustering methods often provides information with regard to the underlying data structure, that is initially hard to detect. It helps to arrange and combine samples in the initial population according to their similarities, remove outliers and provide more homogeneous datasets. Thus, applying data clustering before building a model is often a helpful technique that can improve the performance.

In this work, five clustering algorithms have been compared and will be combined with both multivariate linear regression models and non-linear neural network-based models (ELM and MLP). In addition, the results will be compared with those obtained without performing clustering. The compared algorithms are:

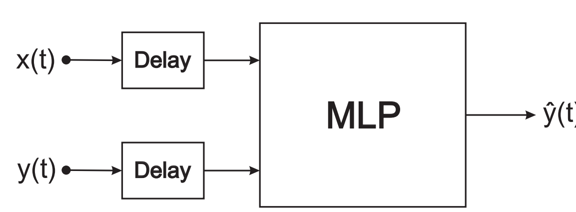

Fig. 2. Dynamic MLP models for NARX implementation in open loop configuration.

1. *K-means*: It is probably the most common clustering algorithm (MacQueen, 1967). The algorithm is based on minimizing the internal distance (the sum of the distances of the patterns assigned to a cluster to the centroid of that cluster).
2. *Fuzzy c-means*: This clustering algorithm is a variant of k-means clustering algorithm. The difference is that, in the latter, each data belongs to exactly one group while fuzzy c-means allows partial membership to several groups with some degree of membership (Dunn, 1973). Another difference is the method to update the centroids.
3. *Cumulative Hierarchical Tree*: The distance between two clusters is computed as the minimum distance between them, i.e. the distance between the two closest elements of the two clusters. This is basically the single linkage algorithm by Kruskal (1956) for graph theory.
4. *DBSCAN*: A clustering algorithm based on the density, or neighbourhood with different numbers of points. It also introduces the noise component in the analysis. The clusters are determined by grouping the points that have in their neighbourhood a number of points greater than or equal to a specified threshold (Ester, Kriegel, Sander, & Xu, 1996).
5. *K-medoids*: The algorithm searches for *k* representative objects, called medoids, that are centred on the clusters that they define (Theodoridis & Koutroumbas, 2006). The medoid, i.e. the object that represents the conglomerate, is the object for which the average dissimilarity to all objects in the cluster is minimal. Indeed, the distribution algorithm of points around medoids (PAM) minimizes the sum of dissimilarities instead of the average dissimilarity.

4. Results

First, a variable relevance analysis using *RReliefF* (Robnik-Šikonja & Kononenko, 2003) will be performed and the following subsections will analyse the predicting capabilities of the different methods, without and with clustering.

4.1. Variable relevance analysis

In order to improve interpretability and knowledge about the relationships between inputs and outputs, the 18 variables have been ranked by the *RReliefF* method, according to their relevance to predict each output (Robnik-Šikonja & Kononenko, 2003; Kira & Rendell, 1992; Kononenko, 1994).

The analysis evaluated, firstly, the value of each attribute by the *RReliefF* method, repeatedly sampling each of the instances and comparing with the same attribute value obtained using the nearest neighbour in the same class and in a different one. The results, using the same notation of Table 1, are listed in Table 3.

Table 3
Input variable relevance ranking by *RReliefF* for each output variable.

Output variable	PT_1	PT_2	PT_3	PT_4
Input variable ranking	Time	Time	CT_3	CT_4
	CT_1	CT_2	Time	Month
	CT_3	OT	CT_4	CT_3
	CT_4	CT_1	CT_1	CT_1
	CT_2	CT_3	Month	DayM
	OT	Month	CT_2	OT
	Month	CT_4	OT	RH
	RH	RH	DayM	CT_2
	ST_1	DayM	RH	Time
	TP_3	ST_1	ST_1	ST_1
	TP_1	TP_3	TP_3	TP_3
	ST_3	TP_1	TP_1	TP_1
	TP_4	ST_3	ST_3	ST_3
	DayM	ST_2	TP_4	TP_4
	TP_2	TP_2	TP_2	TP_2
	ST_2	TP_4	ST_2	ST_2
	ST_4	ST_4	ST_4	ST_4
	DayW	DayW	DayW	DayW

As it seems logical, for each temperature to predict, the temperatures of the other rooms are always at the top of the list because they are highly correlated. The official time is also selected in the first positions, except for room 4 (ninth place). The month of the year also tends to appear with a high relevance value in the ranking. The outside temperature usually falls around the sixth or seventh place in the list, except for room 2, for which it appears in the third place. This indicates that the temperature of this room, either because of a different orientation or isolation, is more affected by outdoor climatic conditions than the rest. For every output variable, the attributes 2 (day of the week) and 10 (thermal power of room 2) are selected as the least relevant.

4.2. Predictions without clustering

All the methods were systematically evaluated on the dataset introduced in Section 2 and the Mean Absolute Error (MAE) on the validation set was calculated as an average of 20 simulations, to make the results independent of the initial choice of parameters for the non-linear methods. The autoregressive methods were built monthly and one-step-ahead prediction was used. The rest of the models were trained using yearly data, using the first 20 days of each month for training and the rest for validation. The individual MAE for each month was obtained for every model, as well as the yearly error average.

The linear regression methods (MLR and RMLR) have been evaluated and compared. Thanks to the improved way of weight calculation, RMLR is more robust against outliers, as expected, and always improves MLR results. Therefore, the latter has been omitted from the results.

With regard to the ELM, the number of sigmoid hidden nodes was evaluated from 10 to 150 in steps of 10 nodes. The best architecture in terms of annual MAE (12 month average) for room 1 was 100 hidden nodes, which produced a MAE of 0.56 °C.

Initially, the values of MAE obtained with ELM using the full dataset were disappointing because they were much higher than those obtained with linear models. The fact of building separate models for each month did not help either. To tackle this issue, a selection of attributes was performed using the *RReliefF* ranking obtained in Section 4.1. In complex multidimensional problems like these, the fact of including irrelevant variables in the analysis can lead to a decrease in performance. Thus, we kept the 10 most significant variables and repeated the tests for room 1. The performance gain was substantial as it can be observed in Fig. 3. The average monthly improvement in MAE when using variable selection is 53%. However, the fact of performing variable selection hardly influences the linear models' error. For the MLR models, the mean annual error increases in 0.8%, while for the RMLR models the average improvement is only 1.6%.

To assess the autoregressive model performance we used a regressor of 24 hours to capture daily periodicity data. Monthly models with orders ranging from 1 to 12 were evaluated and the one which produced the lowest Akaike Information Criterion (AIC) was chosen. The best performing among the linear autoregressive methods was ARX, using the most correlated room temperature with the desired one as input. Therefore, other methods such as AR or ARMA have been omitted from the results.

For the MLP NARX implementation, separate open-loop NARX models were built for each month, using the most correlated room temperature with the desired one as exogenous variable and a lag of 24 h. The MLPs for each monthly NARX model were given one hidden layer composed by 20 nodes with sigmoid activation functions.

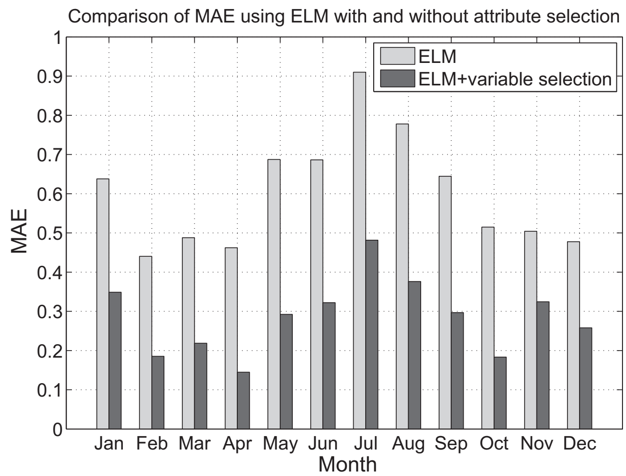

Fig. 3. Comparison of monthly performance of ELM models for room 1 before and after applying *RReliefF* ranking and selecting the 10 most significant variables.

Table 4
Monthly MAE obtained without clustering. RMLR: Robust Multiple Linear Regression, ELM: Extreme Learning Machine, MLP: Multilayer Perceptron, ARX: Autoregressive Exogenous, MLP-NARX: Nonlinear Autoregressive Exogenous Multilayer Perceptron. The best results are highlighted in bold.

Model	RMLR	ELM	MLP	ARX	MLP-NARX
Jan.	0.181	0.349	0.280	0.239	**0.131**
Feb.	**0.123**	0.185	0.181	0.132	0.129
Mar.	0.130	0.219	0.190	0.156	**0.101**
Apr.	0.115	0.145	0.197	0.110	**0.057**
May	0.237	0.292	0.263	0.206	**0.090**
Jun.	0.291	0.322	0.270	0.164	**0.091**
Jul.	0.375	0.482	0.361	0.225	**0.170**
Aug.	0.344	0.376	0.306	0.156	**0.144**
Sep.	0.241	0.297	0.257	0.144	**0.134**
Oct.	0.154	0.183	0.162	0.196	**0.071**
Nov.	0.140	0.324	0.208	0.151	**0.069**
Dec.	0.148	0.258	0.234	0.151	**0.141**
Mean	0.207	0.286	0.242	0.171	**0.111**

The monthly mean absolute errors (MAE) attained for room 1 with the methods proposed in Section 3, without clustering, are summarized in Table 4. The rest of the rooms' behaviour is similar so they have been omitted. From the inspection of the results, one can easily deduce that the MLP and NARX combination yields the best results in terms of MAE and also the standard deviations are the lowest among all the methods. As a consequence, this is not only the most accurate method among the tested, but also the most precise. The MLP NARX predictions for the validation set are represented graphically for a winter month (January) and for a summer month (July) in Fig. 4.

It is also important to highlight that autoregressive-based methods do not need to acquire data for one year but only during the necessary time to build a lag vector, e.g. 24 consecutive hours, to start predicting. As a result, in a practical scenario, they are more convenient than the methods that need the acquisition of the full dataset.

Table 4 lists the results obtained without using clustering.

4.3. Predictions with clustering

Clustering was tested in order to rearrange the data points according to their similarities and independent models were used for each cluster.

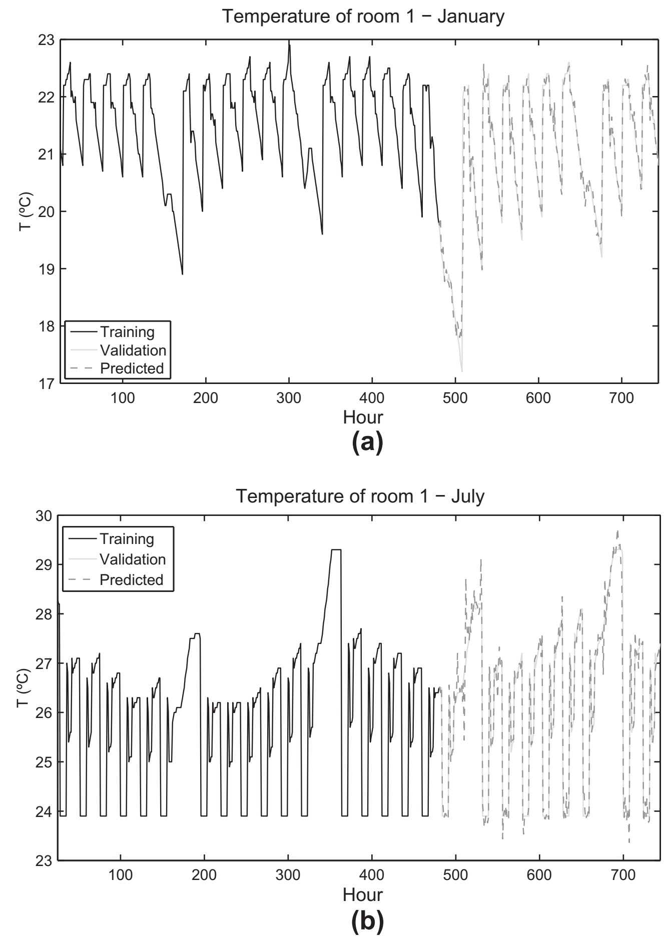

Fig. 4. Prediction of the temperature of room 1 with MLP-NARX models for (a) January and (b) July.

The results obtained when using clustering are presented in Tables 5–9. The number of clusters evaluated for each method ranges from 2 to 5, and three types of models have been built and compared (RMLR, ELM and MLP), based on the same clustered data. The experiments were replicated 20 times and errors were averaged to obtain realiable estimations. Autoregressive methods have been omitted from the analysis because clustering defeats the purpose of the continuous sliding window necessary to build these models.

The results show a clear advantage for the RMLR method against its nonlinear counterparts. This behaviour could be foreseen from the results without clustering (Table 4). The only case in which RMLR does not perform well is during the warmest months of the year (May to September) when using 3-means clustering. This may be attributed to particular difficulties found to split the data of these months in 3 clusters due to the existence of long constant temperature spans when setpoint temperatures are established. K-means naturally tries to make equally sized clusters, which in this case might not be the best solution. In any case, when using clustering, the improvement is minimum and in some cases it deteriorates the estimation, especially for ELM and MLP.

The best MAE was 0.203 °C and was obtained by RMLR using DBSCAN clustering. This seems to be a lower bound that is achieved independently from number of clusters. In any case, all clustering methods give a similar performance with RMLR with any number of clusters. Thus, there is a marginal improvement (~1%) when using RMLR and clustering as compared to the results without clustering. In some isolated cases, e.g. *k*-means with 3 clusters, RMLR yields a poor result because of a bad performance

Table 5
Monthly MAE (°C) obtained with K-means clustering.

	Clusters											
	2			3			4			5		
	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP
Jan.	0.182	0.341	0.241	0.181	0.320	0.254	**0.183**	0.344	0.289	0.185	0.368	0.233
Feb.	0.123	0.177	0.162	0.121	0.180	0.158	0.121	0.185	0.153	**0.120**	0.182	0.159
Mar.	**0.133**	0.210	0.162	0.134	0.202	0.164	0.134	0.215	0.150	0.134	0.224	0.160
Apr.	0.116	0.135	0.178	0.114	0.136	0.180	0.114	0.138	0.127	**0.113**	0.146	0.190
May	0.236	0.356	0.331	0.311	0.345	0.366	0.233	0.324	0.277	**0.232**	0.319	0.416
Jun.	0.287	0.336	0.336	1.588	0.472	0.428	**0.286**	0.710	0.401	**0.286**	1.121	0.490
Jul.	0.369	0.464	0.439	2.606	0.474	0.548	**0.367**	0.468	0.494	**0.367**	3.869	0.688
Aug.	0.339	0.357	**0.338**	3.188	0.381	0.439	**0.338**	0.402	0.419	**0.338**	0.431	0.543
Sep.	0.238	0.317	0.266	0.580	0.309	0.278	**0.237**	0.299	0.307	0.238	0.301	0.345
Oct.	0.153	0.168	0.152	0.153	0.166	0.163	0.152	0.157	**0.142**	0.152	0.159	0.197
Nov.	0.140	0.271	0.160	**0.139**	0.299	0.174	**0.139**	0.292	0.152	**0.139**	0.313	0.161
Dec.	0.149	0.244	0.208	**0.148**	0.236	0.207	0.149	0.248	0.193	**0.148**	0.252	0.197
Mean	0.205	0.281	0.248	0.961	0.293	0.280	0.205	0.315	0.279	**0.204**	0.640	0.315

Table 6
Monthly MAE (°C) obtained with fuzzy c-means clustering.

	Clusters											
	2			3			4			5		
	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP
Jan.	**0.183**	0.335	0.241	0.184	0.329	0.246	**0.183**	0.339	0.251	0.186	0.320	0.240
Feb.	0.123	0.178	0.161	0.123	0.179	0.160	0.121	0.180	0.157	**0.120**	0.172	0.152
Mar.	**0.133**	0.205	0.163	0.134	0.217	0.173	0.135	0.202	0.166	0.134	0.223	0.173
Apr.	0.116	0.132	0.177	0.114	0.141	0.172	0.114	0.139	0.174	**0.112**	0.148	0.173
May	0.235	0.372	0.400	0.233	0.385	0.327	0.234	0.281	0.376	**0.231**	0.336	0.329
Jun.	**0.286**	0.356	0.503	0.352	0.676	0.502	**0.286**	1.043	0.473	**0.286**	1.178	0.396
Jul.	0.368	0.456	0.537	0.554	0.489	0.704	0.367	0.479	0.638	**0.366**	0.490	0.626
Aug.	**0.338**	0.345	0.454	0.576	0.458	0.562	**0.338**	0.488	0.470	**0.338**	0.458	0.534
Sep.	0.238	0.324	0.285	0.243	0.284	0.262	**0.237**	0.284	0.268	0.239	0.310	0.273
Oct.	0.153	0.162	0.153	0.152	0.157	0.157	0.152	0.158	0.158	0.152	**0.151**	0.154
Nov.	0.140	0.264	0.156	0.140	0.295	0.168	0.139	0.288	0.170	**0.139**	0.321	0.161
Dec.	**0.149**	0.244	0.207	**0.149**	0.242	0.209	**0.149**	0.236	0.209	**0.149**	0.247	0.203
Mean	0.205	0.281	0.286	0.246	0.321	0.303	0.205	0.343	0.292	**0.204**	0.363	0.285

Table 7
Monthly MAE (°C) obtained with hierarchical tree clustering.

	Clusters											
	2			3			4			5		
	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP
Jan.	**0.184**	0.328	0.234	**0.184**	0.341	0.272	0.184	0.320	0.242	**0.184**	0.355	0.221
Feb.	0.123	0.182	0.154	0.123	0.187	0.148	**0.122**	0.174	0.151	**0.122**	0.180	0.151
Mar.	0.135	0.231	0.173	**0.134**	0.234	0.162	**0.134**	0.235	0.166	0.135	0.242	0.163
Apr.	0.114	0.157	0.186	0.114	0.151	0.173	**0.113**	0.148	0.183	0.114	0.146	0.166
May	**0.230**	0.395	0.461	0.232	0.619	0.341	0.231	0.571	0.401	0.233	0.703	0.366
Jun.	**0.285**	0.322	0.397	0.287	1.189	0.543	0.286	1.177	0.452	0.286	1.518	0.487
Jul.	**0.366**	0.454	0.474	0.367	0.482	0.730	0.367	0.458	0.567	0.367	0.473	0.526
Aug.	0.338	**0.335**	0.370	0.338	0.464	0.580	0.338	0.454	0.450	0.338	0.430	0.416
Sep.	**0.234**	0.282	0.269	0.236	0.286	0.251	0.236	0.299	0.251	0.239	0.343	0.282
Oct.	0.154	0.184	0.169	0.152	0.151	0.159	0.152	0.153	0.165	0.152	0.159	**0.150**
Nov.	0.140	0.306	0.162	0.140	0.263	0.166	**0.139**	0.302	0.162	**0.139**	0.288	0.159
Dec.	**0.149**	0.256	0.216	**0.149**	0.235	0.208	**0.149**	0.249	0.204	**0.149**	0.248	0.195
Mean	**0.204**	0.286	0.272	0.205	0.384	0.311	**0.204**	0.378	0.283	0.205	0.424	0.273

in the months from June to September that makes the annual MAE rise to 0.961 °C.

The nonlinear methods (ELM and MLP) do not benefit when performing clustering in general. The values of MAE remain close to the ones without clustering when k is low. When k increases, the error rises too. The scenario where the values of MAE improve the most is when using DBSCAN clustering with $k = 2$. In this case the annual MAE of ELM decreases from 0.286 °C to 0.266 °C and for MLP it decreases from 0.242 °C to 0.229 °C. These are also marginal improvements (∼1%).

Table 8
Monthly MAE (°C) obtained with DBSCAN clustering.

	Clusters											
	2			3			4			5		
	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP
Jan.	0.201	0.345	0.265	**0.200**	0.361	0.293	0.201	0.369	0.273	0.201	0.371	0.236
Feb.	0.119	0.176	0.166	**0.118**	0.204	0.174	0.119	0.227	0.168	0.119	0.207	0.160
Mar.	**0.134**	0.200	0.166	**0.134**	0.238	0.174	**0.134**	0.244	0.187	**0.134**	0.239	0.169
Apr.	**0.102**	0.134	0.193	**0.102**	0.148	0.199	**0.102**	0.172	0.205	**0.102**	0.166	0.173
May	**0.226**	0.272	0.286	**0.226**	0.271	0.294	**0.226**	0.264	0.357	**0.226**	0.280	0.341
Jun.	0.284	0.289	0.270	0.283	0.297	**0.267**	0.283	0.289	0.316	0.283	0.299	0.364
Jul.	0.364	0.436	**0.342**	0.364	0.454	0.348	0.364	0.439	0.409	0.364	0.440	0.437
Aug.	0.336	0.332	0.286	0.336	0.337	**0.285**	0.336	0.332	0.319	0.336	0.334	0.369
Sep.	0.241	0.256	0.238	0.241	0.253	0.233	0.241	0.254	**0.227**	0.241	0.255	0.264
Oct.	**0.143**	0.158	0.152	**0.143**	0.256	0.156	0.144	0.222	0.149	0.144	0.236	0.156
Nov.	**0.140**	0.334	0.178	**0.140**	0.369	0.182	**0.140**	0.431	0.174	**0.140**	0.417	0.159
Dec.	0.153	0.266	0.212	**0.152**	0.264	0.223	**0.152**	0.296	0.216	**0.152**	0.280	0.207
Mean	**0.203**	0.266	0.229	**0.203**	0.288	0.236	**0.203**	0.295	0.250	**0.203**	0.294	0.253

Table 9
Monthly MAE (°C) obtained with K-medoids clustering.

	Clusters											
	2			3			4			5		
	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP	RMLR	ELM	MLP
Jan.	0.182	0.347	0.280	**0.181**	0.345	0.268	0.183	0.338	0.251	0.184	0.360	0.243
Feb.	0.123	0.184	0.166	0.122	0.182	0.156	0.121	0.168	0.166	**0.120**	0.184	0.154
Mar.	**0.133**	0.211	0.170	0.134	0.218	0.164	0.134	0.198	0.164	0.134	0.237	0.167
Apr.	0.116	0.139	0.176	0.114	0.140	0.187	0.114	0.139	0.178	**0.113**	0.139	0.173
May	0.236	0.379	0.358	0.233	0.347	0.378	0.233	0.351	0.380	**0.232**	0.294	0.336
Jun.	0.288	0.342	0.430	0.287	0.558	0.413	**0.286**	0.430	0.489	0.373	1.122	0.493
Jul.	**0.370**	0.458	0.510	0.374	0.461	0.477	0.367	0.461	0.678	0.553	0.489	0.691
Aug.	0.340	0.344	0.398	0.341	0.402	0.380	**0.338**	0.384	0.489	0.576	0.432	0.535
Sep.	0.238	0.338	0.271	**0.236**	0.298	0.272	0.237	0.309	0.279	0.267	0.295	0.273
Oct.	0.153	0.171	0.158	0.153	0.170	0.166	**0.152**	0.170	0.161	**0.152**	0.159	0.163
Nov.	0.140	0.273	0.169	**0.139**	0.317	0.160	**0.139**	0.283	0.168	**0.139**	0.293	0.165
Dec.	0.149	0.249	0.214	0.148	0.255	0.214	0.149	0.245	0.199	0.148	0.245	0.198
Mean	**0.205**	0.286	0.275	**0.205**	0.308	0.270	**0.205**	0.290	0.300	0.249	0.354	0.299

5. Conclusions

Indoor temperature prediction is a crucial issue for optimizing the energy management systems that are installed in large buildings. In this article, several linear and non-linear methods have been compared to predict the indoor temperature using simulated building data.

The evaluated linear methods have been Robust Multiple Linear Regression (RMLR) and linear autoregressive methods. ARX was selected as the best performing among the autoregressive methods compared, e.g. AR and ARMA. Their orders were selected according to the minimum Akaike Information Criterion (AIC). The non-linear machine learning techniques used were ELM and MLP-NARX. Finally, clustering was assessed as a possible way to improve the results. Five different clustering methods were compared using up to 5 clusters. The following conclusions can be extracted:

1. The best overall performing method has been MLP-NARX. This method adds the benefit of using an autoregressive approach to build the model, which keeps trace of the daily periodicity with the non-linear prediction capabilities of MLPs. The average annual MAE obtained was 0.11 °C.
2. ARX was the second best performer, with an average annual MAE of 0.17 °C.
3. ELM results were initially disappointing but greatly improved by using variables ranking by the *RReliefF* method and selecting the 10 first variables, leading to errors in the same order of the ones found with linear methods. The rest of the methods did not show any noticeable performance improvement with variable selection.
4. Clustering did not show a clear improvement for either linear or non-linear methods. DBSCAN proved to be the best clustering algorithm with $k = 2$ clusters for all methods, but only a marginal MAE improvement was accomplished ($\sim$1%). In general, MAE increases for the non-linear methods tested when adding clusters, while it remains approximately constant for RMLR.

To sum up, several machine learning methods and clustering tecniques have been compared to deal with the problem of short-term temperature forecasting in a building, yielding good results, especially when combining a NARX process with a MLP, which lead to an average error of nearly 0.1 °C.

Acknowledgements

This work was financed by the Spanish *Ministerio de Economía y Competitividad* (MINECO) and FEDER funds in the framework of the project EFIS: *Un Sistema Inteligente Adaptativo para la Gestión Eficiente de la Energía en Grandes Edificios*, with reference IPT-2011-0962-920000. In addition, we would like to thank Grupotec Servicios Avanzados S.A. for the provided data.

References

Dunn, J. C. (1973). A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics, 3*(3), 32–57.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, & U. M. Fayyad (Eds.), *Proceedings of the second international conference on knowledge discovery and data mining (KDD-96)* (pp. 226–231). AAAI Press.

European Commission. (2003). *European Commission Directorate – General for energy and transport. European Energy and Transport Trends to 2030*, Brussels.

European Commission. (2010). *Energy 2020: A strategy for competitive, sustainable and secure energy. COM(2010) 2020*. Brussels.

Eurostat. (2005). *Energy balance sheets. Data 2002–2003*. Luxemburg: Eurostat.

Haykin, S. (2008). *Neural networks and learning machines* (3rd ed.). Prentice Hall.

Hippert, H., & Pedreira, C. (2004). Estimating temperature profiles for short-term load forecasting: Neural networks compared to linear models. *IEE Proceedings Generation, Transmission and Distribution, 151* (4) 543–547.

Holland, P. W., & Welsch, R. E. (1977). Robust regression using iteratively reweighted least-squares. *Communications in statistics: Theory and methods, A6*, 813–827.

Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme learning machine: A new learning scheme of feedforward networks. *Neurocomputing, 70*, 489–501.

IEA international energy agency. (2010). *Policy pathway – energy performance certification of buildings*, <http://www.iea.org/papers/pathways/buildings_certification.pdf>, 2010.

Kay, S. (1988). *Modern spectral estimation: Theory and application*. Englewood Cliffs, NJ: Prentice Hall.

Kira, K., & Rendell, L. A. (1992). A practical approach to feature selection. In *Proceedings of the ninth international workshop on machine learning, ML '92* (pp. 29–256). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc..

Kononenko, I. (1994). Estimating attributes: Analysis and extensions of RELIEF. In *Proceedings of the European conference on machine learning on machine learning, ECML-94*. Secaucus, NJ, USA: Springer-Verlag New York,Inc., pp. 171-182.

Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American mathematical society, 7*(1), 48–50.

Lentaritidis, I., & Billings, S. (1985a). Input–output parametric models for non-linear systems. Part I: Deterministic non-linear systems. *International Journal of Control, 41*, 303–328.

Lentaritidis, I., & Billings, S. (1985b). Input-output parametric models for non-linear systems. Part I: Stochastic non-linear systems. *International Journal of Control, 41*, 329–344.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In L. M. L. Cam & J. Neyman (Eds.). *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, pp. 281–297). University of California Press.

Makridakis, S. G., Wheelwright, S. C., & Hyndman, R. J. (1998). *Forecasting: Methods and applications* (3rd ed.). John Wiley & Sons.

Mechaqrane, A., & Zouak, M. (2004). A comparison of linear and neural network ARX models applied to a prediction of the indoor temperature of a building. *Neural Computing & Applications, 13*(1), 32–37.

Pons González, M., d. Arco Torres, M.A. (2007). El código técnico de la edificación, Albolote (Granada) Comares, Biblioteca Comares de ciencia jurídica. Urbanismo 22.

Robnik-Šikonja, M., & Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RReliefF. *Machine learning, 53*(1–2), 23–69.

TeKton 3D, Procedimientos-Uno, S.L. (2011). <http://www.nemetschek.es/tekton3d/>.

Theodoridis, S., & Koutroumbas, K. (2006). *Pattern recognition* (3rd ed.). 0123695317. Orlando, FL, USA: Academic Press, Inc..

Thomas, B., & Soleimani-Mohseni, M. (2007). Artificial neural network models for indoor temperature prediction: Investigations in two buildings. *Neural Computing & Applications, 16*, 81–89.

Weisberg, S. (1985). *Applied linear regression* (2nd ed.). New York: John Wiley.