IET Generation, Transmission & Distribution

Research Article

Machine learning based disaggregation of air-conditioning loads using smart meter data

ISSN 1751-8687
Received on 13th April 2020
Revised 7th July 2020
Accepted on 20th July 2020
doi: 10.1049/iet-gtd.2020.0698
www.ietdl.org

Behzad Najafi[1] ✉, Luca Di Narzo[1], Fabio Rinaldi[1], Reza Arghandeh[2]

[1]Department of Energy, Politecnico di Milano, Via Lambruschini 4, Milano 20156, Italy

[2]Department of Computer Science, Electrical Engineering, Mathematical Sciences, Western Norway University of Applied Sciences, Bergen 5063, Norway

✉ *E-mail: behzad.najafi@polimi.it*

Abstract: This study proposes a novel machine learning-based methodology to estimate the air-conditioning (AC) load from the hourly smart meter data. The commonly employed approaches for disaggregating the share of the AC load from the total consumption are either using data obtained from dedicated sensors or high-frequency data that cannot be provided by conventional smart meters. In the present work, an alternative approach is proposed, in which a machine learning-based pipeline is first optimised and trained using the data obtained from a set of households in a period, including both smart meter data and the AC load measurements along with corresponding weather conditions. The obtained optimal pipeline is then utilised to estimate the AC load in another set of buildings in the same period of the year, while providing it with only the smart meter data and weather conditions. As the first step of the pipeline implementation, several features are extracted from the smart meters and weather data. The most promising algorithm is then determined, the corresponding hyper-parameters are optimised and the most influential parameters are finally determined. The proposed method leads to a lower monitoring system's cost, lower user privacy concerns and fewer data processing complexity compared to the conventional energy disaggregation approaches, while providing an acceptable accuracy.

1 Introduction

Cooling energy consumption in buildings has increased from 3.6 to 7 EJ in the period between 2000 and 2017, and it has accordingly become the most rapidly growing energy use in the buildings. It is also estimated that the corresponding energy use would more than double between 2017 and 2040 due to the population and economic growth and increase in the utilisation of air conditioners [1]. Thus, air-conditioning (AC) energy use will notably impact not only the global energy end-use but also the local electrical grids. Furthermore, AC load is one of the main thermostatically controlled loads (TCLs) that can be integrated in demand side management (DSM) programs [2–4], which permits flattening the demand curve and following the generation pattern [5]. TCLs provide an elevated flexibility for demand control owing to the corresponding thermal initial (and can thus be considered as distributed energy storage components [6]), although they are restricted by the constraint of having a low impact on the user's thermal comfort [7].

In this context, determining the AC load of households at each time interval facilitates assessing the corresponding DSM potential and implementing the related strategies. Furthermore, the latter estimation also leads to additional advantages including diagnosing malfunctioning AC units, improved prediction [8–10] of sudden changes in the demand and providing users with recommendations to maximise residential PV generation's self-consumption. Furthermore, it results in raising awareness on energy consumption, which can already lead to a reduction in energy consumption by up to 10% [11]. Owing to frequent deployment and utilisation of smart meters in many countries [12], disaggregating [13–15] the AC load from smart meter data is a promising alternative [16] to commonly employed methods.

As a general terminology, estimation of the electricity consumption of individual appliances from the total load of a household is called energy disaggregation [17, 18] or non-intrusive load monitoring (NILM), which has been a topic of discussion since the 1990s [18, 19]. Several studies, employing different types and granularity of measurements, have been conducted in this field. However, most of the methodologies employed in these studies require the consumption data from dedicated sensors installed on individual appliances.

In another methodology, called unsupervised NILM, the appliance-by-appliance data is not needed and other extracted data including power waveform is instead utilised to detect appliances [20]. The factorial hidden Markov model [16] is commonly utilised in the studies conducted in this field. Nevertheless, most of the proposed approaches in this area need measurements with a high frequency that cannot be supplied by a conventional smart meter [20], in which the data is commonly provided hourly or at the best 15 min time intervals [21]. However, as demonstrated by Perez *et al.* [22] indeed, a granularity lower than 15 min gives poor results.

In this paper, an alternative approach is presented, in which a machine learning based pipeline is first optimised and trained using the data obtained from a set of households including both smart meter data and the disaggregated AC load measurements along with corresponding weather conditions. The obtained optimal pipeline is then utilised to estimate the AC load of another set of buildings, while providing it with only the smart meter data and weather conditions, and the obtained estimation accuracy is determined.

As the first step of the pipeline implementation, several features including seasonality related,lagged, statistics based, regression based and pattern based ones are extracted from the available total load and weather data. In order to conduct the feature extraction step, a set of methods, proposed by Miller and Meggers [23] for the classification of buildings based on smart meter data, are implemented and some additional procedures for extracting seasonality-related and lagged features are added. Next, a feature selection procedure is implemented in order to determine the most influential parameters in the generated pool of features, aiming at reducing the computational cost and enhancing the estimation accuracy. Finally, a genetic algorithm based optimisation procedure, developed in [24], is employed in order to determine the most promising algorithm and the corresponding optimal set of tuning parameters, which maximise the obtained accuracy. The contributions of this paper can thus be summarised as follows:

- Generating some of the temporal features that were proposed in [23] for building classification purposes in the current work aimed at disaggregation of the AC load from the smart meter data;
- Adding additional features including seasonality-based parameters and lagged values aiming at enhancing the obtained accuracy;
- Conducting a feature selection procedure in order to choose the most promising set of features among the generated large pool of features;
- Finally, optimisation of the pipeline, employing a genetic-algorithm based optimisation tool [24], aiming at choosing the most promising algorithm and the corresponding optimal tuning parameters.

2 Case study

In this study, the dataset provided by Pecan street Inc. [25], which includes the total electrical load along with appliance-by-appliance measurements (including air-conditioner) in several residential buildings, is employed. Furthermore, buildings with an integrated PV unit have been chosen; thus, the PV plant's generation at each time-stamp (which represents the solar irradiation) is also available. Accordingly, hourly data on total consumption, AC consumption and PV generation of 192 different buildings, all located in Austin, Texas, for a period of 8 days (Fig. 1) have been utilised in this study. Moreover, historical weather data of Austin in the corresponding timestamps, which includes temperature, humidity, visibility, the speed of wind, cloud coverage, along with the intensity and probability of precipitation, has been added to the dataset.

3 Overall methodology

The overall implemented methodology is represented in Fig. 2. As the first step, the dataset is processed and cleaned and invalid values are removed. The data is then standardised and is finally divided into training and testing tests.

In the next step, the feature extraction procedure is carried out, in which lagged and seasonality related (calendar based) features are first extracted and an already existing state-of the-art methodology [23] is then implemented to extract statistic based, regression model based and pattern based features. Detailed descriptions about the latter features are provided in the corresponding sub-sections.

The third step is instead focused on optimising the employed machine learning model and the corresponding tuning parameters, utilising a genetic algorithm based optimisation procedure. As the last step, a feature selection procedure is conducted, in which the most promising set of features resulting in the highest accuracy, while reducing the computational cost, is determined.

4 Pre-processing of data and feature extraction

4.1 Data pre-processing

The invalid and missing values in the dataset are first removed and buildings with no AC consumption are then discarded. For the case of buildings with multiple air-conditioners, the corresponding loads are summed up. Next, the standardisation step is conducted, which is a common requirement for many machine learning estimators. A conventional standardisation procedure is scaling features in a way that they would lie between a given minimum and maximum value (often between zero and one); thus the following transformation [26] is utilised:

$$X_{\text{std}} = \frac{(X - X_{\min})}{(X_{\max} - X_{\min})} \tag{1}$$

In the pre-processing phase, the One Hot Encoder has also been included. It refers to splitting the column which contains numerical categorical data to many columns (containing ‘0’ or ‘1’ values) depending on the number of categories present in that column. In the last phase of the data processing procedure, the dataset is split into training and testing sets. Twenty buildings are randomly chosen as the testing set and are accordingly excluded in the training phase. Hence, the accuracy of the implemented pipeline in estimating the AC load for the buildings, for which the dedicated AC sensor data is not provided, can be assessed.

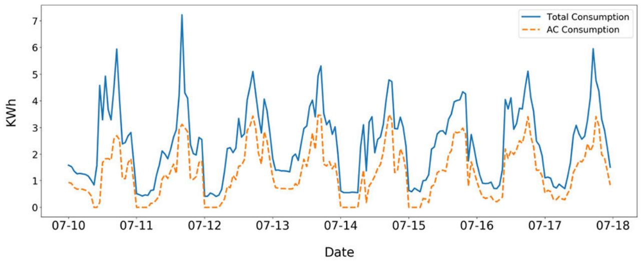


Fig. 1 *Example of total and AC load consumption for one residential building in Pecan Street dataset*

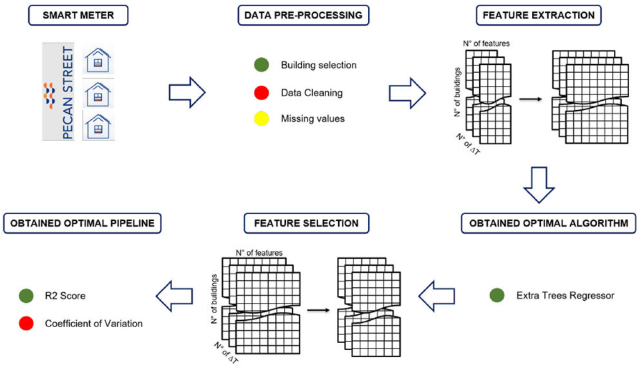


Fig. 2 *Schematic representation of the implemented methodology*

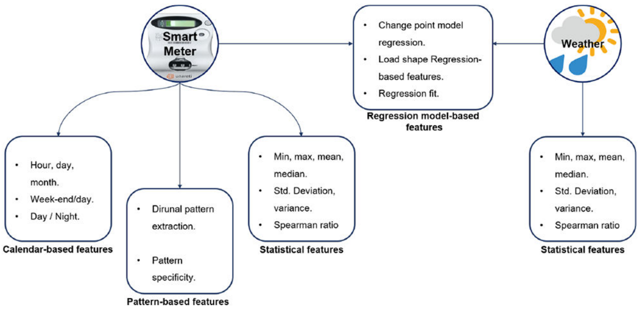


Fig. 3 *Implemented feature extraction procedure*

4.2 Features extraction process

The feature extraction procedure, represented in Fig. 3, is a key step in this study, in which seasonality related (calendar based) and lagged features are first extracted and a state-of-the-art methodology [23] is then employed to extract the statistics based, regression model based and pattern based features.

Seasonality related (calendar based) features: The extracted seasonality related features include: hour (along with cos(hour) and sin(hour)), day, month, day of the week, week of the year, weekend flag, along with the night flag.

Lagged features: Through lagging features, their previous values corresponding to AC consumption at certain time-stamps are taken into account. Adding lagged features is particularly important for the features, such as the solar irradiation (represented by PV generation), that do not have an immediate influence on the indoor temperature and thus on the AC load. In this study, lagged values of total consumption up to 12 h, solar irradiance (PV generation) from 3 to 6 h, and temperature up to 6 h are extracted and added to the dataset.

Statistical-based features: The first set of statistics-based features are the essential temporal statistics, which have been obtained on a daily bases and have been computed both for the total consumption

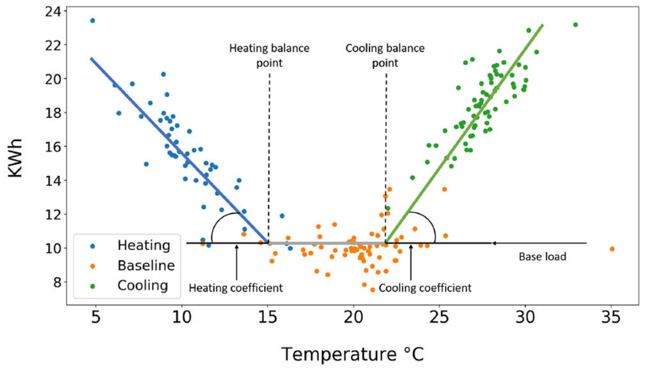


Fig. 4 *Model suggested by pattern of building energy consumption. The heating and cooling coefficients can be extracted*

and temperature. In this context, standard deviation is the square root of the variance, which is determined as

$$\sigma^2 = \frac{\sum_{i=1}^{N} (x_i - \mu)^2}{n} \tag{2}$$

The other statistical based features include ratio based ones and the Spearman rank order correlation [27], which is an index utilised to measure the correlation between the total load and the ambient temperature [23].

Regression based features: The fitted parameters of performance prediction models can provide information about the physical behaviour of buildings. In this context, the time-of-week and temperature model, utilised in the studies conducted by Price [28] and Matthieu *et al.* [29], is employed to generate indices, which assess if the total load of the building is a function of ambient temperature and the schedule and determine the corresponding degree of dependence. The latter model is implemented in EEmeter [30] toolset, which fits a linear model that can estimate the building's energy consumption as a function of the outdoor temperature. As an instance for the buildings with an electrical AC system, as the ambient temperature is rised above a certain balance point, the electrical load of the building becomes linearly proportional to the every degree of increase in ambient temperature. Thus, the slope of the latter linear relationship is the rate of increase in the AC load owing to the variation in the ambient temperature [23].

Fig. 4 represents the relationship between energy use and temperature and the fitted model. In this case, the cooling coefficient is determined and utilised as a feature.

The outputs of the linear change point model are another set of features that represent the effect of climatic conditions on the AC load [23].

Pattern-based features: Pattern-based features facilitate capturing the typical (motifs) and atypical patterns (discords) in the consumption in buildings. In the 'Day Filter' process, the mentioned patterns are extracted on a daily basis. The key feature generated in this procedure is the diurnal pattern frequency that measures the size and the number of motifs obtained for a certain building [23].

5 Machine learning: regularisation, algorithms, feature selection and pipeline optimisation

5.1 Regularisation

Regularisation is a step conducted aiming at avoiding over-fitting. Batch normalisation (BN) has been proven to be effective at accelerating and improving the training of machine learning based pipelines. In this work, L1-Norm has been selected; 1-Norm of a vector $\boldsymbol{w}$ is defined as

$$\| \boldsymbol{w} \| = |w_1| + |w_2| + \cdots + |w_n| \tag{3}$$

A norm is a mathematical function that is applied to a vector. The norm of a vector maps vector values to values in $[0,\infty)$. In machine learning, norms are useful because they are used to express distances: this vector and that vector are so-and-so far apart, according to this-or-that norm [31].

Including normalisation means that the loss equation to minimise is no longer Loss = Error($y_{\text{true}}, y_{\text{predict}}$) but it becomes

$$\text{Loss} = \text{Error}(y_{\text{true}}, y_{\text{predict}}) + \lambda \sum_{i=1}^{N} |w_i| .$$

5.2 Random forests

Random forests or random decision forests, while utilised for regression, are ensemble learning methods which are based on building several decision trees in the training process and providing the average of their predictions as the output. Considering $T_i(x)$ a single regression tree built based on a subset of input features and the bootstrapped samples [32], the tree can be expressed as

$$\hat{f}^{C}_{\text{RF}}(\boldsymbol{x}) = \frac{1}{C} \sum_{i=1}^{T} T_i(\boldsymbol{x}) \tag{4}$$

in which C represents number of trees and x is the vectored input variable [32].

The model, through the training process tries to minimise a given error metric. For the case of the present work, it is the mean squared error, that is defined as

$$\text{MSE} = \frac{1}{2} \sum_{j=0}^{N} (y_j - a_j)^2 \tag{5}$$

5.3 Extra trees regressor

Extra trees is similar to Random Forest as it builds multiple trees and splits nodes using random subsets of features though it does not bootstrap observations (meaning it samples without replacement) and in this method the nodes are split on random splits, not best splits [33].

5.4 Accuracy metrics

The metrics employed in the present work to determine the accuracy of the model are the coefficient of determination and coefficient of variation (CV).

5.4.1 Coefficient of determination: The coefficient of determination ($R2$ score) is the proportion of the variance in the dependent variable that can be predicted by the independent variable(s). It provides a measure of the agreement between the predictions of the model and the observed outcomes, based on the proportion of total variation of outcomes explained by the model [34].

If Ψ is the mean of the observed data

$$\Psi = \frac{1}{n} \sum_{i=1}^{n} y_i \tag{6}$$

then the following sum of squares can be employed to determine the variability of the dataset:

- The total sum of squares

$$\text{SS}_{\text{tot}} = \sum_{i} (y_i - \Psi)^2 \tag{7}$$

- The regression sum of squares [34]

$$\text{SS}_{\text{reg}} = \sum_{i} (f_i - \Psi)^2 \tag{8}$$

where f_i is the *ith* fitted value.

- The sum of squares of residuals

$$\mathrm{SS}_{\mathrm{res}} = \sum_i (y_i - f_i)^2 \tag{9}$$

Thus, $R2$ is defined as [34]

$$R2 = \frac{\mathrm{SS}_{\mathrm{res}}}{\mathrm{SS}_{\mathrm{tot}}} \tag{10}$$

5.4.2 Coefficient of variation: The CV is the ratio of the standard deviation σ to the mean μ, which is expressed in the following equation:

$$\mathrm{CV} = \frac{\sigma}{\mu} \tag{11}$$

5.4.3 Mutual information: Mutual information is an index that quantifies the relationship between two random variables, which are sampled at the same time. For two random variables X and Y, the mutual information is found using the following equation [35, 36]:

$$I(X; Y) = \sum_{x \in X} \sum_{y \in Y} P(x, y) \log \frac{P(x, y)}{P(x)P(x)} \tag{12}$$

in which $P(X, Y)$ is the joint distribution and where $P(X)$ and $P(Y)$ are the marginal distributions of X and Y [35, 36], defined as

$$P_X(x) = \sum_{y \in Y} P_{XY}(x, y) \tag{13}$$

5.5 Pipeline optimisation

The optimisation of the developed machine learning based pipelines is conducted utilising a genetic algorithm based optimisation tool (TPOT [24]), in which random changes are progressively applied on parts of the pipeline aiming at obtaining algorithms with a better performance. Conducting the optimisation procedure permits choosing the most suitable machine learning model and optimising the corresponding hyper-parameters.

5.6 Feature selection process

The feature extraction phase creates a large pool of features which notably increases the computation cost and can also lead to overfitting in the model, which reduces the accuracy. Thus, implementing a feature selection procedure leads to several benefits.

As represented in Fig. 5, in the first step of the implemented feature selection methodology, features are sorted based on their value of the mutual information (determined between each feature and the target, which is the AC load). Next, following the obtained order, the model is trained by adding one feature per time (accumulating features). Thus, the procedure starts with one feature and ends with a set that includes all features. Next, the combination of features (starting from the beginning), which leads to the highest $R2$ score is determined.

Finally, starting from the latter selected combination of features, the model is trained while adding one feature per time among the remaining ones and the resulting $R2$ score is determined. In this procedure, in case the $R2$ score is improved, the added feature is kept and in case not it is discarded.

6 Results and discussions

In this section, some example results of the extracted features are first presented. Next, the obtained most promising algorithm and the corresponding optimal hyper-parameters are provided. The results of feature selection step are then given and the achieved accuracy for each building using the final optimal pipeline is presented. It is worth mentioning that among the 192 available buildings, 20 buildings are employed as the testing set, while the remaining ones are included in the training set.

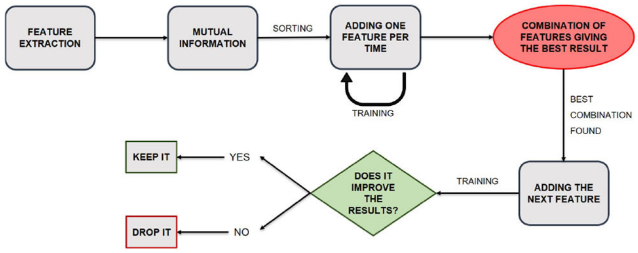


Fig. 5 *Implemented feature selection methodology*

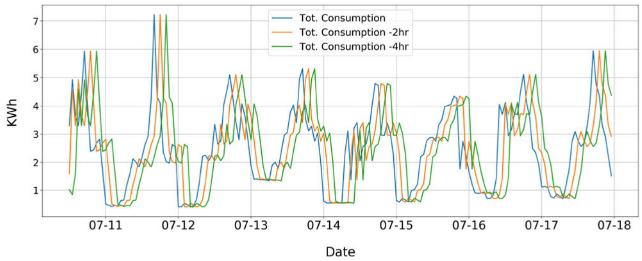


Fig. 6 *Total consumption profile and the corresponding lagged values for one of the considered residential buildings*

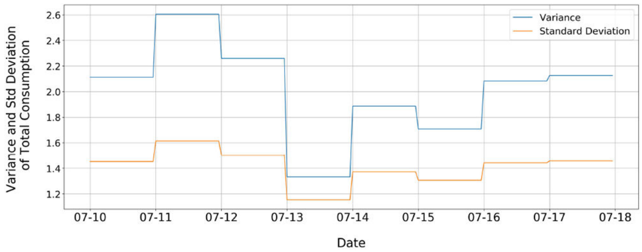


Fig. 7 *Variance and standard deviation of the total consumption for one of the considered residential buildings*

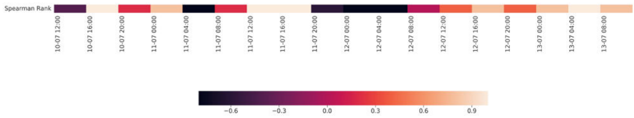


Fig. 8 *Determined Spearman rank order correlation coefficient for a residential building for a duration of 3 days*

6.1 Feature extraction results

By implementing the feature extraction procedure, 424 features, which belong to the five different categories described in Section 4.2, are extracted from the raw data. In the present section, some examples of the generated features are presented.

6.1.1 Lagged features: An example of lagged total consumption values for one of the considered residential buildings is represented in Fig. 6.

6.1.2 Statistics-based features: Fig. 7 represents the values of the standard deviation and variance of the total consumption, calculated on a daily basis, for one of the considered buildings.

Fig. 8 instead demonstrates the Spearman rank order correlation coefficient that represents the influence of the ambient temperature on the total load of the building (the higher the value the stronger is the correlation).

6.1.3 Regression-based features: The regression-based features are generated while seeking a correlation between the total consumption of the building and external factors such as temperature, radiation and their previous values. In particular, as explained in Section 4, the correlation between the total load and

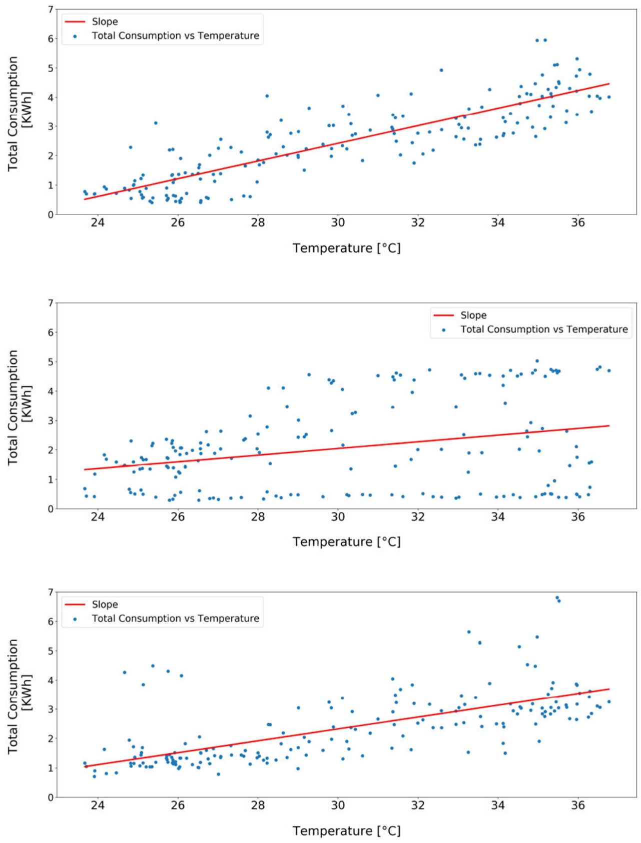


Fig. 9 *Fitted linear models and the cooling coefficients of three different buildings obtained through the regression-based feature extraction step*

Table 1 Obtained optimal hyper-parameters of the selected ExtraTreesRegressor model

Parameter	Value	Description (provided by [37])
bootstrap	false	the whole dataset is used to build each tree
max features	0.7	the number of features taken into account while determining the best split
min samples leaf	2	the minimum number of samples that are needed to be at a leaf node
min samples split	3	the minimum number of samples which are needed to split an internal node
n estimators	100	the number of trees

For the other parameters that are not specified, the default values employed in [37] are utilised.

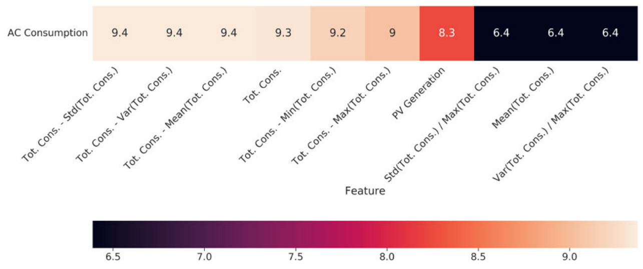


Fig. 10 *Mutual information heat map; higher is the value of the mutual information more the feature are correlated. In the heatmap dark colours mean less correlation*

the ambient temperature allows the estimation of the building cooling coefficient. Fig. 9 represents the fitted linear models for three different buildings and the corresponding obtained slopes, which are the determined cooling coefficients.

6.2 Results of the obtained optimal algorithm

By conducting the genetic algorithm based optimisation step, utilising TPOT [24], while considering a large set of different algorithms and employing mean squared error as the objective function, the Extra Trees Regressor is chosen as the most promising algorithm and the parameters, presented in Table 1, as the most suitable hyper-parameters.

6.3 Feature selection results

The feature selection procedure is next conducted, while employing the optimal algorithm that was obtained in the previous step. As was previously pointed out, the features are first sorted based on their absolute value of mutual information with respect to the target. Fig. 10 shows the heat map of the mutual information values between the target (AC consumption) and various features. It can be observed that the statistical based features, extracted in the previous phase, turn out to have a notable correlation with the AC consumption.

Once the features are sorted, they are added one by one, while keeping the previous features; the training and testing procedure are conducted at each step and the obtained accuracy is stored. Accordingly, this procedure is started with a model that is provided with only one feature and is terminated with a model that is utilising 424 features.

Next, the sequence of features, which results in the highest accuracy is determined. As demonstrated in Fig. 11, by employing the first 39 features, the maximum accuracy can be obtained. However, as can be seen in Fig. 11, several features in the second half of the sequence seem to be redundant as they do not enhance the obtained accuracy and only increase the computational cost. Therefore, in the second step, the first 19 features are chosen as the initial promising set and the remaining features are then added one by one. In case adding a feature increases the accuracy, it is kept and the corresponding obtained accuracy is stored. On the other hand, the other features, adding which reduces the accuracy, are discarded. As can be observed in Fig. 12, carrying out the second step leads to the selection of 31 features. Therefore, it is demonstrated that performing the feature selection procedure substantially reduces the number of features (424 features to 31 ones) and consequently significantly reduces the computational cost, while enhancing the obtained accuracy.

6.4 AC load disaggregation results

As was previously pointed out, 20 buildings were chosen as the testing set and were utilised to determine the accuracy of the model. While employed the final pipeline, obtained through algorithm optimisation and feature selection, the coefficient of determination (*R*2 score) and CV for individual buildings are achieved and are represented in Table 2. It can be observed that the performance of the model in general is very promising, as the obtained accuracy is comparable with the one achieved using the state-of-the-art NILM approaches while these methods utilise either a very high frequency data (every minute or every second) or measurements from installed dedicated sensors.

Nevertheless, a notable difference between the obtained accuracy for different buildings can be observed, which can be attributed to the significant variations in the corresponding occupant behaviour. Fig. 13 demonstrates the comparison between the real and estimated AC load for two different buildings. For the case of Building 13 with a regular and repetitive occupant's daily behaviour, an elevated accuracy is obtained (*R*2 score of 0.98). For the case of building 20 with a notably lower daily behaviour (in terms of turning the conditioner on and off) regularity, the achieved accuracy is reduced (*R*2 score of 0.96).

It is worth noting that, as was previously indicated, the proposed methodology was implemented using a dataset corresponding to the measurements conducted in a period of 8 days that was the duration in which the authors had access to the disaggregated data. Accordingly, although the obtained results demonstrate the performance of the proposed method, in order to determine the overall accuracy in the whole cooling period, a larger

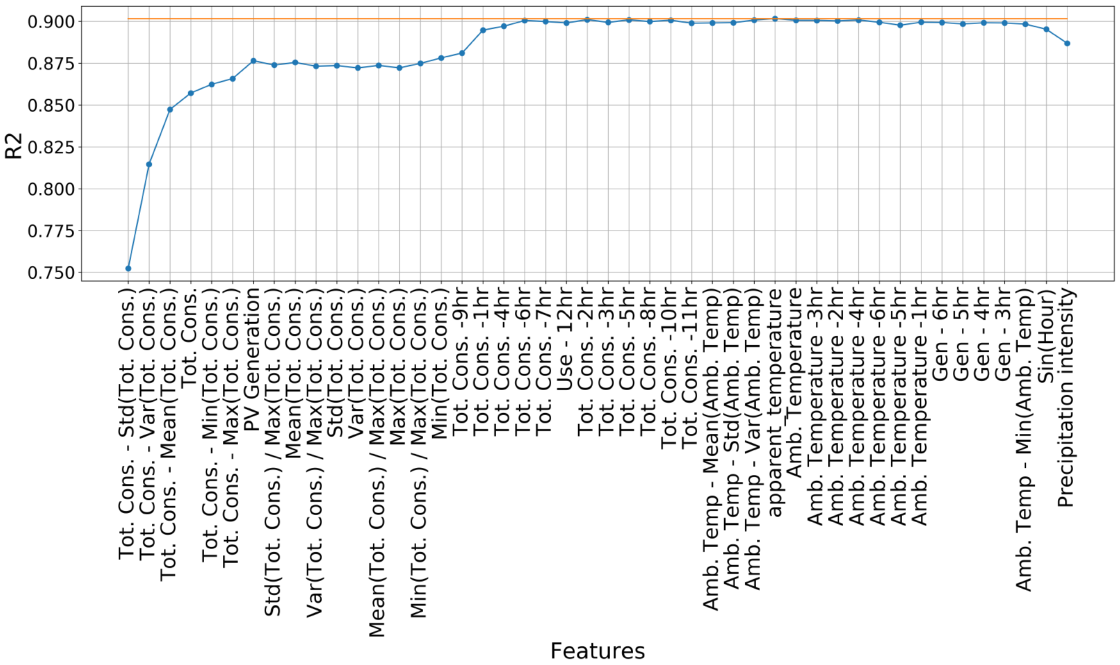


Fig. 11 *Results of the first step of the feature selection procedure*

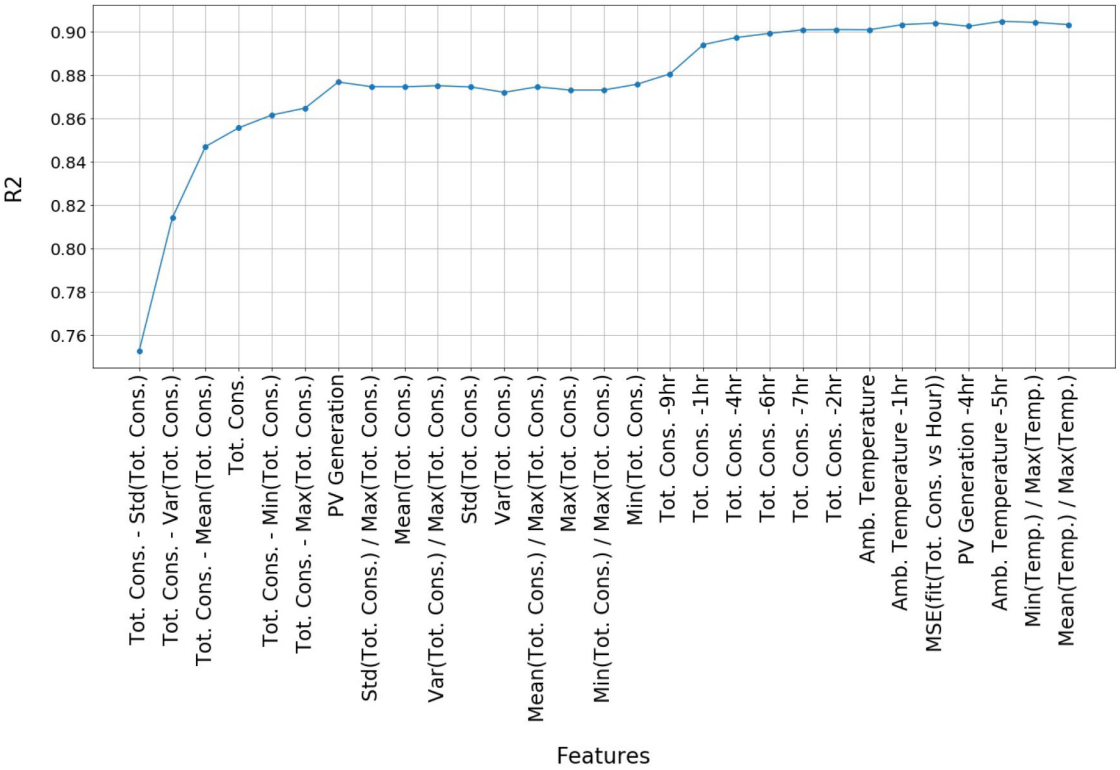


Fig. 12 *Results of the second step of the feature selection procedure*

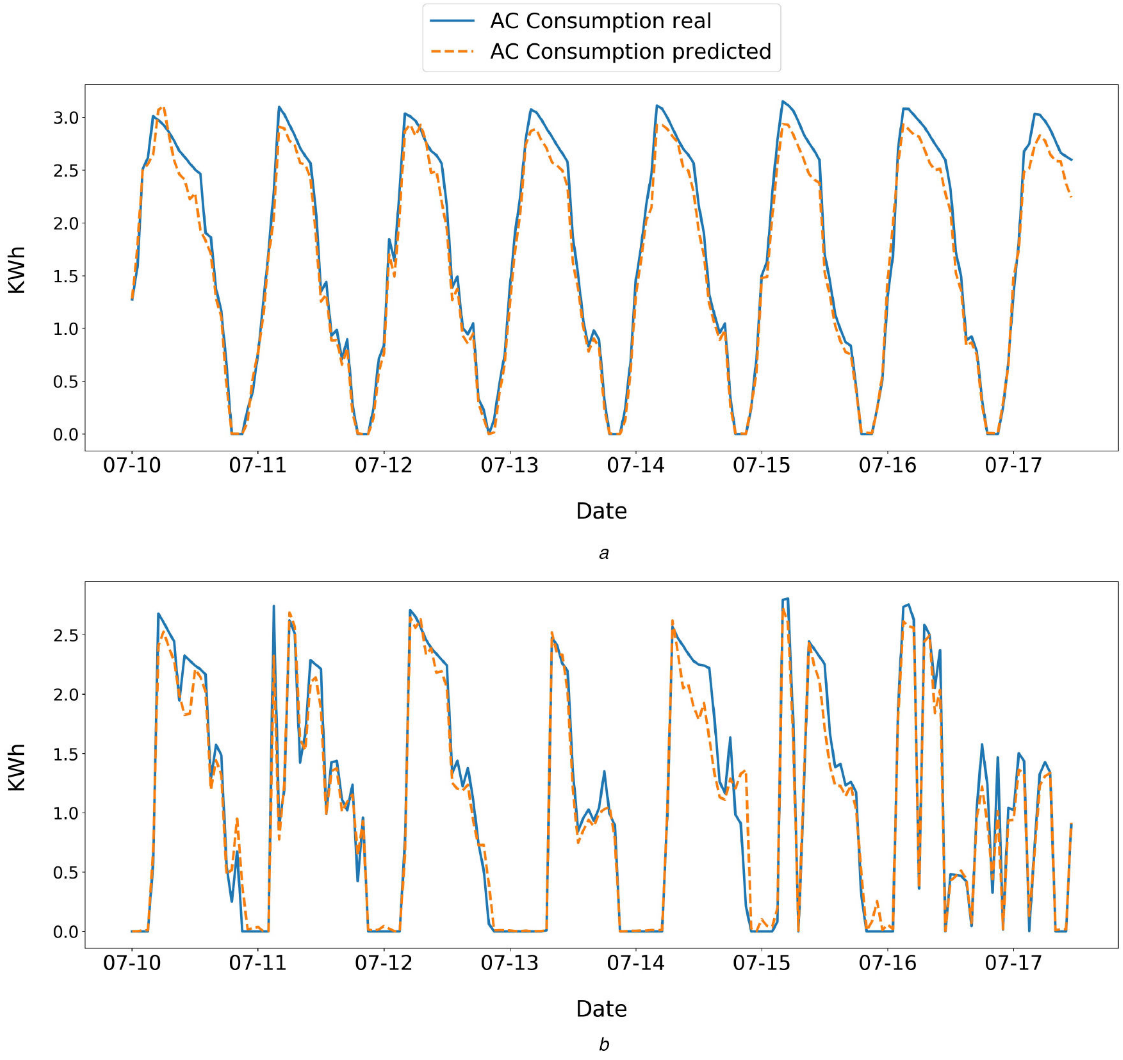
dataset should be analysed. The latter analysis will be conducted in future studies once the authors have access to the disaggregated data in a longer period.

In order to evaluate the contribution of the conducted feature extraction step, a comparison between the obtained accuracies with and without the extracted features is performed. Thus, features of the base case only include the raw data consisting of the hourly total load along with the corresponding climatic condition data. As can be seen in Table 3, the feature extraction step has improved the obtained accuracy for all of the buildings with different extents. The overall improvement (average difference) is more than 5%, which demonstrates the notable contribution of conducting the feature extraction procedure.

In order to provide a comparison between the accuracy obtained using the proposed methodology and the one achieved using conventional data-driven analysis of high frequency smart data, the results of the studies conducted by Perez *et al.* [22] can be considered. In this study, a disaggregation technique was proposed and conducted using minutely (with sampling rate of 1 min) smart meter data. Ninteen houses were sub-metered to validate the accuracy of the disaggregation technique. The $R2$ value determined by comparing the real AC consumption and the predicted value was 0.90, which is similar to the $R2$ value of 0.905 that is obtained in this study. However, utilising high granularity data (e.g. minutely) raises several privacy concerns [38] as it allows the inference of detailed information (that can be sought by several parties including criminals, advertising companies and law

Table 2 $R2$ score and CV for 20 different residential buildings and the overall set of buildings

House ID	$R2$	CV, %	House ID	$R2$	CV, %
1	0.94	17.7	11	0.90	18.9
2	0.96	21.2	12	0.77	26.9
3	0.79	24.4	13	0.98	10
4	0.85	24	14	0.85	32.9
5	0.67	28.2	15	0.92	16.2
6	0.76	30	16	0.94	25.9
7	0.92	14.5	17	0.95	16.9
8	0.96	20	18	0.90	21
9	0.94	12	19	0.85	42
10	0.91	22.4	20	0.96	18
overall				0.905	24



a

b

Fig. 13 *Example of real and predicted AC consumption loads for two residential buildings*
(a) Building 13. $R2 = 0.98$; CV = 10%, ***(b)*** Building 20. $R2 = 0.96$; CV = 18%

enforcement organisations) about the behaviour of the consumers including the time they eat, take a shower and watch TV [38] (through which the state occupants' of presence can also be estimated). Therefore, the methodology proposed in this study permits obtaining a similar accuracy to the ones obtained using high granularity data while only employing hourly data that permits evading the mentioned issues related to the consumers' privacy and also results in reducing the required data communication cost (hourly instead of minutely).

The other alternative would be utilising a dedicated sensor for the air-conditioner in each household that, although clearly guarantees a 100% accuracy, requires a notable investment and accordingly cannot be utilised by utility companies to determine the potential of AC-based DSM in a large number of households. It is worth mentioning that, while employing the obtained optimal pipelines and utilising the dataset considered in this study, the testing procedure (predicting the AC load in test set) takes around 2 s (hardware specifications: CPU: 2 cores 2.2 GHz; RAM: 12 GB; GPU accelerator: 12 GB). Therefore, considering the fact that the smart meter data is communicated hourly, the implemented process can also be utilised in an online manner in order to facilitate the prediction of sudden changes in the demand.

Table 3 $R2$ scores obtained with and without including the extracted features

House ID	Before	After	Diff., %	House ID	Before	After	Diff., %
1	0.92	0.94	1.73	11	0.89	0.90	1.12
2	0.92	0.96	4.35	12	0.74	0.77	4.05
3	0.68	0.79	16.18	13	0.89	0.98	10
4	0.83	0.85	2.41	14	0.79	0.85	7.59
5	0.65	0.67	3.08	15	0.73	0.92	26.03
6	0.71	0.76	7.04	16	0.93	0.94	1.08
7	0.91	0.92	1.1	17	0.92	0.95	2.4
8	0.93	0.96	3.6	18	0.89	0.90	1.12
9	0.91	0.94	3.3	19	0.83	0.85	2.41
10	0.85	0.91	7.06	20	0.91	0.96	5.49
overall					0.86	0.905	5.23

7 Conclusions

In the present work, a machine learning based methodology for determining the AC load from the smart meter data was proposed and implemented. In this context, a feature extraction step was first performed resulting in a pool of 424 features. Next, a genetic algorithm based optimisation procedure was carried out in order to determine the algorithm that leads to the highest accuracy. Extra Trees Regressor, with a specific set of hyper-parameters, was accordingly determined to be the most promising algorithm. In the following step, while utilising the latter algorithm, a feature selection step was conducted that reduced the number of features to 31. By utilising the final pipeline obtained through the above-mentioned steps, an average $R2$ score of 0.905 and CV score of 24% was achieved.

Therefore, it was demonstrated that, by utilising the proposed methodology, the two main limitations of the commonly utilised NILM approaches, i.e. the necessity of providing either high frequency data or appliance-by-appliance measurements, can be overcome, while an elevated accuracy is achieved. Evading the necessity of providing dedicated appliance measurements results in a notable cost saving by avoiding the installation of additional sensors. Evading the necessity of using high frequency data instead facilitates the utilisation of the data from commonly installed smart meters and significantly reduces the user privacy concerns.

8 References

[1] International Energy Agency: 'Energy efficiency 2018', 2018

[2] Perfumo, C., Kofman, E., Braslavsky, J.H., *et al.*: 'Load management: model-based control of aggregate power for populations of thermostatically controlled loads', *Energy Convers. Manage.*, 2012, **55**, pp. 36–48

[3] Mathieu, J.L., Koch, S., Callaway, D.S.: 'State estimation and control of electric loads to manage real-time energy imbalance', *IEEE Trans. Power Syst.*, 2013, **28**, pp. 430–440

[4] Lu, N.: 'An evaluation of the hvac load potential for providing load balancing service', *IEEE Trans. Smart Grid*, 2012, **3**, pp. 1263–1270

[5] Gellings, C.W.: ' Evolving practice of demand-side management', *J. Modern Power Syst. Clean Energy*, 2017, **5**, pp. 1–9

[6] Arghandeh, R., Woyak, J., Onen, A., *et al.*: 'Economic optimal operation of community energy storage systems in competitive energy markets', *Appl. Energy*, 2014, **135**, pp. 71–80. Available at http://www.sciencedirect.com/science/article/pii/S0306261914008770

[7] Pourshahriar, H.: 'Correct vs. accurate prediction: a comparison between prediction power of artificial neural networks and logistic regression in psychological researches', *Procedia Soc. Behav. Sci.*, 2012, **32**, pp. 97–103

[8] Gilanifar, M., Konila Sriram, H., Wang, L.M., *et al.*: 'Multi-task bayesian spatiotemporal gaussian processes for short-term load forecasting', *IEEE Trans. Ind. Electron.*, 2020, **67**, (6), pp. 5132–5143

[9] Konila Sriram, J., Cordova, L.M., Kocatepe, A., *et al.*: 'Combined electricity and traffic short-term load forecasting using bundled causality engine', *IEEE Trans. Intell. Transp. Syst.*, 2019, **20**, (9), pp. 3448–3458

[10] Konila Sriram, L.M., Gilanifar, M., Erman Ozguven, Y., *et al.*: 'Causal markov elman network for load forecasting in multinetwork systems', *IEEE Trans. Ind. Electron.*, 2019, **66**, (2), pp. 1434–1442

[11] Hassan, M.G., Hirst, R., Siemieniuch, C., *et al.*: 'The impact of energy awareness on energy efficiency', *Int. J. Sustain. Eng.*, 2009, **2**, pp. 284–297

[12] Markus, W., Adrian, H., Friedemann, M., *et al.*: 'Leveraging smart meter data to recognize home appliances'. IEEE Int. Conf. Pervasive Computing and Communications, Lugano, Switzerland, 2012

[13] Kong, W., Dong, Z.Y., Ma, J., *et al.*: 'An extensible approach for non-intrusive load disaggregation with smart meter data', *IEEE Trans. Smart Grid*, 2018, **9**, (4), pp. 3362–3372

[14] Ponocko, J., Milanovic, J.V.: 'Forecasting demand flexibility of aggregated residential load using smart meter data', *IEEE Trans. Power Syst.*, 2018, **33**, (5), pp. 5446–5455

[15] Guo, Z., Wang, Z.J., Kashani, A.: 'Home appliance load modeling from aggregated smart meter data', *IEEE Trans. Power Syst.*, 2015, **30**, (1), pp. 254–262

[16] Manivannan, M., Najafi, B., Rinaldi, F.: 'Machine learning-based short-term prediction of air-conditioning load through smart meter analytics', *Energies*, 2017, **10**, (11), 1905

[17] Kelly, J., Knottenbelt, W.: 'Neural nilm: deep neural networks applied to energy disaggregation'. Proc. 2nd ACM Int. Conf. Embedded Systems for Energy-Efficient Built Environments – BuildSys '15, Seoul, South Korea, 2015

[18] Najafi, B., Moaveninejad, S., Rinaldi, F.: 'Data analytics for energy disaggregation: methods and applications, in Arghandeh, R., Zhou, Y. (Eds.) '*Big data application in power systems*' (Elsevier, Netherlands, 2017) pp. 377–408

[19] Hart, G.W.: 'Nonintrusive appliance load monitoring', *Proc. IEEE*, 1992, **80**, pp. 1870–1891

[20] Ayumu-Miyasawa, Y.H., Yu, F.: 'Energy disaggregation based on smart metering data via semi-binary nonnegative matrix factorization', *Energy Build.*, 2019, **183**, pp. 547–558

[21] Gilanifar, M., Wang, H., Ozguven, E.E., *et al.*: 'Bayesian spatiotemporal gaussian process for short-term load forecasting using combined transportation and electricity data', *ACM Trans. Cyber-Phys. Syst.*, 2019, **4**, pp. 2:1–2:25

[22] Perez, K.X., Cole, W.J., Rhodes, J.D., *et al.*: 'Nonintrusive disaggregation of residential air-conditioning loads from sub-hourly smart meter data', *Energy Build.*, 2014, **81**, pp. 316–325

[23] Miller, C., Meggers, F.: 'Mining electrical meter data to predict principal building use, performance class, and operations strategy for hundreds of non-residential buildings', *Energy Build.*, 2017, **156**, pp. 360–373

[24] Olson, R.S., Bartley, N., Urbanowicz, R.J., *et al.*: 'Evaluation of a tree-based pipeline optimization tool for automating data science'. Proc. Genetic and Evolutionary Computation Conf. 2016, Denver, Colorado, USA, 2016, pp. 485–492

[25] Liang, H., Ma, J., Sun, R., *et al.*: 'A data-driven approach for targeting residential customers for energy efficiency programs', *IEEE Trans. Smart Grid*, 2019, **11**, (2), pp. 1229–1238

[26] C. on https://scikit-learn.org, 'Scikit-learn: Minmax scaler'

[27] Miller, C., Schlueter, A.: 'Forensically discovering simulation feedback knowledge from a campus energy information system'. Proc. of the 6th annual Symposium on Simulation for Architecture and Urban Design (SimAUD), Washington, DC, USA, 2015, pp. 136–143

[28] Price, P.: 'Methods for analyzing electric load shape and its variability'. No. LBNL-3713E. Lawrence Berkeley National Lab.(LBNL), Berkeley, CA (United States), 2010

[29] Mathieu, J.L., Price, P.N., Kiliccote, S., *et al.*: 'Quantifying changes in building electricity use, with application to demand response', *IEEE Trans. Smart Grid*, 2011, **2**, (3), pp. 507–518

[30] openeemeter. 'Eemeter', 2018. Available from: https://github.com/openeemeter/eemeter

[31] Ioffe, S., Szegedy, C.: 'Batch normalization: Accelerating deep network training by reducing internal covariate shift'. Proc, of the 32nd International Conference on Machine Learning, Lille, France, 2015

[32] Breiman, L.: 'Random forests', *Mach. Learn.*, 2001, **45**, (1), pp. 5–32

[33] Geurts, P., Ernst, D., Wehenkel, L.: 'Extremely randomized trees', *Mach. Learn.*, 2006, **63**, (1), pp. 3–42

[34] Wikipedia: 'Coefficient of determination', 2019. [Online; 16/12/2019]. Available from: https://en.wikipedia.org/wiki/Coefficient_of_determination

[35] Latham, P.E., Roudi, Y.: 'Mutual information', *Scholarpedia*, 2009, **4**, (1), p. 1658

[36] Learned-Miller, E.G.: 'Entropy and mutual information'. Department of Computer Science, University of Massachusetts, Amherst, 2013

[37] Pedregosa, F., Varoquaux, G., Gramfort, A., *et al.*: 'Scikit-learn: machine learning in Python', *J. Mach. Learn. Res.*, 2011, **12**, pp. 2825–2830

[38] Dong, R., Ratliff, L.J.: 'Energy disaggregation and the utility-privacy tradeoff", in Arghandeh, R., Zhou, Y. (Eds.):'*Big data application in power systems*' (Elsevier, Netherlands, 2018), pp. 377–408