PATH SIGNATURES FOR NON-INTRUSIVE LOAD MONITORING

Paul Moore[⋆], *Theodor-Mihai Iliant*[⋆], *Filip-Alexandru Ion*[⋆], *Yue Wu*[†], *Terry Lyons*[⋆]

[⋆]Mathematical Institute, Oxford University
[†]Department of Mathematics and Statistics, University of Strathclyde, Glasgow

ABSTRACT

Non-intrusive load monitoring (NILM) is the analysis of electricity loads by means of a single supply wire, so avoiding separate monitors on individual appliances. Some approaches to NILM use the V-I trajectory for feature generation but they apply ad-hoc rules to generate the feature vector. This paper demonstrates a systematic method of feature generation called the path signature which has recently been applied in machine learning, often with notable success. We show how the path signature generates features from the V-I trajectory to give a test set accuracy of 98.81% on the COOLL dataset. We conclude that the path signature is easier to use and generalize than ad-hoc features, and it can be applied to many other applications which use multivariate sequential data.

***Index Terms*—** non-intrusive load monitoring, disaggregation, machine learning, feature selection, path signatures

1. INTRODUCTION

Features for machine learning are often manually crafted in a process called feature engineering. In this process the practitioner selects inputs and transformations of those inputs that are expected to be effective for the application. Feature engineering has the advantage that domain expertise can be incorporated *a-priori* into the model. A disadvantage is that the process of feature generation and selection can be arbitrary, and potential sources of information can be missed. This disadvantage is apparent in the case of multivariate data, where interactions between variables may be latent or obscure. In this paper we demonstrate the effectiveness of the path signature in finding interactions within load data for identifying appliances.

1.1. Non-intrusive load monitoring

Non-intrusive load monitoring (NILM) involves the monitoring of the electricity load at a single wire, for example on the supply to a house, and its analysis into individual appliance usage. The research field is usually traced back to [1], which described a load monitoring device based on cluster analysis. The field has progressed in the last two decades motivated in part by the need to reduce consumer energy use, and its implementation has been facilitated by machine learning. There are a number of reasons why appliances are monitored: to allow consumers to manage their energy usage, to detect appliance faults or aging, and for gathering long-term appliance usage statistics. So there is a strong motivation to pursue the goal of effective appliance monitoring at the supply.

1.2. Datasets

Academic treatments of the subject have been stimulated by the collection of datasets and the development of analysis techniques and software such as the NILMTK toolkit [2]. The first major database to be collected was the the MIT Reference Energy Disaggregation Data Set (REDD) database [3] in 2011, and more datasets have subsequently been released by researchers in different countries. Summaries of databases can be found in [2, 4–6].

The data used in this paper from is from the Controlled On/Off Loads Library (COOLL) dataset [7], which holds high frequency current and voltage measurements representing the load characteristics of individual appliances. These measurements were taken under laboratory conditions with precise control of the times when the appliances were switched on and off. The COOLL dataset is appropriate for this study because measurements are available for single appliances switched on one at a time, so facilitating the analysis of machine learning features. The high frequency COOLL data is compressed using the Free Lossless Audio Codec (FLAC) format with separate files for the current and voltage signals. A total of 42 appliances of 12 types were measured at a sampling frequency of 100 kHz. For each appliance, 20 measurements are available, each corresponding to a specific phase delay ranging from 0 to 19 ms with a step of 1 ms. In this way, an appliance is sampled over the whole range of phase delays in a single voltage cycle.

1.3. NILM features

In NILM research features are typically used for both supervised learning, where they are mapped to a label, and for unsupervised learning where the feature values are organised without the use of labelled data. Unsupervised approaches,

Supported by UK EPSRC grant EP/S026347/1.

for example clustering, have been of interest in recent years because they are more practically applicable than supervised models, which may not have been trained on all the appliances to be used. A short review of unsupervised methods for NILM is given in [6], and more general reviews can be found in [4, 8, 9]. A convenient taxonomy of features commonly used in NILM is given in [10].

Use of V-I trajectories was first suggested in [11] which describes the analysis of more than 120 loads and proposes the V-I trajectory as a basis for classifying electric load signatures. The method is further developed in [12] and [13] in which a feature vector is derived from the V-I trajectory by quantifying 'shape features' of the trajectory. For example, a resistive load, in which current is proportional to voltage has a straight line trajectory, while non-linear loads give rise to a loop whose area is proportional to the phase delay. Shape features have been found to be both interpretable and effective at classifying aggregate loads into meaningful groups of appliances [13]. Further work on deriving features from the V-I trajectory is described in [14], which proposed 10 features derived from the shape of the trajectory, and which are interpretable in physical terms. A new set of 12 shape features, both steady-state and transient, is proposed in [15]. These features are used as a reference set whose classification results we reproduce to provide a baseline for the experiments using the path signature.

2. METHOD

In this section we introduce the path signature and show how it can transform the V-I trajectory into a sequence of numbers which can be used as predictors in machine learning. We investigate two separate supervised learning tasks, 1) predicting the reference features themselves, and 2) predicting appliance labels from the COOLL dataset. The first task uses the path signature to predict the reference features which are derived from the shape of the V-I trajectory. The second task is to compare the performance of the path signature and the reference features in predicting appliance labels.

2.1. Path signature

In practical terms, the path signature is easy to use: Python packages *esig* and *iisignature* can be used to compute path signatures directly from the V-I trajectory. A visual example of signature computation can be found in [16, p.7]. Here we give the relevant mathematical theory of the signature method, which is essentially a sequence of real numbers which summarizes the trajectory. It is defined as follows: a trajectory or path X through a space $\mathbb{R}^d$ is a continuous mapping from an interval $[a, b]$ to $\mathbb{R}^d$. The path is dependent on parameter $t \in [a, b]$, and can be written,

$$X_t = \{X_t^1, X_t^2, X_t^3, \dots X_t^d\} \tag{1}$$

The kth-fold iterated integral of X is given by,

$$S(X)_{a,t}^{i_1,\dots i_k} = \int_{a<t_k<t} \cdots \int_{a<t_1<t_2} dX_{t_1}^{i_1} \dots dX_{t_k}^{i_k} \tag{2}$$

The path signature is a collection of all the iterated integrals of X,

$$S(X)_{a,b} = (1, S(X)_{a,b}^1, S(X)_{a,b}^2, S(X)_{a,b}^{1,1}, S(X)_{a,b}^{1,2}, \dots) \tag{3}$$

$S(X)_{a,b}$ is a sequence of real numbers, and the superscripts are drawn from the set G of all multi-indexes,

$$G = \{(i_1, \dots, i_k) | k \geqslant 1, i_1, \dots, i_k \in \{1, \dots, d\}\} \tag{4}$$

So in two dimensions a path signature of degree 2 is $S = \{1, S^{(1)}, S^{(2)}, S^{(1,1)}, S^{(1,2)}, S^{(2,1)}, S^{(2,2)}\}$ while a path signature of degree 3 would include the terms $S^{(1,1,1)}, S^{(1,1,2)}$ etc.. The path signature was originally introduced by Chen [17] who applied it to piecewise smooth paths, and it was further developed by Lyons and others [18, 19]. In this study we use the log signature [16] which holds the same information as the path signature but in a more compact form: for a given degree, the log signature is shorter. Visualizations of the first two levels of the path signature are shown in [16, p.22], and examples of its application are given in [20, 21].

In the experiments we use a path in 3-dimensions: time, voltage and current, where time increases from 0 to 1 over the cycle, and voltage and current are measured in volts and amperes respectively. We generate the log signature from each of the 840 samples in the COOLL dataset for both transient and steady state cycles.

2.2. Reference features

As a reference set of features derived from the shape of the V-I trajectory, we use those proposed in [15]. A summary of each feature is given later in Table 1, and a more precise definition can be found in [15]. The first 8 reference features are derived from the steady state trajectory, denoted *STD*, and the last 4 features are derived from both steady state and the transient trajectory, denoted *STD/TRN*.

We generate the reference set of features for the COOLL data using the software from [15]. Steady state features are generated from the V-I trajectory which starts at 60 voltage cycles after the appliance is switched on, from the point where the voltage transitions from a negative to a positive value. Similarly, transient features are sampled at 2 voltage cycles after the switch-on time, again starting from when the voltage crosses zero from a negative value. The trajectory used for generating features is the path of the time, voltage and current values over the nominal period of a single 50 Hz cycle, that is 20 ms.

2.3. Machine learning

We use the voltage and current data from the COOLL dataset for two supervised learning tasks. The first task is to predict the values of reference features using the log signature of the steady state trajectory. The second task is to classify the appliance labels using the log signature as input, and compare results with those using the reference feature set. In both tasks we train models by 5-fold cross-validation on 80% of the 840 samples, and use the remaining 20% as a test set. Since each individual appliance is sampled 20 times over different phase delays, the training set is constructed from 16 randomly chosen samples from each appliance, the test set from the remaining 4 samples. This method ensures that distribution of appliance labels is similar in both the test and training sets. In common with the results published in [15] we take as the output labels the appliance type, rather than the specific instances of each appliance.

2.3.1. *Predicting features*

For predicting the reference features, an ensemble model is used, consisting of a bag of decision trees, also called a random forest. A single set of model hyperparameters is used for predicting all the reference features. For predicting those features derived only from the steady state trajectory, the log signature of the steady state trajectory is used as a feature vector. For predicting the reference features which use both transient and steady state trajectories, the feature vector is formed by concatenating the respective log signatures.

2.3.2. *Predicting appliance labels*

We use the log signature of degree 3 to predict appliance labels, and compare the results with a replication of the results published in [15]. Then we use a selected subset of features for each of the reference and signature sets, and compare those results. For replicating the published results we use the three types of classifier described in [15] with some differences in the training protocol. The classifiers are: an ensemble of decision trees (ENS), a K-nearest neighbour (KNN) classifier, and a support vector machine (SVM). Each model is selected by minimising the 5-fold cross-validation error on the training set. For the ENS classifier, the specific model chosen was a bag of trees (random forest), for the KNN, a one-nearest-neighbour was used with Euclidean distance, and for the SVM, a polynomial kernel was used. Full details of the model choices are available in the code repository at `https://github.com/Fivetuple/nilm_sig`. Using each chosen model we train on the training set and report both the cross-validation training accuracy and the accuracy on the separate test set.

3. RESULTS

3.1. Predicting reference features

The results for predicting each reference feature from the corresponding log signature of the time-voltage-current path are shown in Fig. 1, with numerical errors given in Table 1. Most features are predicted to within 20% using the median absolute percentage error as a metric.

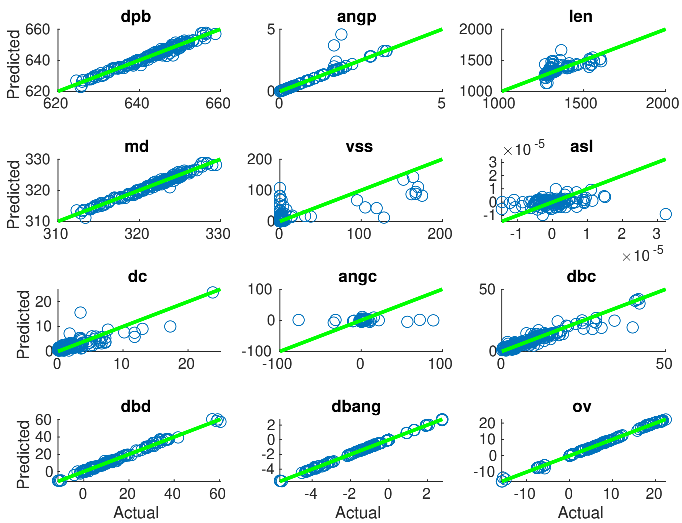

Fig. 1. Prediction of the 12 features in the reference set using a log path signature of degree 5. The line $x = y$ is shown. Axis limits have been chosen to exclude a small number of outliers.

Table 1. Predicting reference features

Name	Trajectory	Geometric interpretation	% Error
dpb	STD	Distance from min to max points	0.09
angp	STD	Angle between min and max points	1.19
len	STD	Length of trajectory	0.40
md	STD	Max distance to origin	0.09
vss	STD	Variations of the signal slope	239.17
asl	STD	Average slope value	104.29
dc	STD	Distance from centroid to origin	36.64
angc	STD	Angle between centroid and origin	99.62
dbc	STD/TRN	Difference between centroids	14.60
dbd	STD/TRN	Difference between dbp	8.24
dbang	STD/TRN	Difference between angp	2.11
ov	STD/TRN	Difference between max current	2.19

STD denotes the steady state trajectory, *TRN* the transient.
Features with a high median absolute error are highlighted.

The features *vss*, *asl*, and *angc* each have prediction errors close to or greater than 100%. Feature *vss* is the variation of the signal slope, which is a way of representing signal frequency information, and *asl* represents the average slope value. Both these features take a threshold whose value is fixed. The feature *angc* is the angle, in degrees, between the centroid of the V-I trajectory (steady state) and the origin. In theoretical terms, since the the V-I trajectory is uniquely represented by a signature of sufficient degree any features derived solely from the trajectory can be predicted accurately. In practice however insufficient training data can contribute to prediction errors.

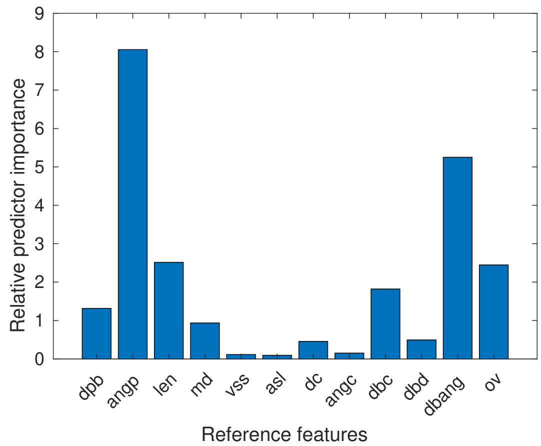

Fig. 2. Relative predictor importance for the reference features when used for predicting appliance labels. A more important feature has a greater predictive value.

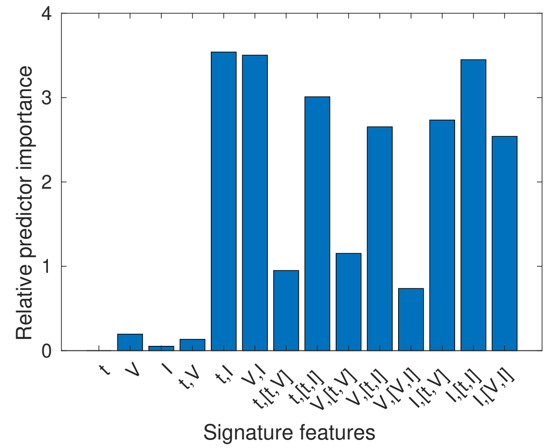

Fig. 3. Relative predictor importance for the log signature when used for predicting appliance labels. A more important feature has a greater predictive value.

3.2. Predicting appliance labels

3.2.1. Using reference features

We first replicate the experiment in [15] in which the appliance labels are predicted using the reference set of features[1]. Our replicated cross-validation and test set results are shown in Table 2, along with those published in [15] for comparison. The reproduced results are close to those in the earlier study, allowing for differences in training and model selection.

Table 2. Appliance classification using reference features.

Method	ENS	KNN	SVM
Published [15]	98.10 (1.04)	87.94 (2.50)	94.29 (1.76)
Replication	97.77 (0.74)	85.42 (2.40)	93.30 (2.37)
Test set	97.62	83.33	95.24

Accuracy shown as percentage correct with standard deviation in brackets.

Fig. 2 shows the relative importance of each feature in the reference set when the ENS model is used for label prediction. The two most important features are *angp* and *dbang*. By comparison [15] identified five features using a selection algorithm: *angp*, *md*, and *dbang* from the reference set, and two features from [14], *ar*, an area proportional to the phase difference between current and voltage, and *r*, the curvature of the mean trajectory line. Fig. 2 also shows that the features *vss*, *asl*, and *angc* have almost no predictive value. These three features were those found to be difficult to predict using the log signature, as shown in Table 1.

3.2.2. Using signatures

Table 3, third row, shows the results of predicting appliance labels using the log signature and the ENS classifier. For comparison, the first two rows give the published and replicated results. The last three rows are for a selected set of features, again giving published and replication results for comparison. The selected reference features used were *angp*, *md*, and *dbang*, with *ar* and *r* from [14]. For the signature features, the 7 most important terms identified from Fig. 3 were used. The replication shows a lower mean cross-validation result than the published value, but a similar test set result. The accuracy when using the 7 most important terms in the log signature is similar to that for the reference features.

Table 3. Appliance classification using signature features.

Set	Method	Features	Accuracy(SD)	Test set
Full	Published [15]	12	98.10 (1.04)	–
	Replication	12	97.77 (0.74)	97.62
	Signature	28	98.81 (1.13)	98.81
Selected	Published [15]	5	99.37 (0.63)	–
	Replication	5	98.51 (1.17)	99.40
	Signature	7	99.11 (0.82)	98.81

Accuracy shown as percentage correct with standard deviation in brackets.

Conclusion

The path signature uniquely characterizes a V-I trajectory with a sequence of real numbers which can be used as a feature vector for machine learning. In this application it provides accuracy comparable with shape features, but while these ad-hoc features have been carefully refined since their inception, we found some to be redundant. The path signature provides a more systematic and generalizable approach to feature generation. Since software is available for its computation it is in effect easier to implement than ad-hoc features. As such it is a good choice for this application, and it is widely applicable when using multivariate sequential data in machine learning.

[1]We thank Bruna Mulinari for help in replicating the experiments in [15]

4. REFERENCES

[1] George W Hart, Edward C Kern Jr, and Fred C Schweppe, "Non-intrusive appliance monitor apparatus," aug 1989, US Patent 4,858,141.

[2] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh, and Mani Srivastava, "NILMTK: an open source toolkit for non-intrusive load monitoring," in *Proceedings of the 5th international conference on Future energy systems*, 2014, pp. 265–276.

[3] J Zico Kolter and Matthew J Johnson, "REDD: A public data set for energy disaggregation research," *Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA*, vol. 25, no. Citeseer, pp. 59–62, 2011.

[4] R Gopinath, Mukesh Kumar, C Prakash Chandra Joshua, and Kota Srinivas, "Energy management using non-intrusive load monitoring techniques-state-of-the-art and future research directions," *Sustainable Cities and Society*, p. 102411, 2020.

[5] Jack Kelly and William Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific data*, vol. 2, no. 1, pp. 1–14, 2015.

[6] Roberto Bonfigli, Stefano Squartini, Marco Fagiani, and Francesco Piazza, "Unsupervised algorithms for non-intrusive load monitoring: An up-to-date overview," in *2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC)*. IEEE, 2015, pp. 1175–1180.

[7] Thomas Picon, Mohamed Nait Meziane, Philippe Ravier, Guy Lamarque, Clarisse Novello, Jean-Charles Le Bunetel, and Yves Raingeaud, "COOLL: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification," *arXiv preprint arXiv:1611.05803*, 2016.

[8] Hui Liu, *Non-intrusive Load Monitoring: Theory, Technologies and Applications*, Springer Nature, 2019.

[9] Nasrin Sadeghianpourhamami, Joeri Ruyssinck, Dirk Deschrijver, Tom Dhaene, and Chris Develder, "Comprehensive feature selection for appliance classification in nilm," *Energy and Buildings*, vol. 151, pp. 98–106, 2017.

[10] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16838–16866, 2012.

[11] KH Ting, Mark Lucente, George SK Fung, WK Lee, and SYR Hui, "A taxonomy of load signatures for single-phase electric appliances," in *IEEE PESC (Power Electronics Specialist Conference)*, 2005, pp. 12–18.

[12] HY Lam, KH Ting, WK Lee, and GSK Fung, "An analytical understanding on voltage-current curve of electrical load," in *International Conference on Electrical Engineering (ICEE)*, 2006, pp. 1–6.

[13] Hong Yin Lam, GSK Fung, and WK Lee, "A novel method to construct taxonomy electrical appliances based on load signatures," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 653–660, 2007.

[14] A Longjun Wang, B Xiaomin Chen, C Gang Wang, and D Hua, "Non-intrusive load monitoring algorithm based on features of V–I trajectory," *Electric Power Systems Research*, vol. 157, pp. 134–144, 2018.

[15] Bruna M Mulinari, Daniel P de Campos, Clayton H da Costa, Hellen C Ancelmo, André E Lazzaretti, Elder Oroski, Carlos RE Lima, Douglas PB Renaux, Fabiana Pottker, and Robson R Linhares, "A new set of steady-state and transient features for power signature analysis based on VI trajectory," in *2019 IEEE PES Innovative Smart Grid Technologies Conference-Latin America (ISGT Latin America)*. IEEE, 2019, pp. 1–6.

[16] Ilya Chevyrev and Andrey Kormilitzin, "A primer on the signature method in machine learning," *arXiv preprint arXiv:1603.03788*, 2016.

[17] Kuo-Tsai Chen, "Integration of paths–a faithful representation of paths by noncommutative formal power series," *Transactions of the American Mathematical Society*, vol. 89, no. 2, pp. 395–407, 1958.

[18] Ben Hambly and Terry Lyons, "Uniqueness for the signature of a path of bounded variation and the reduced path group," *Annals of Mathematics*, pp. 109–167, 2010.

[19] Terry Lyons, "Rough paths, signatures and the modelling of functions on streams," *arXiv preprint arXiv:1405.4537*, 2014.

[20] PJ Moore, TJ Lyons, J Gallacher, and Alzheimers Disease Neuroimaging Initiative, "Using path signatures to predict a diagnosis of Alzheimers disease," *PloS one*, vol. 14, no. 9, pp. e0222212, 2019.

[21] Imanol Perez Arribas, Guy M Goodwin, John R Geddes, Terry Lyons, and Kate EA Saunders, "A signature-based machine learning model for distinguishing bipolar disorder and borderline personality disorder," *Translational psychiatry*, vol. 8, no. 1, pp. 1–7, 2018.