2020:00530- Unrestricted

Report

Applications of Big Data and Data Science in the Electricity Distribution Grid

State-of-the-art

Author(s)
Karoline Ingebrigtsen
Volker Hoffmann

SINTEF Energy Research
Energy Systems

2020-11-02

SINTEF Energi AS
SINTEF Energy Research
Address:
Postboks 4761 Torgarden
NO-7465 Trondheim
NORWAY
Switchboard: +47 45456000

energy.research@sintef.no

Enterprise /VAT No:
NO 939 350 675 MVA

KEYWORDS:
Smart electricity meter
AMS
Big data
Data science
Distribution grid

Report

Applications of Big Data and Data Science in the Electricity Distribution Grid

State-of-the-art

VERSION	DATE
1.0	2020-11-02

AUTHOR(S)
Karoline Ingebrigtsen
Volker Hoffmann

CLIENT(S)	CLIENT'S REF.
Hafslund Nett AS	Åshild Vatne
PROJECT NO.	**NUMBER OF PAGES**
502001581	45

ABSTRACT
Almost every household in Norway has a smart electricity meter installed. The advanced functionality of the meters and the large amounts of data transmitted from them have the potential to contribute to more than just a more accurate billing of customers. This report is a result from the ENERGYTICS project, which aims at exploring this potential.

This state-of-the-art includes a description of experiences and technical advances regarding deployment of smart meters in the Scandinavian countries and Finland and possible benefits and advantages of deploying data science on data from the electricity grid. Different data sources in the distribution electricity grid and relevant data science techniques are presented. The focus of the state-of-the-art is on two applications of data science: clustering of customers and disaggregation of total electricity consumption into the consumption of individual loads.

PREPARED BY	SIGNATURE
Karoline Ingebrigtsen	Karoline Ingebrigtsen
CHECKED BY	**SIGNATURE**
Eivind Solvang	Eivind Solvang Eivind Solvang (Nov 5, 2020 12:17 GMT+1)
APPROVED BY	**SIGNATURE**
Knut Samdal	Knut Samdal Knut Samdal (Nov 5, 2020 12:32 GMT+1)

REPORT NO.	ISBN	CLASSIFICATION	CLASSIFICATION THIS PAGE
2020:00530	978-82-14-06508-4	Unrestricted	Unrestricted

Table of contents

Abbreviations

Abbreviation	Explanation
AMS	Advanced metering system (smart electricity meter)
AC	Air conditioning
AUPs	Appliance usage patterns
BIC	Bayesian Information Criterion
CF-PSO	Constriction Factor Particle Swarm Optimisation
CFSFD	Fast search and find of density peaks
CNN	Convolutional Neural Network
COP index	Context-independent, Optimality, and Partiality index
DAE	Denoising autoencoder
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DFT	Discrete Fourier Transform
DMS	Distribution Management System
DSO	Distribution system operator
DTW	Dynamic Time Warping
EDHMM-diff	Explicit-Duration Hidden Markov Model with differential observations
END	Equal N-Dimensional
EVs	Electric vehicles
FASIT	"Failures and outages in the power system"
FCM	Fuzzy C-means
FDI	False Data Injection
FFT	Fast Fourier Transform
FFW-CRBMs	Factored Four-Way Conditional Restricted Boltzmann Machines
FMM	Finite mixture model
GMM	Gaussian Mixture Model
GPRS	General Packet Radio Service
GSP	Graph Signal Processing
HLPC	Hybrid Load Profile Clustering
HMM	Hidden Markov models
HnMM	Hidden non-Markovian Models
HSID	Hybrid Signature-based Iterative Disaggregation
HVAC	Heating, Ventilation and Air Conditioning
ICA	Independent Component Analysis

Abbreviation	Explanation
IDEStream	Incremental density-based ensemble clustering
JSD	Jensen-Shannon divergence
KL index	Krzanowski and Lai index
KM	K-means
KNN	K nearest neighbours
LSTM	Long Short-Term Memory
MAP-PF	Maximum a-posterior particle filter
MIC	Maximum Information Coefficient
MLCDDL	Multi label consistent deep dictionary learning
MLCDTL	Multi label consistent deep transform learning
MLP	Multilayer perceptron
NI	Non-intrusive
NILM	Non-intrusive load monitoring
NMF	Non-negative matrix factorisation
NWD	Not weekday
PAA	Piecewise aggregate approximation
PCA	Principal Component Analysis
PLC	Power Line Communication
PQ	Power quality
RNN	Recurrent Neural Network
RMS	Root mean square
RP	Random Projection
SI	Skewness Index
SMBM	Smart Meter Based Models
SOM	Self-organising map
SVM	Support Vector Machines
SVR	Support Vector Regression
THD	Total harmonic distortion
VAE	Variational Autoencoders
WCBCR	Within Cluster Sum of Squares to Between Cluster Variation
WD	Weekday

1 Introduction

The Norwegian government decided that by 2019, all households should have a smart electricity meter (AMS – advanced metering system) installed. By January 2019, 97% of the households in Norway had smart meters installed [1]. In Norway, no restrictions have been set on the type and make of meter, but there are certain regulations the meter must fulfil [2]:

- Read power consumption with intervals of 60 minutes, and be able to read power consumption with intervals of 15 minutes
- Follow a standard interface to enable communication with external equipment
- Guarantee that no data is lost even during an outage
- Avoid data and load control misuse by providing security solutions
- Meter both active and reactive power (in- and outbound)

The Norwegian DSOs (distribution system operators) had the responsibility of installing smart meters, and the cost is divided between the electricity customers. There are three suppliers of smart meters in Norway; Aidon, Kamstrup and Kaifa [3]. The readings should be stored in the meters and reported to the DSO daily and by 7 am the next day. The smart meters chosen by the Norwegian DSOs also have the possibility to meter voltage.

The advanced functionality of the meters and the large amounts of data transmitted from them have the potential to contribute to more than just a more accurate billing of customers. This report is a result from the ENERGYTICS project, which aims at exploring this potential. The goal of the project is to demonstrate application of Big Data technologies and data science to perform analysis on the smart meter data, and in this way demonstrate an added value both to grid companies and customers. In the start of 2019, the Elhub was operational in Norway, where electricity consumption data is gathered in one large hub and customers can easily access their consumption data and switch between electricity suppliers.

This state-of-the-art includes a description of experiences and technical advances regarding deployment of smart meters in the Scandinavian countries and Finland in chapter 2. Possible benefits and advantages of deploying data science on data from the electricity grid are described in chapter 3. Chapter 4 presents different data sources in the power grid and chapter 5 includes brief explanations of relevant data science techniques for two topics examined in ENERGYTICS – clustering of customers and disaggregation of total electricity consumption into the consumption of individual loads. Chapter 6 describes the methodology for the literature search in this state-of-the-art, and chapter 7 present the results.

2 Deployment of smart electricity meters in neighbouring countries to Norway

This chapter includes experiences from electricity meters in neighbouring countries to Norway, where such meters have been deployed for a longer time than in Norway.

Sweden

Earlier, the electricity bill was based on one reading of the electricity consumption in the previous year. Because this leads to very inaccurate billing, many customers complained. In addition, a cost analysis was performed in the spring of 2002, suggesting possible reductions in energy usage in the range of 1-2% if monthly billing was deployed [4]. Therefore, later in 2002, the Swedish government reacted by requiring that electricity meter readings and billing must be done monthly for all customers by July 2009. The responsibility for achieving monthly readings lies with the DSOs (distribution system operators), but the cost is placed on the end-user [5]. There is no legal requirement to replace the old meters by smart meters, hence some operators have chosen meters that do not support dynamic pricing and giving feedback to the user [5]. In 2012, financial incentives were introduced to encourage investments in smart meters, and customers who subscribed to hourly-based electricity billing contracts received hourly metering for free even if they needed to have a new electricity meter installed [4]. In 2009, Sweden reached 100% penetration of smart meters, and was the first EU country to reach this level [6]. In a report from the European Commission in June 2014 [7], they report energy savings in the order of 1-3% after the deployment of a monthly billing and invoices based on accurate readings, because of reduced energy usage by customers.

Two of the largest DSOs in Sweden, Vattenfall and E.ON Sweden chose to have several contractors delivering the smart meters, while Fortum, another large DSO chose to use only one; Telenor Cinclus [6], [8]. Vattenfall was the first DSO to start the rollout. Because there was no legal requirement on the type of meter, the resulting distribution is that approximately 85% of the meters support reading of hourly values (hence 15% only supports monthly readings), while 41% and 33% is compatible with remote load control and remote disconnection, respectively [6].

The type of communication solutions used also varies. In 2013, 60% of the meters used PLC (Power Line Communication), 20% used GPRS (General Packet Radio Service), and 20% used radio. Vattenfall has mainly used PLC, Fortum uses GPRS, and E.ON Sweden uses both PLC and GPRS [6].

Finland

In Finland, a cost analysis was performed on the potential for demand response and concluded with a positive result given a rollout of smart meters in 2008. Thereafter, the government set a goal to achieve at least 80% penetration rate by the end of 2013 [4]. It is the system operators which has the responsibility of achieving this penetration of hourly metering [9]. It is a requirement that the metering of consumption is done hourly [10], and that data is transferred to the system operator at least daily [9]. The consumption metering must either be cumulative or average load each hour. In addition, start and end times of outages which lasts longer than 3 minutes must be registered by the meters. It is also recommended that the meters measure voltage levels on all three phases, but this is not a requirement [9]. According to [7], the roll-out of smart meters in Finland is complete, and energy savings of 1-3% has been registered due to demand response. Examples of suppliers for the smart meter solutions are Schneider Electric [10] and Landis+Gyr [11].

At the CIRED conference in 2017, the Finnish DSO Elenia Oy presented their experiences with AMS (advanced metering system) and their applications of the system [12]. The company was the first to achieve 100% penetration of smart meters to all their associated customers. The meters themselves and the communication system are from external suppliers, but the system used for data management is property of

Elenia Oy. They also offer a mobile app and a web-based app where customers can see their hourly energy consumption, and it updates once a day. Elenia Oy has also integrated the data collection infrastructure with their Distribution Management System (DMS) to gain a higher level of automatic monitoring of the low voltage grid. This enables better fault management, as the DMS control centre receive alarms from meters if a fault occurs (for example if the voltage is asymmetrical), and then queries can be sent to nearby meters to see if they are affected. The smart meters employed by Elenia Oy can also monitor voltage quality, as they register averaged voltage values every 10 minutes for all three phases, in addition to minimum and maximum voltage values each day.

Denmark

As Denmark has a high penetration of wind energy in their electricity network, it is beneficial to utilise smart meters to be able to synchronise the production and consumption as much as possible.

For customers with a higher energy consumption than 200,000 kWh hourly meter readings have been mandatory since 2003, and for customers with a consumption of more than 100,000 kWh this has been the case since 2005 [4]. In 2013 an economic analysis was performed, and the results showed that a roll-out of smart meters to all customers would create a net annual benefit of DKK 10 million [4]. A complete smart meter roll-out was set to be finished in 2020 and requirements for the meters were decided. The smart meters should be able to register electricity consumption with a resolution of 15 minutes or smaller, to store and transmit data, and have remote control settings for the measuring resolution. It is the DSOs which have the responsibility for the roll-out, and they can adopt any distribution tariff if it does not violate the regulation of allowed revenues for the operator [4]. In 2016, more than 50% of the households in the country had smart meters installed [13].

In Denmark, a Datahub has been created, where electricity consumption data is gathered in one large hub, allowing customers to easier access their consumption data and making switches between electricity suppliers easier [4].

3 Why deploy data science on data from the electricity grid?

There are many benefits one can achieve when deploying data science techniques on data from the power grid, and selected ones are described in this chapter.

More efficient operation of the distribution network

Using modern technology, customers can gain information about their electricity consumption. Raising awareness among customers on their electricity usage patterns can lead to a more efficient use of the capacity of the electricity grid, and it can also enable demand response. Demand response is when customers change their consumption patterns according to different incentives and can be performed either manually by customers or automatically by sensors and timers. There are four types of customer response; the peak loads can be reduced, the electricity consumption can be increased in times of low consumption, the electricity consumption can be moved from periods of high consumption to periods of low consumption, or the total energy consumption can be reduced. If the flexibility of the customers if utilised efficiently, there may be less need for network investments and upgrades. It can also enable integration of renewable power generation, as the consumption can be coordinated with the intermittent power production to a higher extent than without demand response.

In [14], the theoretical potential for demand response both in the residential and industrial sector in Europe is assessed, and the result is that a load reduction between 61 and 172 GW, and a load increase between 68 and 499 GW is available every hour even though the value fluctuates through the year. It is also found that the average load reduction per inhabitant is particularly high in Norway compared to other countries. It is found in [15] that an optimal resolution of smart meter data readings is in the range 21 to 57 minutes when taking an economic perspective and balancing the cost of the meters and the savings by demand response programs. Thus, the resolution of 60 or 15 minutes in smart meters in Norway seems to be an appropriate choice for utilising the opportunities of demand response.

Improved handling and locating of faults and failures

Algorithms can be used to detect and locate earth faults and incorrectly installed electricity meters at customers in an automatic manner.

Monitoring power and voltage quality

Changes in customer's electricity usage patterns, with a higher number of high-power appliances, requires increased monitoring and control of power and voltage quality. Data science and big data techniques can contribute to this, for example in the sense that all smart meters in Norway measure power consumption, while most of them in addition can measure voltage.

Enhancing maintenance schemes and supporting reinvestment decisions

Data science can be used to enhance the maintenance schemes of the distribution grids, so that components are maintained before they are near failure. This can also help in reinvestment decisions, by finding the optimal balance between performing maintenance and reinvestments in the network.

4 Data sources in the power grid

AMS data

As already mentioned in the introduction, the AMS meters deployed in Norway are required by law to measure active and reactive power. In addition, most meters can measure voltage and current. With the introduction of smart meters, substantial amounts of data will be available for the network operators. These data from households can be explored by using data science and contribute to several purposes, as already mentioned in chapter 3.

Historical failure data in the grid

The network operators in Norway are required to register all failures and outages occurring in the grid in a system called FASIT ("failures and outages in the power system")[16]. There are rules for which parameters that should be registered, and national fault and outage statistics are published each year. Datasets with historical failures and outages are therefore available from all network operators, thus the barrier for using these data for analyses is low. However, the datasets are often not very accurate nor extensive. It is for example not common to register the exact location of the event, thus the exact characteristics of the system where the event occurred are often not known.

Network topology and power flow data

Most distribution networks in Norway are operated in a radial manner. To be able to achieve an improved operation of the distribution network, the topology (the hierarchical structure) of the network must in many cases be known. Some models for data analysis include power flow data; how much power that is being transferred at every point in the network.

Power quality (PQ) data

In an ideal network, the electricity supply is an ideal voltage source with zero impedance, thus having a constant voltage regardless of the current. However, different customers have different demands on these parameters, because they have different variation, distortion and fluctuation in the current. The power quality concerns the deviation between the ideal and reality [17]. The power quality is a combination of the voltage quality and the current quality, which are defined by the magnitude, frequency and waveform of the voltage and current respectively.

There are basically two types of power quality disturbances [17]:

- Variations: small deviations from the ideal values, measured at any moment in time
- Events: larger deviations which has a defined start and end time given by when the deviation crosses a threshold. The threshold is set arbitrarily, and therefore the differentiation of variations and events can be difficult

Today's networks are already designed to be able to handle most events, but handling variations is still being researched.

5 Data Science & Machine Learning

Contemporary analytics of smart metering and power quality data is mostly focussed on utilizing techniques from machine learning and statistical analysis. However, the quality of results from machine learning very strongly depends on transforming the data to be represented by the "right" features. As such, there exists a large body of methods aimed at feature generation and reduction. In this chapter, we cover first feature generation & reduction (pre-processing) before giving a wide (although arguably not deep) overview over various machine learning techniques. For more details on most of the techniques outlined here, we refer the reader to the following excellent introductions [18], [19].

5.1 Pre-processing

5.1.1 Feature Generation

For time-series data, directly operating on the raw series (i.e., a sequence of values) may not yield the best possible analytics results. In many cases, we will want to decompose the series into a weighted sum of periodic functions (and then look at the weights) or sets of discrete states. Either procedure generates new features. Of course, new features can also be generated in the time-domain. We now outline several feature generation techniques, sorted into the domain they represent the data in.

Domain	Technique	Description
Time-Domain	Autocorrelation Feature Extraction	This feature extraction is done by calculating the correlation between time steps. Plotting these values can show different structures and trends in the data [20].
	Correlation-Based Feature Selection	From a given set of features, it selects a subset of these features which has high correlation with the response variable, and at the same time has low correlation between themselves [21].
Frequency-Domain	Fourier Transform	The Fourier transform decomposes the signal into a weighted sum of trigonometric functions of different periods. From the weights, we can determine which kind of periodicity is most important in the signal. This is called the power spectrum. For example, when sampling a 50 Hz AC power signal, we would find the largest weight (most power at) at the term corresponding to a sinusoid with a frequency of 50 Hz. The most basic algorithm (Discrete Fourier Transform; DFT) is fairly slow, but fast algorithms exist (FFT) [22]**.** To overcome noise, Fourier transforms are usually windowed and/or averaged [23].
	Wavelet Feature Extraction Wavelet Transformation	The wavelet transformation transforms a signal in the time domain to a domain composed of orthogonal finite waves. It is a transformation which represent both smooth functions and functions with discrete jumps effectively [18]. Wavelet transformation is used to remove coefficients in the data which are not significant [20]. There also exists an alternative Wavelet transformation which is faster to perform, and it is called Fast Wavelet Transformation.

Domain	Technique	Description
State-Based	Markov models	If an element in a time series only depend on the previous element it can be modelled by a Markov chain [24]. States of the time series are modelled, where the states are associated with observable physical events [25].
	Hidden Markov models (HMM)	An HMM is a Markov model where the states of the time series are not observable, but rather that the observations are functions of the states [25].
	Jump Models	A way to model the states of a time series (for example load signatures of household appliances) given that the time series rarely change state over two successive time samples [26].

5.1.2 Feature Reduction (Dimensionality Reduction)

There exist several techniques for reducing the dimensionality of data by extracting the most important features from a dataset. Feature reduction is used to speed up computation during analysis as well as aiding the human analyst during interpretation and communication to stakeholders.

Technique	Description
Independent Component Analysis (ICA)	Signal processing technique which aims at expressing random variables as linear combinations of statistically independent component variables [27].
Principal Component Analysis (PCA)	This technique aims at representing a dataset containing many correlated variables with a dataset with a smaller number of features that explain most of the variability in the original dataset [19]. The dimensions in the new dataset are linear combinations of the original features. Kernel PCA is an expansion of PCA where the original features are expanded as if by non-linear transformations and then PCA is performed [18]. PCA is special case of ICA with orthogonal basis vectors.
Karhunen Loève Transform	An algorithm to perform noise filtering which can identify signals in all types of noise [28]. Also related to PCA and ICA but operates on the correlation matrix instead of the covariance matrix (like PCA and ICA do).[1]
Sparse Coding	A type of unsupervised neural network which extract basic elements from data [29]. The method is inspired by mechanisms of neuron networks in the brain and assume that a vector can be represented by a linear combination of basic vectors [30].
Random Projection (RP)	Dimensionality reduction by RP uses the statement that the scaled pairwise distances between points are preserved if a higher dimensional dataset is projected to a lower dimension by using a subspace which is random. The projection is normally done using a random matrix which have elements that are Gaussian distributed, but other distributions are also used [31].

[1] The correlation matrix is a scaled version of the covariance matrix.

Technique	Description
Synthesis and analysis sparse coding/Dictionary learning and transform learning	We distinguish between two types of sparse coding; synthesis and analysis. In synthesis sparse coding, a dictionary and sparse coefficients are learnt given some data, which together can reconstruct the data. In analysis sparse coding, a signal is analysed to obtain a clean version of the data and a dictionary, which together can give the sparse coefficients [32].
K-SVD	K-SVD is an iterative algorithm, and an iteration consists of two steps. First, sparse coding is used to represent a vector by a dictionary of basic vectors, and then the basic vectors in the dictionary are updated to minimise the reconstruction error (the error when transforming the vectors back to their original form) [30].
Symbolic Aggregate Approximation	Used for dimension reduction of numeric time series where time series are discretised to symbolic strings [24]. This is done in two steps. First, the time series are transformed to a piecewise aggregate approximation (PAA) representation by replacing the amplitude values in a time interval with their mean. Then, a mapping from a PAA to a word is done by assigning a letter to each amplitude level.

5.1.3 Graph Signal Processing (GSP)

Represents a dataset by a discrete signal indexed by a set of nodes and a weighted adjacency matrix. The nodes represent an element in the dataset, while the adjacency matrix represents the edges in the dataset and the correlation between nodes (also called weights). Figure 5.1 shows an example of creating a graph representation of a times series with power consumption data. The times series signal is shown in the upper plot, showing events that are detected, which are when the power is increasing or decreasing with a value above a set threshold. All these rising and falling edges are in the middle plot represented by nodes. The adjacency matrix will in this case contain the distance between each of the nodes. The bottom plot in Figure 5.1 shows the first step of clustering the power edges, where edges similar to the first one (the one to the left) are marked as red. This way all edges can be clustered.

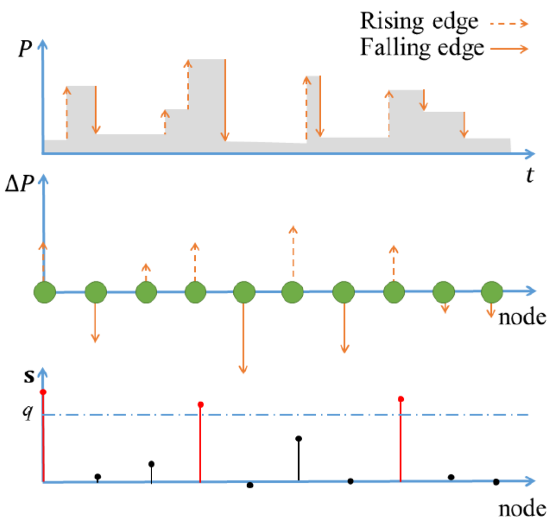

Figure 5.1 Generation of a graph representation of power consumption data

5.2 Machine Learning

Broadly speaking, machine learning can be broken down into the areas of classification, regression, and clustering. In classification and regression, an algorithm learns the relation between features and labels from known data, such that it can propose the label for unknown data in the future. This is called supervised learning. In clustering, algorithms attempt to group similar datapoints. This is called unsupervised learning because it does not rely on any kind of label. Frequently, dimensionality reduction techniques discussed above are also considered unsupervised learning methods. Figure 5.2 summarizes some of the available methods in each of these groups. We now discuss some of these methods along with some technical details. A machine learning technique that has received a lot of attention recently is neural networks. As such, we have decided to devote them a separate section (section 5.2.4), but note that they are basically yet another technique that can be applied to classification and regression problems.

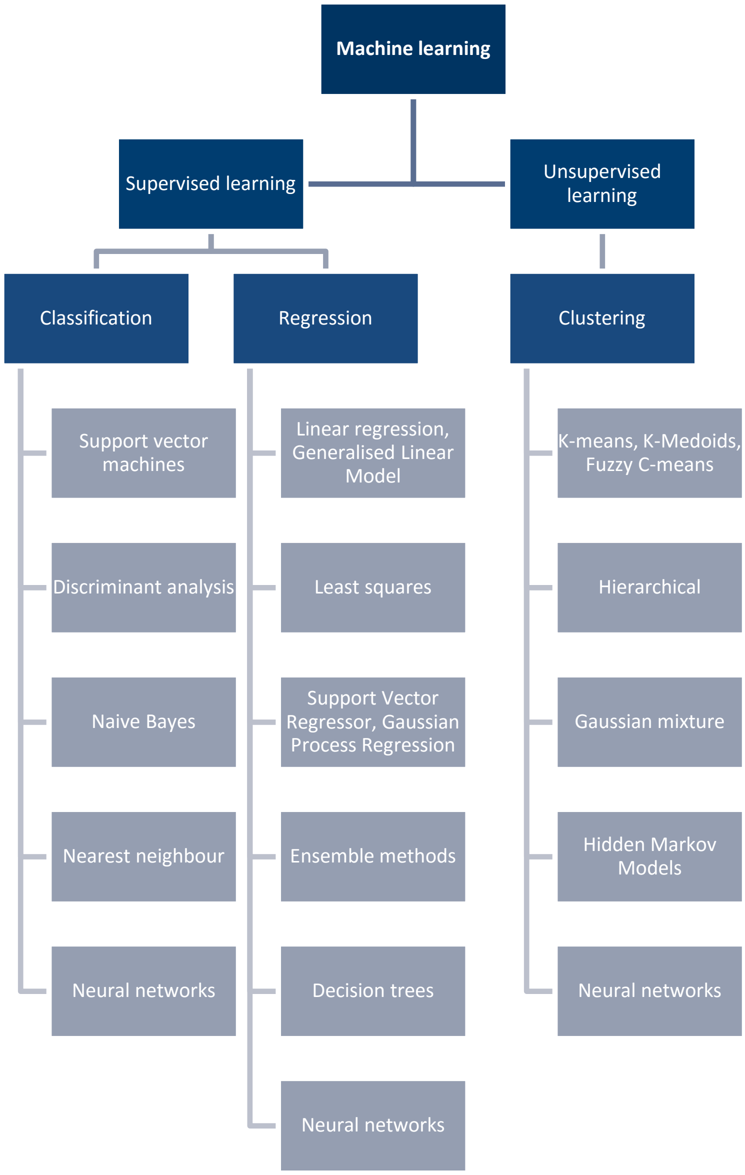

Figure 5.2 Overview of machine learning techniques. Inspired by figure in [33]

5.2.1 Supervised Learning: Classification

In classification, the objective is to assign data points or observations to distinctive classes.

Technique	Description
Bayes classifier	The Bayes classifier assigns observations to the most likely class. The most likely class is decided by the conditional probability that the observation belongs to a class when its input values are given [19]. A Naïve Bayes classifier assumes the observations are independent [34].
Decision/classification trees	Partition the possible values for an observation into smaller and simpler regions. A given observation is classified to the class which is most common in the training data in the region the observation lies [19].
Support Vector Machines (SVM)	Kernel methods are a group of classification methods in machine learning, and support vector machines (SVM) is maybe the most widely used algorithm in this group. The method works by finding optimal decision boundaries between sets of points belonging to two distinct categories. When training the model, it looks for a line or surface separating the training data into two spaces. To classify new data the model draws its conclusion based on which side of the decision boundary the data point is. The boundary is found during two steps [34]: 1. Mapping the data to a representation of high dimension where the decision boundary can be formulated as a hyperplane (a plane which can be expressed by one linear equation, for example a straight line if the data is originally two-dimensional). This can often be a difficult task due to complicated calculations, and hence a kernel function is used. Such functions can map any pair of data points in the original space to the high-dimensional space without explicitly calculating their new coordinates, but rather only using the distance between the two data points in the new space. Kernel functions are usually constructed manually and not learned from the data 2. Computing the decision boundary by maximising the distance between the boundary and all the data points from each category which are closest to the boundary A drawback with SVM appear when using it on large datasets. To deploy the method in complex problems, the input data must be presented in useful representations, which are found manually. This step is called feature engineering, is often difficult and time-consuming, and requires domain-knowledge of the data and its source.

5.2.2 Supervised Learning: Regression

In regression, the objective is to learn the relation between observations and a continuous target. Fitting a line through a set of points is the simplest example.

Technique	Description
Least squares	From a set of input variables an output variable can be represented by a set of equations (linear or non-linear). In the method of least squares a set of equations with more unknowns than number of equations is solved to minimise the sum of squared residuals, residuals being the error between predicted and actual values of the output variables [18]. In a *weighted* least squares problem, not all data points are of equal importance, and the weights are found so that each data point is given the correct influence on the parameters [35].
Regression trees	Partition the possible values for an observation into smaller and simpler regions. The response of a given variable to an observation is predicted to be the mean of the responses of the training data for the region the observation lies in [19].
Support Vector Regression (SVR)	A type of Support Vector Machine used for regression, see chapter 5.2.1

5.2.3 Unsupervised Learning: Clustering

Clustering attempts to identify groups of neighbouring data points.

Technique	Description
K-means (KM)	KM takes all data points as one big cluster as input, and then divides them into a predetermined number k of smaller clusters. The algorithm finds centroids (the averages) of each cluster, and then reclusters the data points according to the nearest centroid based on a distance calculation between the data points. The last two steps are repeated until convergence [18]. Different measures of distance between time series can be used [18].
Mini-batch K-means	A version of K-means which is more suitable for streaming data. For each iteration it divides the data randomly into smaller batches of the same size and tries to minimise the same objective function as the regular K-means algorithm [36].
K-median	Data points are assigned to clusters so that the distance between the data point and the median of all data points in the cluster is minimised [37].
K-medoid	This clustering method assigns data points to groups so that the distance between data points in the same group is minimised and at the same time so that the distance between clusters is maximised [38].

Technique	Description
K nearest neighbours (KNN)	This method decides the cluster of a data point by using the k nearest training points [18]. The data point will be classified to the cluster in which the majority of the k nearest training points are.
K-shape	Considers the shape of the time series to a larger extent than k-means because in opposition to k-means, k-shape does not treat observations in a time series as independent. The method uses a normalised type of cross-correlation as distance measure and derives a shape-based distance measure. Cluster centroids are calculated, and these are used in the iterations to update which data points belong to which clusters [39].
Fuzzy c-means (FCM)	In the FCM clustering algorithm, data points are considered to belong to all clusters, but with different degrees [40]. Initially, a guess is made for the cluster centres and then each data point is given a random degree of belonging to every cluster. Finally, an iterative process updates the positions of the cluster centres and the belonging of the data points to each cluster.
G-means	This method is like k-means but calculates the optimal number of clusters and make less assumptions about the spatial distribution of the data points within clusters. G-means is (like k-means) iterative but increase the number of clusters in each iteration by splitting clusters whose data points are not from a Gaussian distribution. Between every increase of clusters, k-means is run on the resulting cluster centres. The only parameter which needs to be specified beforehand is the significance level of the test whether clusters are Gaussian.
Constriction Factor Particle Swarm Optimisation (CF-PSO)	Particle swarm optimisation is an evolutionary optimisation technique and is inspired by social behaviour. The system to solve is initialised with a population of random solutions and each potential solution (called particles) also get a random velocity assigned to it and then the particles are "flown" through the problem space [41]. To ensure convergence of the optimisation a constriction factor can be used, which is a value that controls the velocities when there is a fast change in position of the particles [42].
Density-Based Spatial Clustering of Applications with Noise (DBSCAN)	A density-based spatial clustering algorithm [43]. For time series, clustering is performed for each time step. Advantages with the algorithm is that it can identify irregularly shaped clusters, distinguish noise from data, and the number of clusters need not be predefined.
Fast search and find of density peaks (CFSFD)	This clustering method assumes that cluster centres have a higher local density and that their distance to other points with high density is relatively large [24]. Neighbours to a point is defined by either a soft or a hard threshold. The minimum distance between a data point and other data points with high density is defined so that the point with highest density has the highest minimum distance. The points with highest density and minimum distance to other high-density points are called cluster centres, and then the rest of the data points are assigned to cluster centres. An advantage with this method is that assigning points to clusters do not need any iterations.

Technique	Description
Finite mixture model (FMM) clustering	In this method it is assumed that measurements can be described by a finite mixture of distributions [44]. Then the expectation-maximisation algorithm can be used, which finds a local maximum by iterating between two steps. First, the expectation of the log-likelihood is calculated to update the probabilities of each data point to belong to a cluster. Then, this expected log-likelihood is maximised to update mixing proportions and distribution parameters. When convergence occur, either the data points are assigned to a cluster with a probability calculated by all final probabilities or they are assigned to the cluster with the largest probability. Advantages are that it can model a mixture of continuous and categorical data and that it does not need a predefined distance measure. Disadvantages are that the number of clusters must be defined beforehand and that it is sensitive to chosen attributes.
Gaussian Mixture Model (GMM) clustering	In GMM clustering, it is assumed that the data points consists of a mix of a finite number of Gaussian distributions [45]. In opposition to k-means which assigns every point to a cluster which has a constant radius, GMM gives the probability that a point belongs to clusters which can have an oblong shape [46].
Mean-Shift clustering algorithm	Each of the data points are shifted to the mean of their neighbouring data points. K-means is a special case of mean-shift clustering [47].
Self-organising map (SOM)	K points are first initialised to lie in a one- or two-dimensional plane [18]. The SOM method then iterate through all data points to be clustered and curve the plane so that the initial points fit the data points as good as possible. Finally, the data points can be mapped to the one- or two-dimensional plane.
Hierarchical clustering	Hierarchical clustering techniques work in levels, where the highest level is one cluster which contains all data points and at the lowest level each cluster contains one data point. In contrast to the K-means method, these techniques do not need a specification of the number of clusters. However, what is needed is a specification of what measure to use for pairwise dissimilarity between data points in two groups. There are two types of hierarchical techniques [18]: 1. Agglomerative (bottom-up strategy): starts at the lowest level and in each iteration, it merges two clusters. The algorithm selects the two clusters with lowest dissimilarity between them 2. Divisive (top-down strategy): starts at the highest level and in each iteration splits one cluster into two. The algorithm chooses a cluster to split which gives two clusters with highest dissimilarity between them The algorithms can work until they reach the highest or lowest level, and the level which represents the optimal clustering must be chosen manually. There exist several statistic measures which can be calculated for each level and help in the decision process.

5.2.3.1 Unsupervised Learning: Performance Metrics for Clustering

In clustering, performance metrics are used to quantify the performance of the clustering.

Technique	Description
Bayesian Information Criterion (BIC)	The BIC is a function of sum of squared error between the data points and the centre of the cluster they have been assigned to [48].
Bootstrapping	The clustering is repeated many times with different realisations of the data, and the uncertainty of the classification is calculated. In addition, a measure for how well separated the classes are can be calculated.
Context-independent, Optimality, and Partiality (COP) index	Is the ratio between the intra-cluster variance (defined as the mean DTW distance between the data points in a cluster to the cluster's most centrally located point) and the inter-cluster variance (defined as the DTW distance between the data points furthest away from each other in a cluster) [38].
Davies Bouldin-index	Calculates the performance of a clustering by taking into account the dispersion between clusters and the so-called Minowski metric of the centroids of the clusters [49].
Krzanowski and Lai (KL) index	A function of a parameter q and the within-group dispersion, where the value of q giving the highest value of the KL index stipulates the optimal number of clusters [50].
Scatter-Distance index	Based on the average scattering within clusters and separation between the clusters [51].
Within Cluster Sum of Squares to Between Cluster Variation (WCBCR)	Defined as the ratio between the sum of distances between points in a cluster and the sum of distances between cluster centres [52].
Xie Beni index	Based on distance between cluster centres and the average compactness and separation [53].

5.2.3.2 Unsupervised Learning: Distance Measures for Clustering

There exist a wide range of methods to the defined and calculate distance between data points (in the time, frequency, or state-based domains). Distance metrics are important for clustering as well as for identifying how well data is aligned in time.

Technique	Description
Correlation-based distance	Focuses on the similarity of the shape of the load profiles, and two observations are considered similar if if their features are correlated [19].
Cosine similarity	Based on the normalised dot product between two observations [54].

Technique	Description
Dynamic Time Warping (DTW)	Similar to Euclidean distance in that one take the square root of sum of squares, but the two data points that are being compared need not be from the same time step [55]. The aim of the method is to compare data points from time steps which overall gives the best result.
Euclidean Distance	The straight-line-distance between two points [18].
Hausdorff distance	The Hausdorff distance between a point and a closed subset of data points is the minimum of the distances between x and the points in the subset. The Haudsorff distance between two subsets are defined as the maximum out of all the possible distances between points in the two subsets [56].
Maximum Information Coefficient (MIC)	Based on calculations of the mutual information between two variables. The value of MIC is between 0 and 1 and the closer to 1, the better correlation.
Pearson Correlation Coefficient	Gives a number between -1 and 1 which describes the degree of linear relationship between two time series [56].

5.2.4 Machine Learning: Neural Networks

Neural networks are built up by combining many (dozens or even hundreds) artifical neurons (frequently called "perceptrons") into different layers, cf. Figure 5.3. In a given layer, neurons receive the output from the preceding layer as input, compute a weighted sum of inputs, which is then fed to the next layer. By presenting the network with training data (pairs of inputs and outputs to the entire network) and adopting a training procedure, the weights of each neuron are iteratively adjusted such that the network learns to map inputs to outputs (supervised learning).

We describe different network architectures as well as training schemes.

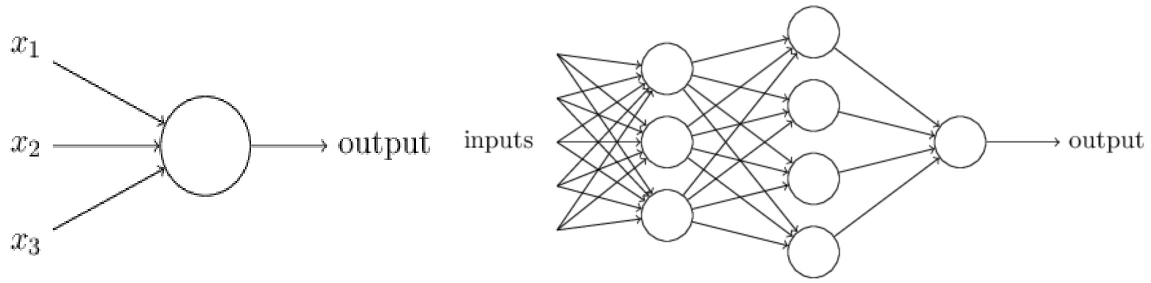

Figure 5.3 Illustration of a single preceptron (*left*) as well as a simple multilayer perceptron (*right*). The network on the right is a feedforward neural network with two hidden layers. Both illustrations are taken from chapter 1 in [57]

5.2.4.1 Architectures

There is a wide range of neural network architectures. Typically, particular architectures are geared towards processing a particular type of data structure. For example, CNNs (convolutional neural networks) are most suitable for image or voice processing, RNNs (recurrent neural networks) for sequence processing, and vanilla MLPs (multilayer perceptron) for generic supervised learning tasks.

Technique	Description
Feedforward neural network	Neural networks which process the whole sequence of input data, and not just as separate portions of data which are processed individually [34]. Feedforward architectures are contrasted by recurrent architectures such as RNNs (and their extensions such as LSTMs)
Multilayer perceptron (MLP)	A feedforward neural network where the input data goes from the input layer to the output layer in one direction, using a backpropagation learning algorithm [58]. It can solve problems which are not linearly separable [59].
Convolutional Neural Network (CNN)	CNNs consists of convolution layers, which apply a geometric transformation to batches of the input data. These layers look at portions of data, and therefore they often have better performance on data which is not sequential (thus where the order of the data is not important) [34].
Recurrent Neural Network (RNN)	RNNs consists of recurrent layers which process whole sequences of data in a sequential way. It "remembers" previous input data by storing a state based on information processed previously in a sequence of data, but reset its state between every sequence it processes [34].
Long Short-Term Memory (LSTM)	LSTM is a widely used recurrent layer which can transfer information across many timesteps, thus remembering information from many timesteps before [34].
Restricted Boltzmann Machines	Restricted Boltzmann Machines are often used as basic blocks of deep learning models. They consist of two layers; one visible and one hidden [60].
Variational Autoencoders (VAE)	An encoder is a module which learns low-dimensional representations of input data, from which a decoder can sample a point and create a realistic output [34]. VAEs consists of an encoder and a decoder and aims at reconstructing an output from the input data [61].
Denoising autoencoder (DAE)	DAEs also consists of a decoder and an encoder and are often used for feature extraction of data. The input is corrupted by randomly setting some of the input values to zero, and then the DAEs try to reconstruct the original input data [62].

5.2.4.2 Training Schemes

Given a set of inputs and expected outputs, we can derive the weights of individual neurons by optimizing a loss function. Given a set of weights, inputs, and outputs, the loss function describes the difference between the output that a given input of the neural network actually returns and the output that would be expected. By optimizing this function towards zero, suitable weights are derived.

Technique	Description
Backpropagation	Neural networks which use the backpropagation algorithm for adjusting the network after processing batches of the data, to obtain a better performance. Backpropagation starts with the loss value (the difference between the model output and the target output) and works backwards through the layer, calculating the contribution of each parameter on the loss value [34].
Extreme Learning	This is a supervised learning algorithm for feedforward neural networks with only one hidden layer. It chooses hidden nodes randomly and determines the output weights of the network analytically by finding a least-square solution [63], [64]. Because of this, the training is not iterative, and is hence faster than conventional gradient-based learning of feedforward neural networks [65].

6 Methodology for literature review

In this article, searches in literature databases are done for different areas of interest regarding smart electricity meters and applications of big data techniques. The work has focused on two areas of data science applied on data available to grid operators: clustering of AMS data or other grid data and load disaggregation of AMS data.

The literature search was done in the engineering database EI Compendex, including journal and conference papers. The search words for the state-of-the-art of clustering and disaggregation was as follows:

- ("smart meter*" OR AMS OR AMI) AND (clustering) AND (electricity OR power)
- ("smart meter*" OR AMS OR AMI) AND (disaggregation OR NILM) AND (electricity OR power)

Due to a very large number of search results, the results were restricted to papers with these search words in the title, abstract or subject. The search for clustering then resulted in 180 papers and the search for disaggregation resulted in 81 papers. The abstracts were read for all these papers, and papers found relevant were included in the state-of-the-art.

7 Results from literature review

7.1 Clustering of customers

Today, households are often divided into various categories, but these are based on characteristics about the household (e.g. number of persons, size of fuse etc). This is not guaranteed to separate households with different electricity consumption habits. There exists a rich literature on different methods for clustering customers and on using clustering to improve other applications, for example improving forecasting, detection of cyber-attacks, and demand response.

Several papers focus on improving existing clustering methods. In [66], an extended k-means method is proposed, which examine the intra-cluster similarity of the clustered energy consumption patterns. If an initial clustering using k-means give clusters which do not have a high such similarity they are fed back for re-clustering. The process is repeated until intra-cluster similarity is high enough. A similar extended k-means is proposed in [67], where the process of re-clustering is repeated until no data points change cluster. In [68], a clustering method which focus more on the shape of the load profiles is proposed, and it uses DTW-based distance.

There is a rich literature on feature extraction and dimensionality reduction of data before clustering is performed. Two techniques are compared in [31], namely Random Projection (RP) and Principal Component Analysis (PCA). The mean and standard deviation of the relative error for the Euclidean distance between datapoints in the reduced data and the original data is compared for the two methods. Both the original data and reduced data is clustered using k-means and the clustering performance is compared. The result is that the PCA performs better except for smaller dimensions.

The authors of [69] use kernel PCA analysis for feature extraction before using k-means clustering. Normalisation, wavelet transformation and autocorrelation feature extraction are used in [20] to account for temporal components (for example trends, oscillations and repeating patterns) in the data before clustering using k-means. Electricity consumption data from factories are clustered in [70] and a feature selection is performed by exploiting the evolution-based characteristics of the data. Then, k-means is used for clustering, and to estimate the number of clusters data visualisation of the clusters (in the dimensions of two principle components) is done for different numbers of clusters.

Another paper, [30], use K-SVD to decompose load profiles from businesses into linear combinations of different partial usage patterns, compressing the data and extracting hidden consumption patterns. Then use the extracted patterns and an SVM-based method to classify the load profiles into two groups (residential and small/medium sized businesses). [24] focus on the behaviour dynamics (here defined as transitions between consumption levels in adjacent periods) instead of the shape of the load curves. Symbolic aggregate approximation is applied for dimension reduction of the dataset and time-based Markov model is applied to model the dynamics of the consumption (and not the shape of daily profiles) and reduces the dataset to a matrix of state transitions. The clustering technique used is clustering by fast search and find of density peaks (CFSFD), which cluster based on density. Because the data amount is big, k-means is first applied to obtain clusters of customers at local sites and then CFSFD is used at global sites.

Three different methods for constructing features of daily electricity load curves are proposed in [71]: 1) use prior knowledge of aggregated daily consumption in the dataset to create new time-resolution-based features from existing features (for example minimum consumption from midnight to 2 PM and its corresponding time), 2) calibration and normalisation of the features created in method 1) where calibration is done based on if the households are connected to the gas network in addition to electricity network, and 3) clustering based on profile errors (mean absolute percentage error between profile estimate and actual consumption). All methods achieve better results than using k-means with default features.

Calculating features from the original data has also been done in literature. Hierarchical clustering and self-organising map (SOM) is used in [72] to cluster consumers, and in addition to raw consumption data the consumption indicators *responsibility factor* (how much each customer is contributing to the peak load for all customers aggregated) and *consumption variability* are used as input data. Both gas and electricity smart meter data are used in [73], where k-means is used to cluster the customers according to the sum of the squared distances between load profiles. The number of clusters is decided by visual model selection.

In [44], each day is divided into four periods and different attributes are calculated (for example average power, standard deviation, difference between weekday and weekend, etc.) for each period. Finite mixture model-based clustering is performed where the number of clusters is decided using Bayesian Information Criterion (BIC). An existing bootstrap technique is used to determine whether the clustering is reliable. Also in [74] the day is divided into four periods. Relative standard deviation increases in aggregate demand during winter and average aggregate demand during summer (where the lowest consumption was registered) are chosen as attributes.

Several papers group the daily load profiles according to type of day. Small and medium sized commercial buildings are clustered in [75] to compare their energy performance. The daily consumption is first grouped according to season and workday/holiday before normalised load curves are clustered using k-means. Unsupervised classification based on a constrained Gaussian mixture model (constrained in the way that the parameters vary according to the type of day (weekday, Saturday or Sunday)) is proposed in [76]. The results from the clustering is then compared to known socio-economic characteristics of the households. Also, variations of the clusters over time is examined. In [52] the data is partitioned into whether it is a weekday (WD) or not (NWD) and transformed to the frequency domain using Fast Fourier Transform. K-means is then performed on WD and NWD data separately, and the number of clusters is chosen based on Within Cluster Sum of Squares to Between Cluster Variation (WCBCR).

Hourly electricity consumption data from households and data from interviews with the households (number of adults and children, average age of adults, floor size, education, employment and income) are used in [77]. First calculate average consumption in each hour over typical days based on season and day (weekday or not) and normalise the result for each household before k-means is used for clustering. The authors of [78] cluster customers based on demand signature, which is a synthetisation of the daily load curves where only the most relevant information is contained. The measurement for selecting the most relevant information is called the fitness, and an evolutionary approach is used to select the most relevant information.

Some papers use other data in addition to electricity consumption data. Both temperature data and smart meter data are used in [79]. A Hidden Markov Model first learns the dynamic behaviour of each customer under different temperature environments then k-means clustering is performed to find typical load profiles. In [80], k-means is used to cluster real power and total harmonic distortion (THD) load data independently. To cluster on the combination of real power and THD load, fuzzy logic is used.

Dynamic clustering of hourly residential consumption is done in [56], where the patterns of a cluster depends on profiles which evolves in time. It has been decided a fixed number of 10 clusters, based on previous research. The framework for dynamic clustering used is called Equal N-Dimensional (END), where different methods for static clustering has been modified and tried. The static methods in this article is K-means (KM) and Fuzzy C-means (FCM). Further, three different measures of similarity are used for each method; Euclidean distance, Pearson correlation coefficient, and Hausdorff distance. To compare the algorithms, the following indices are calculated; Davies-Bouldin index, Scatter-Distance index, and Xie-Beni index. The conclusion is that the methods using Euclidean and Hausdorff distances has the best performance.

Two papers use density-based clustering. The first one [43] use the density-based algorithm DBSCAN for finding typical daily load profiles for each customer, and in addition a correction strategy is performed on the results to eliminate distortion. Then, k-means is used on the resulting typical profiles for each customer to find typical load profiles among all the customers. They propose an evaluation function for clustering effect, which is applied to determine the optimal number of clusters. All customers are characterised by a set of indices, for example average load rate during the peaks of the daily profile, ratio of average peak load to average valley load etc, and then each cluster achieves a range for each index. This enables the cluster of an individual customer to be easily determined. The second paper propose a method called incremental density-based ensemble clustering (IDEStream) to cluster real-time data streams [81]. Its aim is to group factories based on their electricity consumption data, and it first uses a gamma mixture model to eliminate outliers (assume the data is a mixture of gamma distributions like finite mixture model clustering explained in chapter 5.2.3 to identify dense values and dissolve sparse values). Then they use k-means on data in a given time window. The number of clusters is found by using the KL index.

Several papers use a two-step clustering. In [82], clustering is first performed on the load data of each customer separately to find typical load profiles for each of them. Then, the typical load profiles are clustered. Use an adapted weighted fuzzy C-means as clustering algorithm and PCA for dimensionality reduction. The same two-step procedure is followed in [83], but with g-means as clustering algorithm and fast wavelet transformation to reduce the number of features for the clustering algorithm to perform on. In [84] they first use unsupervised clustering on daily load curves (dimension reduction by 1D Discrete Wavelet Transform and clustering by k-means) to extract typical consumption behaviours of each consumer. Secondly, typical load patterns of different consumers are clustered using unsupervised clustering (novel algorithms proposed in the paper) to identify different consumer categories. Also, the consumption characteristics of each group is found by looking at which consumers are in which groups and select the customers which are in the same groups and assigning them to the same consumer category. Finally use supervised clustering (k nearest neighbours) to classify typical load patterns of new consumers into one of the groups from the second clustering.

An algorithm for clustering the power consumption time series of one customer into variable length segments of data is presented in [85] which is based on Jensen-Shannon divergence (JSD, a similarity measure for discrete probability distributions [86]). The method uses sliding window and detection of change points to make the algorithm adaptive and automatic. Descriptions of the segments based on parameters from descriptive statistics (mean, standard deviation, etc.) are created for clustering. Finally, clustering is performed using self-organising map clustering. Another new algorithm, called Hybrid Load Profile Clustering (HLPC), is proposed in [87]. The data is first pre-clustered according to the Skewness Index (SI) (sensitive score to quantify the spike amplitude in the data) into three groups; "spike" (SI above a threshold K_l), "fluctuation" (SI below a threshold K_u) and "undetermined" (SI between K_u and K_l). The similarity of the load profiles in the first two groups is estimated to cluster them further and finally, hierarchical clustering is performed on the two groups. The profiles in the "undetermined" group are afterwards estimated to either be in a cluster in the "spike" group or a cluster in the "fluctuating group".

Forecasting

Several papers use clustering to improve load forecasting. [88] proposes a new clustering method in which they cluster the electricity consumption of the individual consumers before forecasting the load for each cluster. [89] and [90] use k-means to cluster customers, taking into account different day types (workdays, holidays, etc.) to improve short-term forecasting. In [91], forecasting of aggregated load is improved by clustering using k-means, and the number of clusters is decided based on the forecast performance. Improvement of forecasting is also the aim of clustering in [39] and [92], where the former uses k-shape and the latter fuzzy C-means based on PCA.

In another paper consumption data is normalised (where variables used for normalising are stored) and dimensionality reduction by using a multiple linear regression model is performed before k-means is used on the new representations [93]. The centroids from the clustering are used to train a forecasting method and finally, the forecasts are denormalised using the stored values. The authors of [55] cluster smart meter data by using K-means and use dynamic time warping as distance measure between time series. Then the load probability distribution in every cluster is examined to make stochastic models of customers, and the results are used for spatial load forecasting. In [94], weekly profiles of 5 weekdays from customers under a feeder are clustered by the use of k-means. Several metrics are used to decide an optimal k. The result is used to determine how many customers under the distribution transformer needs to be monitored in real-time to be able to estimate the total load at the distribution transformer in real-time.

Three different methods for feature selection in the clustering are performed and compared in [95]: regression coefficients, average daily load pattern, and full load pattern. Regression coefficients together with normalisation gave the best results. In [21] approximately 800 features are created from historical data (electricity consumption, day of week, hour of day, temperature and humidity data) and then Correlation-based Feature Selection are performed on these features for each household. Different clustering techniques are defined and compared: Max-AC (maximise autocorrelation of the clusters' consumption), Min-Stdev (minimise fluctuations in the clusters' consumption), Max-Sim (maximise similarity among customers in a cluster), and Random (randomly assign customers to clusters).

In [96], the aim is to forecast highly variable loads. The data is first clustered using extended k-means, and the load profiles within each cluster is aggregated to create a representative profile for each cluster. To deal with the issue of variability, the resulting clustered loads are smoothed using a moving average, and non-linear patterns are linearised using Taylor series linearisation.

Clustering, classification and forecast is the focus of [97]. Input data (smart meter data, weather and temporal variables) is cleaned and pre-processed in the form of daily profiles. The data is represented in three different ways: raw daily profiles, statistical parameters representing five time periods each day, and by PCA. Two clustering techniques are tried: k-means (number of clusters decided according to silhouette and Davies-Bouldin indices) and self-organising maps. After the clustering, the results are used in two ways:

1. To understand if type of day or weather variables influence the type of load profiles of a household by using a Classification and Regression Tree method
2. To build Smart Meter Based Models (SMBM) to forecast electricity usage

Detecting electricity theft

There are examples in literature where clustering is used to detect electricity theft. The algorithm in [98] first estimates the technical loss in transmission lines. Then, dimensionality reduction is done on smart meter data from the customers and k-means is performed. Synthetic data samples which contain energy theft are created, and an SVM classifier is trained on annotated data (contains energy theft or not). Detection of electricity theft by using density-based clustering is done in [99]. They use a new theory called densityClust, which opposed to traditional density-clustering does not need a defined radius of neighbourhood and density threshold. Two techniques for detection of electricity theft are combined in [100]. The first is Maximum Information Coefficient (MIC), which is used to find correlations between non-technical loss and different behaviours of the customers and hence can find thefts that appear as a normal load shape. The second is clustering by fast search and find of density peaks (CFSFDP), which find abnormal load shapes among many customers.

Cyber-attacks

Detection of cyber-attacks can also benefit from clustering. In [36], mini-batch k-means clustering algorithm is used on real-time streaming smart meter data to discover anomalous traffic caused by intrusion attacks to the AMI system. The model learns and updates the clusters in real time, and groups which are very small are labelled anomalous. A new Gaussian-Mixture Model-based detection scheme is proposed in [101], which cluster historical data and learn the minimum and maximum values of the clusters to reduce the range of data considered normal.

An improved Constriction Factor Particle Swarm Optimisation (CF-PSO) based hybrid clustering method is used in [102] to group nodes into three groups according to how vulnerable they are to False Data Injection (FDI) attacks. Detection of false data injection attacks is also the focus in [103], where data from trusted nodes (transmitting encrypted data and authenticated identity) are used to detect abnormal activity in other nodes. First, k-means is used to find buildings with similar consumption, and then the consumption of the other nodes is predicted using measurements from trusted meters, and if the actual and predicted values are different, it can indicate an attack.

Demand response

Clustering can also be used for optimising demand response. Smart meter data from households is clustered by sparse coding (a type of neural network) in [29]. To extract characteristics of fluctuating loads, the data was normalised to remove fixed loads, and the result of the clustering is used to achieve valuable demand response. In [38], normalised daily load profiles are clustered to target customers for demand response. k-medoid is used with a dynamic time warping distance measure. To find the optimal number of clusters, the COP index cluster validation technique is used.

Pricing strategies

A couple of articles was found which aims to design pricing strategies. The first one is [104], where clustering of customers in microgrids is done to find optimal real-time pricing strategies. An improved k-means called weighted fuzzy average k-means is used. A fuzzy average calculates the centroid of the clusters not by regular average but places it among more densely situated points. The second one classifies residential electricity customers based on hourly AMI metered values [64] by using k-means, PCA, and extreme learning machine. The resulting typical load profiles are used to design individualised electricity price designs for each group. The work in [105] cluster based on a so-called load index to verify appropriate tariff structures for residential customers. The load index is calculated using the maximum distance between the centroid of a cluster and the elements in the cluster. The electricity consumption for a group of customers metered in Ampere is used for clustering, and the hours of the day are clustered into peak or off peak for the group of customers.

Confirm topology, phase connectivity or voltage sensitivities

In [106] voltage data is clustered using k-means to verify and correct the existing connectivity model of a distribution network. The data is transformed into two datasets in addition to the original: continuous fluctuations and discretised fluctuations. Three different distance metrics are tried: Euclidean, Cosine and correlation. Clustering on discretised data performs best, while the results with different distance metrics are similar. The results can be used to classify customers under different feeders if they are powered by different substation transformers.

Two papers were found on using clustering on smart meter data to confirm the phase connection of customers. One cluster voltage magnitude time-series by using K-means and Gaussian Mixture Model [107], and another proposes an improved constrained k-means clustering technique, which considers the structure of the LV distribution networks without knowing the topology of the network beforehand [108].

In [109], customers are clustered by using hierarchical clustering and the result is used to estimate bus voltage sensitivities to power changes. The number of unknown parameters is reduced compared to the case without clustering, because only one voltage needs to be selected for each cluster of customers. The voltage closest to the centroid is chosen and used in a regression to estimate bus voltage sensitivities to power changes.

Data quality

There were two papers where characteristic load curves obtained from clustering is used to detect bad data. Clustering is done using a modified fuzzy c-means in the first one [110], where the number of clusters, initial clustering centres and initial clusters each customer belongs to are determined based on the data using hill-climbing (assuming all data points are possible cluster centres, calculating a climbing function for each point and choose the clustering centres as the data points with highest value of climbing function). The other one uses k-means to identify and remove abnormal data before performing other statistical analyses, namely identifying patterns of consumption by locality and by strata (social stratification in Bogotá) [111].

7.2 Non-intrusive load disaggregation

Load disaggregation involves identifying which electrical devices are active in a household. This can enable better understanding of customers' electricity demand, ability to enhance the information given to consumers about their consumption, and then improvement of energy efficiency. It can also improve hourly dynamic pricing. Today, the number of appliances used in households having high efficiency increases. These devices draw substantial amounts of power during brief time intervals. An example is the charging of electric vehicles. This can cause strained capacity of the grid in periods of peak demand. Estimating the available flexibility of consumers in peak periods becomes easier if the use of high-power loads is known.

Load disaggregation can either be done intrusively or non-intrusively. Intrusive disaggregation requires measuring each device in the household. The focus of this work lies with non-intrusive (NI) load disaggregation (also called non-intrusive load monitoring, in short NILM), where data analyses are done on the total electricity consumption of the household to divide it into consumption of different devices. Thus, there are no extra metering devices installed in the households.
As an alternative to regular signal processing, [112] deploys graph signal processing (GSP). For pre-processing the data with 10-second and 1-minute time resolution, the base load is detected and removed, and the disaggregation is done in a supervised manner where individual load power data are defined as signals on the graph constructed from the aggregate power data. Finally, a post-processing step refines the results further with respect to inference. The unsupervised method in [113] also deploys GSP for disaggregation of active power time series and consists of three parts: event detection (an event is defined as a change in power above a set threshold), feature extraction/clustering (cluster according to the size of power change in the events, and if it is positive or negative) and feature matching (match the events in a cluster of increasing events to the events in a similar cluster of negative events with respect to the magnitude of the edges and the time between events). The method is tested on data with 1 and 8 second resolution.

Several papers use Hidden Markov Models to model appliances before solving the combination of these HMMs - a Factorial Hidden Markov Model - to find the most probable state of each appliance in a household, and these are described in the three next paragraphs. [114] uses a bootstrap filter for solving and in a given timestep, several trajectories are generated, and each trajectory is weighted by the likelihood of its observation, i.e. the trajectories with higher weights have a higher probability to continue in the next time step. There are two ways to update the weights: parametric (the weight is the likelihood of a sum of series independent Gaussian distributed variables) and resampling (sample the power consumption for each appliance before calculating the observation likelihood). In [115], the solving is done based on an optimisation method trained on measurements of individual appliances, and a particle filtering solver (a type of Monte Carlo algorithm) is used in [116] and [117]. Solving the FHMM is done using a maximum a-posterior particle filter (MAP-PF) in [118]. Different appliances are modelled using HMMs which consider duration modelling and differential observations in [119], while an Explicit-Duration Hidden Markov Model with differential observations (EDHMM-diff) is proposed to disaggregate individual appliances in the total electricity consumption of households.

An adaptive approach which uses Factorial Hidden Markov Model for estimating the power consumptions (power 5 seconds after the device is turned on, average power and power 5 seconds before it is turned off) of appliances and their two-way interactions is described in [120]. First, a database of appliances is created using a portion of the smart meter data, and then disaggregation is performed on the rest of the data. When a change in the aggregate signal occurs, the algorithm performs a search to find out which appliance has been turned on. The estimates of power consumption and two-way interactions of the appliances are updated for every sample of the smart meter data. The same method is used in [121], but without updating appliance models.

Virtual Stochastic Sensors (VSS) which employs Hidden non-Markovian Models (HnMM) (an extension of Hidden Markov Models that explicitly considers the duration of appliance activations), is the disaggregation method of [122]. A method for NILM called Hybrid Signature-based Iterative Disaggregation (HSID) is proposed in [123] which combines Factorial Hidden Markov Models (gives approximation of end-use trajectories) and Iterative Subsequence Dynamic Time Warping (processes the end-use trajectories and match the power consumption of each appliance). One version of the method is supervised and needs measurements of individual appliances, and another is semi-supervised and infers consumption of individual appliances from the aggregated smart meter data.

Some papers compare different methods for disaggregation. One example is [124], where different data features and classification algorithms are evaluated. The data features are either steady state features (active and reactive power variation, current waveform, voltage-current trajectory, or harmonics) or transient state features (nonlinear power transitions when an appliance change operating state, for example over/under shoot amplitude when turned on, rise/fall time or settling time). The classifiers used was k-nearest neighbour, Naïve Bayes, Support Vector Machine, artificial neural network, and decision tree, and they are evaluated based on their recognition error. Another example is [58], which use current data from two sensors installed at low and high-power circuit outlets respectively. It uses 8 odd harmonics as features for disaggregation and PCA for dimensionality reduction. The authors train and test a Bayes classifier, a support vector machine, and an MLP neural network and compared their performances. Also, white noise was added to see how the classifiers handled it.

Two algorithms for NILM on data with lower than 1 second time resolution are introduced in [125]. One is supervised and based on decision trees and one is unsupervised and based on DTW. In both algorithms, pre-processing of data include detecting and amending or removing faulty data, detecting changes in the consumption above the base load, and the events are characterised. Then each algorithm classifies the events to electrical appliances.

The work in [126] use deep neural networks which combines convolutional neural networks (CNN) and variational autoencoders (VAE) which are trained and tested on data with 6 second resolution. In [127], the focus is on disaggregating appliances which are close or overlapping. Use clustering results of measurements of different appliances and propose a method to separate overlapping clusters. In [128] filtering is used to enhance detection of events in active and reactive power data from households. Thereafter they use fuzzy c-means clustering to create signatures for different appliances which also take as input the time of day. Events in the data are matched with appliance signatures in the data base using KNN. Jump models are used in [26] to model different electric appliances, and then the aggregated power consumption by a household is represented by the sum of the jump models and a dynamic programming filtering algorithm is used to reconstruct the power consumption of individual appliances.

In [129], background consumption is detected and events (changes in consumption) are analysed. By comparing the consumption in the event with the average before and after the event, it is decided whether the event is due to an appliance state change or a noise spike. Finally, the appliance state changes are classified using pattern recognition techniques and an appliance database.

A NILM approach for active power data with 1 second resolution or lower is presented in [130]. The first step is to identify appliances which are turned on in a given time period using a spectral signature approach based on the Karhunen Loéve transform. The second step is to estimate the active power consumption signal of the individual appliances using a simplified Mean-Shift Algorithm for clustering of sliding windows, an elimination process to reduce the number of cluster averages which can match the consumption in a defined time window and Bayesian classification to find the most likely combination. A method based on uncorrelated spectral components of an active power consumption signal is introduced in [131] and [132]. They use Karhunen Loéve expansion to split the active power data into subspace components to construct appliance signatures. To reduce the number of possible appliance combinations, power conditions for the subspace components were introduced. An algorithm for identifying which appliances are turned on in a time window was proposed, and finally an energy estimation algorithm was introduced to disaggregate the individual appliances in the combination. In [132], the method was modified to also consider usage behaviour of different residences adaptively.

Three algorithms for analysis sparse coding to use on 10-minute data with different levels of complexity are presented in [32]. In the algorithms, a dictionary of appliances is learned. The algorithms are in order of increasing complexity: simple sparse coding, distinctive analysis sparse coding (the dictionaries for individual appliances should be distinctive), and disaggregating analysis sparse coding (dictionaries for one appliance should not sparsely represent data signals from other appliances). The study in [133] presents a dictionary learning-based approach which is deep in the way that it learns multiple layers of dictionaries for each appliance. The method is used on 10-minute data. In [134], two deep learning multi-label classification algorithms (different appliances represent different classes) are presented, namely multi label consistent deep dictionary learning (MLCDDL) and multi label consistent deep transform learning (MLCDTL).

A semi-supervised method based on non-negative matrix factorisation (NMF, decomposing a matrix which contains only non-negative elements into the product of two matrices containing only non-negative elements) is presented in [135]. The method uses auxiliary feedback from the customer. The auxiliary feedback is in the form of confirmation if the algorithm has disaggregated the correct appliances. The authors of [136] use qualitative data from interviews with households in addition to smart meter data and measurements of a few appliances in the households. Relationships between activities in households and appliances used are mapped and this is used together with disaggregation results to examine when and how long activities are performed. The disaggregation algorithm is based on decision tree.

The work in [137] use data from households which have submetered appliances to train two deep-learning models. One is Factored Four-Way Conditional Restricted Boltzmann Machines (FFW-CRBMs), and the other is Disjunctive FFW-CRBM. The models consist of four layers of neurons:

1. The history layer, which hold the last time series values
2. The present layer, which encodes the current time step
3. The hidden layer, which assures that the model is capable to model the complexity of the time series
4. The label layer, which classify the different time series

The layers are connected with a factorised fourth order tensor, and there is no conditional dependence between the neurons from different layers. The Disjunctive FFW-CRBM contains two factorised fourth order tensors, one for classification and one for regression. The data used has 1 to 3 second resolution.

Low resolution data (15 minutes to 1-hour)

The data from AMS-meters in Norway will have hourly to 15 minutes resolution. However, most successful attempts at NI load disaggregation have been done on data with higher resolution. In this section papers found which use data with 15 minutes to 1-hour resolution, hence are relevant for the Norwegian case, are presented.

The performance of two NILM methods are compared in [138] when data with different time resolutions (6 sec, 30 sec, 1 min, 2 min, 3 min, 4 min, 5 min, 10 min and 15 min) are used as input. The first method is based on FHMM and the other is built on HMM and can handle complex combinations of states for the appliances to be detected. The result is that the performance decreases nonlinearly as the resolution decreases. In [139], the authors use active power data with 15 minute resolution. The smart meter data is decomposed into a discrete set of pulses where each pulse is associated with the operation of appliances. There are three stages: pulse extraction (creating pulses from power data), pulse clustering and classification (use KNN to find a cluster of the most probable appliance for each pulse), and pulse to appliance association (associate the appliance with highest probability to each pulse). In [140], disaggregation is done on 1 hour real and reactive power data using a database of typical appliances in a household and an optimization algorithm. The study in [141] use hourly smart meter data from Norwegian households, weather data, and responses from a household survey in Norway, together with and multiple regression models. Space heating is disaggregated from the total consumption by breaking the total consumption into a temperature-dependent part and a temperature-independent part. The temperature dependent part is taken as the space heating.

The work in [142] have data where some of the households have sub-metering of individual appliances, and a neural network for load disaggregation is trained using the sub-metered data. Then, the trained neural network is used for disaggregation of the households' forecasted load. The aim is to find the percentage of households which should have submetering to be able to decide the load composition of the total forecasted consumption of all households. The data from different households have resolution between 1 and 60 minutes.

Commercial and industrial customers

Not all papers focus on residential customers. In [65], NILM is used on commercial buildings. Smart meter data with 1-minute resolution typical for commercial buildings are created in a laboratory. Feature extraction is done using a technique based on signal processing data transformation techniques, and to classify the different load operations, an extreme learning machine with one hidden layer feed-forward neural network is used. The authors of [143] also focus on commercial and industrial customers. Input is electricity demand from smart meters, demographic data and outdoor temperature. The output is energy-saving advice consisting of a graph and text explaining the graph. Heating, Ventilation and Air Conditioning (HVAC) is disaggregated using a simplified difference method where the consumption without HVAC is taken as the average hourly electricity demand in a month with minimum electricity consumption and the HVAC consumption is taken as the total net hourly electricity demand for consumption without HVAC. The method is used separately on workdays and weekends/holidays. Electricity consumption for lighting is estimated by multiplying the consumption without HVAC with the typical share of lighting in the respective type of building.

Information to customers

NILM is used for a demand side personalised recommendation system in [144]. The NILM algorithm is based on particle filtering (a type of Monte Carlo algorithm), and the appliances are modelled by HMM. NILM is used for a homecare monitoring system which triggers an alarm every time the occupant deviates from their normal routine in [145]. The disaggregation is event-based and use high-resolution measurements. Timestamps when a change occur in the current signal are inferred from the current signal. Then the appliance corresponding to each change is classified by using the active- reactive and distortion power trajectories, which is unique for different appliances.

Smart meters with built-in disaggregation

An improved smart meter which is capable of NILM analysis is presented in [146]. It uses measurements of typical appliances and train a modified Backpropagation neural network to identify appliances. The time series are modified before sent to the neural network. The electricity bill of each appliance is shown in the display of the meter. Another smart meter with NILM is proposed in [147], which also creates an electricity bill showing the contribution from each appliance in the household. They use transient and steady state current signals, where the signal is modified by time series modify (lag-1 method, in which each data point in the time series is replaced by the difference between the current data point and the previous data point in the time series) and an Extreme Learning Machine is designed to perform the disaggregation. A third smart meter with NILM embedded is described in [148]. The meter stores RMS (root mean square) values with 2 Hz resolution from AC signals into a circular buffer. The NILM algorithm first detects changes in the consumption. Then the power density (power content at different frequencies) is calculated and the circular buffer is used to detect periodic components typical for different appliances.

Demand response

Disaggregation is used to improve residential demand response in [149]. An unsupervised method is deployed, using a Multiple Conditional Factorial Hidden Markov Model to detect an appliance and its operation state in real power, reactive power and current harmonics data. In addition, parameters for how many of each appliance a household typically owns and typical correlation between usages of different appliances are used.

Clustering before disaggregating

A few papers deploy clustering before disaggregating. The authors of [150] use k-means clustering and Support Vector Machine on data with 1 minute resolution or higher, where k-means is used to reduce the dataset before training the SVM. A labelled training data set inquired from appliance measurements or time diaries is used. The study in [151] use changes in active and reactive power as features, and events are clustered using dynamic fuzzy c-means. To reveal operating states, the clusters are denoised and processed. Finally, a Finite State Machine (a mathematical model which can be in one state at a time and each state model a state of an appliance) is used to estimate the energy consumption of different appliances. In [37], the data is clustered using k-median, and dynamic time warping is used to identify the operating appliance. Then, the energy consumption from an appliance is retrieved using an optimised support vector regression.

Disaggregating a specific load

Some papers focus on disaggregating one specific load from the total consumption, instead of disaggregating the total consumption into all its individual loads. Three papers were found which considers air conditioning (AC). [59] use two kinds of neural networks, namely Multi-layer Perceptron (MLP) and Long Short-Term Memory (LSTM) layer, and then compare their performances. [152] disaggregate the AC load in power data with 1 second resolution. First, events are detected (appliances turned on or off). Then, features which describe the difference between AC and other appliances are extracted, and the extracted features are used to build an SVM model. In [153], HVAC is disaggregated from total power data with 1 minute resolution to discover air leakages in buildings. One recurrent neural network with LSTM layers and one denoising autoencoder is used to detect the cycles of the HVAC. The neural networks are trained on a dataset which contains measurements of the AC unit in the house and measurements of Air Changes per Hour (which is used as a parameter for whether the house has a leakage).

A few papers also consider disaggregation of electric vehicles. In [27], the result when performed on 1 minute smart meter power data is used to define flexibility for the aggregated charging demand. The paper uses an unsupervised method which deploys the signal processing method ICA to extract the charging profiles. The method uses predefined levels of amplitudes of the chargings (in kW). [154] use the same method, but in addition separate chargings into three stages: gradual increase, steady charging and gradual decrease and extract each stage of charging separately from the power data. In [155], total electricity consumed by electric vehicles (EVs) is calculated using data down to 1 second resolution. First, baseline noise is removed from the signal, and since EVs are usually charging with more than 3 kW, a threshold of 2.5 kW is used to filter the signal. Then they find the number of segments in the data, and the start and end point of these. A segment filter removes segments from other appliances than EV depending on the lengths of the segments. The width and height of the remaining segments are determined and compared with a database of EV profiles.

One paper also considers disaggregation of solar generation from net consumption data in households using an unsupervised method. The first step is to distinguish the customers with and without PV using k-means clustering. It is assumed that the consumption behaviour is a mixture of latent representative behaviours, and clustering of customers without PV is performed (using k-medoid clustering with Dynamic Time Warping as distance measure) to find these. The second step is to disaggregate the consumption data of the customers with PV using the centroids of the resulting clusters from the previous step, in addition to solar irradiation.

Forecasting

In a few cases, NILM has been used to enhance forecasting of electricity consumption. In [156], appliance usage patterns (AUPs) from a household are learned using a NILM algorithm based on spectral decomposition on measurements of power of individual appliances. The learnt AUPs are then used to bias the probabilities of different appliances in a fuzzy system. Finally, the total power consumption of some households was forecasted a few minutes ahead. In [157], NILM and clustering of appliances with similar behaviour are used as a pre-processing step before performing short-term load forecasting.

Substation level

Data from a distribution feeder is disaggregated into data for individual customers in [158], using a database with load profiles from smart meter data from the same geographical area as the feeder. Also in [159], data from substations are used, where a feedforward neural network is used on RMS measurements of active and reactive power and voltage to perform real-time estimation of the share of different loads in the data. Monte Carlo simulations are used to create data for training and validation.

8 Summary

Almost every household in Norway had a smart electricity meter installed by the end of 2018. The advanced functionality of the meters and the large amounts of data transmitted from them have the potential to contribute to more than just a more accurate billing of customers. This report is a result from the ENERGYTICS project, which aims at exploring this potential.

This state-of-the-art included a description of experiences and technical advances regarding deployment of smart meters (AMS) in the Scandinavian countries and Finland. Sweden has deployed smart meters with hourly electricity readings since 2009 and was the first EU country to reach 100% penetration of smart meters. The financial incentive was that customers who subscribed to hourly-based electricity billing got this for free. Finland had a goal of reaching 80% penetration of hourly metering by the end of 2013, and it was the responsibility of the DSOs to reach the goal. In Denmark there was a goal to reach 100% smart meter coverage among households in 2020, and in 2016, more than 50% had smart meters installed.

Possible benefits and advantages of deploying data science on data from the electricity grid were explored. A key takeaway is that a more efficient use of the capacity of the distribution network can be implemented, for example by deploying demand response. In addition, an improved and faster handling and locating of faults and failures can be achieved, power and voltage quality can be monitored, and maintenance schemes and reinvestment decisions can be enhanced and optimised.

Different data sources in the distribution electricity grid and relevant data science techniques were presented. Data sources include AMS data, data about failures which has happened in the grid, grid topology, power flow data, and power quality data. Data science techniques include pre-processing of data before the main analysis is performed. This often means representing the data in another or simpler way which easier can be used in the analysis. The analysis is often done with machine learning, which can be divided into supervised learning and unsupervised learning. In supervised learning, algorithms learn the relation between features and labels from known data such that it can propose the label for unknown data in the future. Unsupervised learning does not rely on labels for the data.

The focus of the state-of-the-art was on two applications of data science: clustering of customers and disaggregation of total electricity consumption into the consumption of individual loads.
When it comes to clustering of customers, there are different attempts at pre-processing data before clustering. For the clustering algorithms themselves, there are both attempts at improving existing clustering algorithms and proposing new ones. It is common to use clustering as a first step to be able to improve forecasting of electricity consumption of households. The aim of the clustering can also be to detect electricity theft, cyber-attacks, or bad quality of the metered data. A few papers explore the possibility of using the clustering results to enhance demand response of households, and others use them to optimise pricing strategies. Finally, the clustering results can also be used to confirm the topology of the grid, and that customers are connected in the way which the DSO has registered.

For disaggregation of total electricity, the focus is on disaggregation where there is no extra metering installed in the households, and only data available to the DSO is used. Some papers use the pre-processing techniques of Hidden Markov Models to represent the consumption data as states when appliances are turned on and off, and in many papers noise is removed before the disaggregation is performed. Several papers compare different methods for disaggregation, and others introduce new methods. Of particular relevance to disaggregation of smart meter data in Norway are the analyses done on data with 15 minutes resolution or lower, but the number of such analyses is small. Common goals of the disaggregation is to be able to aggregate a specific load (e.g. air conditioning, ventilation, heating, and electric vehicle), give customers more information about their consumption and enhanced demand response. Like clustering, disaggregation can be used to improve forecasting of electricity demand in households.

9 References

[1] 'Bytte av strømmålere har vært en suksess'. https://www.nymaler.no/aktuelt/nyheter-2019/bytte-av-strommalere-har-vart-en-suksess/ (accessed Sep. 06, 2019).
[2] Olje- og energidepartementet, 'Forskrift om måling, avregning, fakturering av nettjenester og elektrisk energi, nettselskapets nøytralitet mv - Lovdata'. https://lovdata.no/dokument/SF/forskrift/1999-03-11-301 (accessed Mar. 22, 2019).
[3] A. Venjum, 'Smarte målere (AMS) - Status og planer for installasjon per 1. halvår 2016', NVE, 79–2016, Oct. 2016. Accessed: Sep. 06, 2019. [Online]. Available: http://publikasjoner.nve.no/rapport/2016/rapport2016_79.pdf.
[4] S. Zhou and M. A. Brown, 'Smart meter deployment in Europe: A comparative case study on the impacts of national policy schemes', *J. Clean. Prod.*, vol. 144, pp. 22–32, Feb. 2017, doi: 10.1016/j.jclepro.2016.12.031.
[5] C. Dromaque, S. Xu, and S. Baynes, 'Case study on innovative smart billing for household consumers', 2013. Accessed: Mar. 22, 2019. [Online]. Available: https://wec-policies.enerdata.net/Documents/cases-studies/Smart_Billing.pdf.
[6] A. Christensson, N. Gerson, and E. Wallin, 'A Mapping of Scandinavian Smart Grid Development in the Distribution System from an ICT perspective', Kungliga Tekniska högskolan, 2013.
[7] European Commission, 'Benchmarking smart metering deployment in the EU-27 with a focus on electricity', Jun. 2014. Accessed: Mar. 22, 2019. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52014DC0356&from=EN.
[8] Fortum, 'Fortum invests in automatic meter management in Sweden', May 31, 2006. https://www.fortum.com/media/2006/05/fortum-invests-automatic-meter-management-sweden (accessed Mar. 22, 2019).
[9] Finnish Energy Industries, 'Principles of hourly metering 2010', 2010. Accessed: Mar. 22, 2019. [Online]. Available: https://energia.fi/files/570/Principles_of_hourly_metering_-_recommendation.pdf.
[10] Fortum, 'Fortum completes smart meters installation project in Finland', Oct. 04, 2013. https://www.fortum.com/media/2013/10/fortum-completes-smart-meters-installation-project-finland (accessed Mar. 22, 2019).
[11] Landis+Gyr, 'Landis+Gyr Win €10 million Euro Smart Meter Project with Finnish Utility'. https://www.landisgyr.eu/news/landisgyr-win-e10-million-euro-smart-meter-project-with-finnish-utility/ (accessed Mar. 22, 2019).
[12] M. Kauppinen, J. Pylvänäinen, J. Karjalainen, and V. Sihvola, 'Experiences of using AMI system for DSO's business operation', *CIRED - Open Access Proc. J.*, vol. 2017, no. 1, pp. 2756–2759, 2017, doi: 10.1049/oap-cired.2017.0764.
[13] USmartConsumer Project, 'European Smart Metering Landscape Report', Nov. 2016. Accessed: Mar. 22, 2019. [Online]. Available: http://www.escansa.es/usmartconsumer/documentos/USmartConsumer_European_Landscape_Report_2016_web.pdf.
[14] H. C. Gils, 'Assessment of the theoretical demand response potential in Europe', *Energy*, vol. 67, pp. 1–18, Apr. 2014, doi: 10.1016/j.energy.2014.02.019.
[15] S. Feuerriegel, P. Bodenbenner, and D. Neumann, 'Value and Granularity of ICT and Smart Meter Data in Demand Response Systems', Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3215398, Oct. 2015. Accessed: Mar. 25, 2019. [Online]. Available: https://papers.ssrn.com/abstract=3215398.
[16] 'Årsrapporter fra Feilanalyse', *Statnett*. https://www.statnett.no/for-aktorer-i-kraftbransjen/systemansvaret/praktisering-av-systemansvaret/arsrapporter-fra-feilanalyse/ (accessed Oct. 29, 2020).
[17] M. H. J. Bollen, 'What is power quality?', *Electr. Power Syst. Res.*, vol. 66, no. 1, pp. 5–14, Jul. 2003, doi: 10.1016/S0378-7796(03)00067-1.

[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. .
[19] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*. New York: Springer-Verlag, 2013.
[20] A. Tureczek, P. S. Nielsen, and H. Madsen, 'Electricity Consumption Clustering Using Smart Meter Data', *Energies*, vol. 11, no. 4, p. 859, Apr. 2018, doi: 10.3390/en11040859.
[21] T. K. Wijaya, M. Vasirani, S. Humeau, and K. Aberer, 'Cluster-based aggregate forecasting for residential electricity demand using smart meter data', in *2015 IEEE International Conference on Big Data (Big Data)*, Oct. 2015, pp. 879–887, doi: 10.1109/BigData.2015.7363836.
[22] C. S. Burrus, M. Frigo, and G. S. Johnson, *Fast Fourier Transforms*. Samurai Media Limited, 2018.
[23] P. Welch, 'The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms', *IEEE Trans. Audio Electroacoustics*, vol. 15, no. 2, pp. 70–73, Jun. 1967, doi: 10.1109/TAU.1967.1161901.
[24] Y. Wang, Q. Chen, C. Kang, and Q. Xia, 'Clustering of Electricity Consumption Behavior Dynamics Toward Big Data Applications', *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2437–2447, Sep. 2016, doi: 10.1109/TSG.2016.2548565.
[25] L. R. Rabiner, 'A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition', Feb. 02, 1989. https://www.cs.ubc.ca/~murphyk/Bayes/rabiner.pdf (accessed Sep. 30, 2019).
[26] V. Breschi, D. Piga, and A. Bemporad, 'Jump model learning and filtering for energy end-use disaggregation', *IFAC-Pap.*, vol. 51, no. 15, pp. 275–280, Jan. 2018, doi: 10.1016/j.ifacol.2018.09.147.
[27] A. A. Munshi and Y. A. I. Mohamed, 'Extracting and Defining Flexibility of Residential Electrical Vehicle Charging Loads', *IEEE Trans. Ind. Inform.*, vol. 14, no. 2, pp. 448–461, Feb. 2018, doi: 10.1109/TII.2017.2724559.
[28] C. Maccone, Ed., 'A simple introduction to the KLT (Karhunen—Loève Transform)', in *Deep Space Flight and Communications: Exploiting the Sun as a Gravitational Lens*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 151–179.
[29] S. Ikeda and H. Nishi, 'Sparse-coding-based household clustering for demand response services', in *2016 IEEE 25th International Symposium on Industrial Electronics (ISIE)*, Jun. 2016, pp. 744–749, doi: 10.1109/ISIE.2016.7744982.
[30] Y. Wang, Q. Chen, C. Kang, Q. Xia, and M. Luo, 'Sparse and Redundant Representation-Based Smart Meter Data Compression and Pattern Extraction', *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2142–2151, May 2017, doi: 10.1109/TPWRS.2016.2604389.
[31] A. Aleshinloye, A. Bais, and I. Al-Anbagi, 'Performance analysis of dimensionality reduction techniques for demand side management', in *2017 IEEE Electrical Power and Energy Conference (EPEC)*, Oct. 2017, pp. 1–6, doi: 10.1109/EPEC.2017.8286232.
[32] S. Singh and A. Majumdar, 'Analysis Co-Sparse Coding for Energy Disaggregation', *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 462–470, Jan. 2019, doi: 10.1109/TSG.2017.2743763.
[33] Mathworks, 'What Is Machine Learning?' https://se.mathworks.com/discovery/machine-learning.html (accessed Mar. 25, 2019).
[34] F. Chollet, *Deep learning with Python*. Shelter Island, New York: Manning Publications Co, 2018.
[35] Natinal Institute of Standards and Technology and SEMATECH, 'Weighted Least Squares Regression'. https://www.itl.nist.gov/div898/handbook/pmd/section1/pmd143.htm (accessed Mar. 25, 2019).
[36] F. A. A. Alseiari and Z. Aung, 'Real-time anomaly-based distributed intrusion detection systems for advanced Metering Infrastructure utilizing stream data mining', in *2015 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE)*, Oct. 2015, pp. 148–153, doi: 10.1109/ICSGCE.2015.7454287.

[37] Q. Yuan, H. Wang, B. Wu, Y. Song, and H. Wang, ‘A Fusion Load Disaggregation Method Based on Clustering Algorithm and Support Vector Regression Optimization for Low Sampling Data’, *Future Internet*, vol. 11, no. 2, p. 51, Feb. 2019, doi: 10.3390/fi11020051.

[38] J. Oyedokun, S. Bu, Y. Xiao, and Z. Han, ‘Smart Meter Data Characterization and Clustering for Peak Demand Targeting in Smart Grids’, in *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Oct. 2018, pp. 1–6, doi: 10.1109/ISGTEurope.2018.8571875.

[39] F. Fahiman, S. M. Erfani, S. Rajasegarar, M. Palaniswami, and C. Leckie, ‘Improving load forecasting based on deep learning and K-shape clustering’, in *2017 International Joint Conference on Neural Networks (IJCNN)*, May 2017, pp. 4134–4141, doi: 10.1109/IJCNN.2017.7966378.

[40] MATLAB & Simulink, ‘Fuzzy C-Means Clustering’. https://www.mathworks.com/help/fuzzy/fuzzy-c-means-clustering.html (accessed Jul. 12, 2019).

[41] R. C. Eberhart and Y. Shi, ‘Comparing inertia weights and constriction factors in particle swarm optimization’, in *Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512)*, Jul. 2000, vol. 1, pp. 84–88 vol.1, doi: 10.1109/CEC.2000.870279.

[42] S. Raha, T. Som, and N. Chakraborty, ‘Constriction factor based particle swarm optimization applied to reactive power dispatch in transmission system’, in *International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2011)*, Jul. 2011, pp. 335–339, doi: 10.1049/cp.2011.0385.

[43] W. Qiu, F. Zhai, Z. Bao, B. Li, Q. Yang, and Y. Cao, ‘Clustering approach and characteristic indices for load profiles of customers using data from AMI’, in *2016 China International Conference on Electricity Distribution (CICED)*, Aug. 2016, pp. 1–5, doi: 10.1109/CICED.2016.7576194.

[44] S. Haben, C. Singleton, and P. Grindrod, ‘Analysis and Clustering of Residential Customers Energy Behavioral Demand Using Smart Meter Data’, *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 136–144, Jan. 2016, doi: 10.1109/TSG.2015.2409786.

[45] ‘Gaussian mixture models — scikit-learn 0.21.3 documentation’. https://scikit-learn.org/stable/modules/mixture.html (accessed Sep. 05, 2019).

[46] C. Maklin, ‘Gaussian Mixture Models Clustering Algorithm Explained’, *Medium*, Jul. 16, 2019. https://towardsdatascience.com/gaussian-mixture-models-d13a5e915c8e (accessed Sep. 05, 2019).

[47] Yizong Cheng, ‘Mean shift, mode seeking, and clustering’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995, doi: 10.1109/34.400568.

[48] C. Cobos, M. Mendoza, M. Manic, E. León, and E. Herrera-Viedma, ‘Clustering of web search results based on an Iterative Fuzzy C-means Algorithm and Bayesian Information Criterion’, in *2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS)*, Jun. 2013, pp. 507–512, doi: 10.1109/IFSA-NAFIPS.2013.6608452.

[49] D. L. Davies and D. W. Bouldin, ‘A Cluster Separation Measure’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979, doi: 10.1109/TPAMI.1979.4766909.

[50] M. Charrad, N. Ghazzali, V. Boiteau, and A. Niknafs, ‘NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set’, *J. Stat. Softw.*, vol. 61, no. 1, pp. 1–36, Nov. 2014, doi: 10.18637/jss.v061.i06.

[51] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, ‘On Clustering Validation Techniques’, *J. Intell. Inf. Syst.*, 2001.

[52] A. G. Kazaki and T. A. Papadopoulos, ‘Cluster analysis of university campus smart meter data’, in *2018 53rd International Universities Power Engineering Conference (UPEC)*, Sep. 2018, pp. 1–6, doi: 10.1109/UPEC.2018.8541941.

[53] X. L. Xie and G. Beni, ‘A validity measure for fuzzy clustering’, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 8, pp. 841–847, Aug. 1991, doi: 10.1109/34.85677.

[54] scikit-learn developers, ‘cosine_similarity - scikit-learn 0.21.3 documentation’. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html (accessed Sep. 05, 2019).

[55] M. Grabner, Z. Bregar, Š. Ivanjko, and L. Valenčič, ‘Improved model for the spatial load forecasting of the Slovenian distribution network’, *CIRED - Open Access Proc. J.*, vol. 2017, no. 1, pp. 2354–2357, 2017, doi: 10.1049/oap-cired.2017.1020.
[56] I. Benítez, J.-L. Díez, A. Quijano, and I. Delgado, ‘Dynamic clustering of residential electricity consumption time series data based on Hausdorff distance’, *Electr. Power Syst. Res.*, vol. 140, pp. 517–526, Nov. 2016, doi: 10.1016/j.epsr.2016.05.023.
[57] M. A. Nielsen, *Neural Networks and Deep Learning*. 2015.
[58] S. Semwal, M. Singh, and R. S. Prasad, ‘Group control and identification of residential appliances using a nonintrusive method’, *Turk. J. Electr. Eng. Comput. Sci.*, vol. 23, pp. 1805–1816, 2015, doi: 10.3906/elk-1404-59.
[59] J. Cho, Z. Hu, and M. Sartipi, ‘Non-Intrusive A/C Load Disaggregation Using Deep Learning’, in *2018 IEEE/PES Transmission and Distribution Conference and Exposition (T D)*, Apr. 2018, pp. 1–5, doi: 10.1109/TDC.2018.8440358.
[60] D. C. Mocanu, E. Mocanu, P. H. Nguyen, M. Gibescu, and A. Liotta, ‘A topological insight into restricted Boltzmann machines’, *Mach. Learn.*, vol. 104, no. 2, pp. 243–270, Sep. 2016, doi: 10.1007/s10994-016-5570-z.
[61] D. Monn, ‘What are Variational Autoencoders? A simple explanation’, *Medium*, Dec. 11, 2017. https://medium.com/@dmonn/what-are-variational-autoencoders-a-simple-explanation-ea7dccafb0e3 (accessed Sep. 09, 2019).
[62] D. Monn, ‘Denoising Autoencoders explained’, *Medium*, Jul. 18, 2017. https://towardsdatascience.com/denoising-autoencoders-explained-dbb82467fc2 (accessed Sep. 09, 2019).
[63] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, ‘Extreme learning machine: Theory and applications’, *Neurocomputing*, vol. 70, no. 1, pp. 489–501, Dec. 2006, doi: 10.1016/j.neucom.2005.12.126.
[64] T. Chen, K. Qian, A. Mutanen, B. Schuller, P. Järventausta, and W. Su, ‘Classification of electricity customer groups towards individualized price scheme design’, in *2017 North American Power Symposium (NAPS)*, Sep. 2017, pp. 1–4, doi: 10.1109/NAPS.2017.8107189.
[65] K. Khalid, A. Mohamed, R. Mohamed, and H. Shareef, ‘Nonintrusive load identification using extreme learning machine and TT-transform’, in *2016 International Conference on Advances in Electrical, Electronic and Systems Engineering (ICAEES)*, Nov. 2016, pp. 271–276, doi: 10.1109/ICAEES.2016.7888051.
[66] Z. A. Khan and D. Jayaweera, ‘Approach for smart meter load profiling in Monte Carlo simulation applications’, *Transm. Distrib. IET Gener.*, vol. 11, no. 7, pp. 1856–1864, 2017, doi: 10.1049/iet-gtd.2016.2084.
[67] Z. A. Khan, D. Jayaweera, and H. Gunduz, ‘Smart meter data taxonomy for demand side management in smart grids’, in *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, Oct. 2016, pp. 1–8, doi: 10.1109/PMAPS.2016.7764143.
[68] L. Wen, K. Zhou, and S. Yang, ‘A shape-based clustering method for pattern recognition of residential electricity consumption’, *J. Clean. Prod.*, vol. 212, pp. 475–488, Mar. 2019, doi: 10.1016/j.jclepro.2018.12.067.
[69] E. Pan, H. Li, L. Song, and Z. Han, ‘Kernel-based non-parametric clustering for load profiling of big smart meter data’, in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, Mar. 2015, pp. 2251–2255, doi: 10.1109/WCNC.2015.7127817.
[70] I. Khan, J. Z. Huang, M. A. Masud, and Q. Jiang, ‘Segmentation of Factories on Electricity Consumption Behaviors Using Load Profile Data’, *IEEE Access*, vol. 4, pp. 8394–8406, 2016, doi: 10.1109/ACCESS.2016.2619898.
[71] R. Al-Otaibi, N. Jin, T. Wilcox, and P. Flach, ‘Feature Construction and Calibration for Clustering Daily Load Curves from Smart-Meter Data’, *IEEE Trans. Ind. Inform.*, vol. 12, no. 2, pp. 645–654, Apr. 2016, doi: 10.1109/TII.2016.2528819.

[72] M. Azaza and F. Wallin, 'Smart meter data clustering using consumption indicators: responsibility factor and consumption variability', *Energy Procedia*, vol. 142, pp. 2236–2242, Dec. 2017, doi: 10.1016/j.egypro.2017.12.624.
[73] S. Iyengar, S. Lee, D. Irwin, and P. Shenoy, 'Analyzing Energy Usage on a City-scale Using Utility Smart Meters', in *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*, New York, NY, USA, 2016, pp. 51–60, doi: 10.1145/2993422.2993425.
[74] J. E. Parra, F. L. Quilumba, and H. N. Arcos, 'Customers' demand clustering analysis — A case study using smart meter data', in *2016 IEEE PES Transmission Distribution Conference and Exposition-Latin America (PES T D-LA)*, Sep. 2016, pp. 1–7, doi: 10.1109/TDC-LA.2016.7805685.
[75] X. Luo, T. Hong, Y. Chen, and M. A. Piette, 'Electric load shape benchmarking for small- and medium-sized commercial buildings', *Appl. Energy*, vol. 204, pp. 715–725, Oct. 2017, doi: 10.1016/j.apenergy.2017.07.108.
[76] F. N. Melzi, A. Same, M. H. Zayani, and L. Oukhellou, 'A Dedicated Mixture Model for Clustering Smart Meter Data: Identification and Analysis of Electricity Consumption Behaviors', *Energies*, vol. 10, no. 10, p. 1446, Oct. 2017, doi: 10.3390/en10101446.
[77] A.-L. Klingler and F. Schuhmacher, 'Residential photovoltaic self-consumption: Identifying representative household groups based on a cluster analysis of hourly smart-meter data', *Energy Effic.*, vol. 11, no. 7, pp. 1689–1701, Oct. 2018, doi: 10.1007/s12053-017-9554-z.
[78] A. Camero, G. Luque, Y. Bravo, and E. Alba, 'Customer Segmentation Based on the Electricity Demand Signature: The Andalusian Case', *Energies*, vol. 11, no. 7, p. 1788, Jul. 2018, doi: 10.3390/en11071788.
[79] H. Fang, Y. Zhang, M. Liu, and W. Shen, 'Clustering and Analysis of Household Power Load Based on HMM and Multi-factors', in *2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD))*, May 2018, pp. 491–495, doi: 10.1109/CSCWD.2018.8465353.
[80] M. M. Selvam, R. Gnanadass, and N. P. Padhy, 'Fuzzy based clustering of smart meter data using real power and THD patterns', *Energy Procedia*, vol. 117, pp. 401–408, Jun. 2017, doi: 10.1016/j.egypro.2017.05.158.
[81] I. Khan, J. Z. Huang, and K. Ivanov, 'Incremental density-based ensemble clustering over evolving data streams', *Neurocomputing*, vol. 191, pp. 34–43, May 2016, doi: 10.1016/j.neucom.2016.01.009.
[82] Y. Lu, T. Zhang, and Z. Zeng, 'Adaptive weighted fuzzy clustering algorithm for load profiling of smart grid customers', in *2016 IEEE/CIC International Conference on Communications in China (ICCC)*, Jul. 2016, pp. 1–6, doi: 10.1109/ICCChina.2016.7636874.
[83] K. Mets, F. Depuydt, and C. Develder, 'Two-Stage Load Pattern Clustering Using Fast Wavelet Transformation', *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2250–2259, Sep. 2016, doi: 10.1109/TSG.2015.2446935.
[84] Z. Jiang, R. Lin, and F. Yang, 'A Hybrid Machine Learning Model for Electricity Consumer Categorization Using Smart Meter Data', *Energies*, vol. 11, no. 9, p. 2235, Sep. 2018, doi: 10.3390/en11092235.
[85] I. Pintér, L. Kovács, R. Drenyovszi, A. Oláh, and K. Tornai, 'Jensen-Shannon divergence based algorithm for adaptive segmentation and labelling of household's electricity power consumption data series', in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2016, pp. 001912–001916, doi: 10.1109/SMC.2016.7844518.
[86] I. Pintér, L. Kovács, A. Oláh, R. Drenyovszki, D. Tisza, and K. Tornai, 'Automatic segmentation of electricity consumption data series with Jensen-Shannon divergence', in *2014 IEEE International Energy Conference (ENERGYCON)*, May 2014, pp. 828–832, doi: 10.1109/ENERGYCON.2014.6850521.
[87] J. Zhong, J. Wang, X. Yu, Q. Wang, M. Combariza, and G. Holmes, 'Hybrid Load Profile Clustering for identifying patterns of electricity consumers', in *2016 IEEE 25th International Symposium on Industrial Electronics (ISIE)*, Jun. 2016, pp. 708–713, doi: 10.1109/ISIE.2016.7744976.

[88] S. Bandyopadhyay, T. Ganu, H. Khadilkar, and V. Arya, 'Individual and Aggregate Electrical Load Forecasting: One for All and All for One', in *Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems*, New York, NY, USA, 2015, pp. 121–130, doi: 10.1145/2768510.2768539.

[89] Y. Li, P. Guo, and X. Li, 'Short-Term Load Forecasting Based on the Analysis of User Electricity Behavior', *Algorithms*, vol. 9, no. 4, p. 80, Dec. 2016, doi: 10.3390/a9040080.

[90] A. Kell, A. S. McGough, and M. Forshaw, 'Segmenting Residential Smart Meter Data for Short-Term Load Forecasting', in *Proceedings of the Ninth International Conference on Future Energy Systems*, New York, NY, USA, 2018, pp. 91–96, doi: 10.1145/3208903.3208923.

[91] F. L. Quilumba, W. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, 'Using Smart Meter Data to Improve the Accuracy of Intraday Load Forecasting Considering Customer Behavior Similarities', *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 911–918, Mar. 2015, doi: 10.1109/TSG.2014.2364233.

[92] Y. Lu, T. Zhang, Z. Zeng, and J. Loo, 'An improved RBF neural network for short-term load forecast in smart grids', in *2016 IEEE International Conference on Communication Systems (ICCS)*, Dec. 2016, pp. 1–6, doi: 10.1109/ICCS.2016.7833643.

[93] P. Laurinec and M. Lucká, 'New clustering-based forecasting method for disaggregated end-consumer electricity load using smart grid data', in *2017 IEEE 14th International Scientific Conference on Informatics*, Nov. 2017, pp. 210–215, doi: 10.1109/INFORMATICS.2017.8327248.

[94] U. Singh, V. Zamani, and M. Baran, 'On-line load estimation for distribution automation using AMI data', in *2016 IEEE Power and Energy Society General Meeting (PESGM)*, Jul. 2016, pp. 1–5, doi: 10.1109/PESGM.2016.7741806.

[95] A. Shahzadeh, A. Khosravi, and S. Nahavandi, 'Improving load forecast accuracy by clustering consumers using smart meter data', in *2015 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2015, pp. 1–7, doi: 10.1109/IJCNN.2015.7280393.

[96] Z. A. Khan and D. Jayaweera, 'Approach for forecasting smart customer demand with significant energy demand variability', in *2018 1st International Conference on Power, Energy and Smart Grid (ICPESG)*, Apr. 2018, pp. 1–5, doi: 10.1109/ICPESG.2018.8384528.

[97] B. Yildiz, J. I. Bilbao, J. Dore, and A. Sproul, 'Household electricity load forecasting using historical smart meter data with clustering and classification techniques', in *2018 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, May 2018, pp. 873–879, doi: 10.1109/ISGT-Asia.2018.8467837.

[98] P. Jokar, N. Arianpoo, and V. C. M. Leung, 'Electricity Theft Detection in AMI Using Customers' Consumption Patterns', *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 216–226, Jan. 2016, doi: 10.1109/TSG.2015.2425222.

[99] K. Zheng, Y. Wang, Q. Chen, and Y. Li, 'Electricity theft detecting based on density-clustering method', in *2017 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia)*, Dec. 2017, pp. 1–6, doi: 10.1109/ISGT-Asia.2017.8378347.

[100] K. Zheng, Q. Chen, Y. Wang, C. Kang, and Q. Xia, 'A Novel Combined Data-Driven Approach for Electricity Theft Detection', *IEEE Trans. Ind. Inform.*, vol. 15, no. 3, pp. 1809–1819, Mar. 2019, doi: 10.1109/TII.2018.2873814.

[101] X. Yang, X. Zhang, J. Lin, W. Yu, and P. Zhao, 'A Gaussian-Mixture Model Based Detection Scheme against Data Integrity Attacks in the Smart Grid', in *2016 25th International Conference on Computer Communication and Networks (ICCCN)*, Aug. 2016, pp. 1–9, doi: 10.1109/ICCCN.2016.7568478.

[102] A. Anwar, A. N. Mahmood, and Z. Tari, 'Identification of vulnerable node clusters against false data injection attack in an AMI based Smart Grid', *Inf. Syst.*, vol. 53, pp. 201–212, Oct. 2015, doi: 10.1016/j.is.2014.12.001.

[103] M. G. Kallitsis, S. Bhattacharya, S. Stoev, and G. Michailidis, 'Adaptive statistical detection of false data injection attacks in smart grids', in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Dec. 2016, pp. 826–830, doi: 10.1109/GlobalSIP.2016.7905958.

[104] H. Liu, N. Mahmoudi, and K. Chen, 'Microgrids Real-Time Pricing Based on Clustering Techniques', *Energies*, vol. 11, no. 6, p. 1388, Jun. 2018, doi: 10.3390/en11061388.
[105] V. Jain, N. Jain, and R. R. Joshi, 'Load index based clustering algorithm using centroid model in different pricing environment', in *2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, Dec. 2016, pp. 1–7, doi: 10.1109/ICRAIE.2016.7939472.
[106] R. Mitra *et al.*, 'Voltage Correlations in Smart Meter Data', in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2015, pp. 1999–2008, doi: 10.1145/2783258.2788594.
[107] F. Ni, J. Q. Liu, F. Wei, C. D. Zhu, and S. X. Xie, 'Phase identification in distribution systems by data mining methods', in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, Nov. 2017, pp. 1–6, doi: 10.1109/EI2.2017.8245748.
[108] F. Olivier, A. Sutera, P. Geurts, R. Fonteneau, and D. Ernst, 'Phase Identification of Smart Meters by Clustering Voltage Measurements', in *2018 Power Systems Computation Conference (PSCC)*, Jun. 2018, pp. 1–8, doi: 10.23919/PSCC.2018.8442853.
[109] G. Valverde, T. Zufferey, S. Karagiannopoulos, and G. Hug, 'Estimation of voltage sensitivities to power injections using smart meter data', in *2018 IEEE International Energy Conference (ENERGYCON)*, Jun. 2018, pp. 1–6, doi: 10.1109/ENERGYCON.2018.8398841.
[110] Z. Wu *et al.*, 'Power system bad load data detection based on an improved fuzzy C-means clustering algorithm', in *2017 IEEE Power Energy Society General Meeting*, Jul. 2017, pp. 1–5, doi: 10.1109/PESGM.2017.8274732.
[111] S. Celis, L. F. Giraldo, P. D. O.-D. Jesus, and C. A. Rincon, 'A Clustering Approach for Domestic Smart Metering Data Preprocessing', in *2018 IEEE ANDESCON*, Aug. 2018, pp. 1–3, doi: 10.1109/ANDESCON.2018.8564597.
[112] K. Kumar, R. Sinha, M. G. Chandra, and N. K. Thokala, 'Data-driven electrical load disaggregation using graph signal processing', in *2016 IEEE Annual India Conference (INDICON)*, Dec. 2016, pp. 1–6, doi: 10.1109/INDICON.2016.7839109.
[113] B. Zhao, L. Stankovic, and V. Stankovic, 'On a Training-Less Solution for Non-Intrusive Appliance Load Monitoring Using Graph Signal Processing', *IEEE Access*, vol. 4, pp. 1784–1799, 2016, doi: 10.1109/ACCESS.2016.2557460.
[114] Weicong Kong, Zhaoyang Dong, Yan Xu, and D. Hill, 'An enhanced bootstrap filtering method for non-intrusive load monitoring', in *2016 IEEE Power and Energy Society General Meeting (PESGM)*, Jul. 2016, pp. 1–5, doi: 10.1109/PESGM.2016.7741487.
[115] W. Kong, Z. Y. Dong, J. Ma, D. J. Hill, J. Zhao, and F. Luo, 'An Extensible Approach for Non-Intrusive Load Disaggregation With Smart Meter Data', *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3362–3372, Jul. 2018, doi: 10.1109/TSG.2016.2631238.
[116] W. Kong, Z. Y. Dong, D. J. Hill, J. Ma, J. H. Zhao, and F. J. Luo, 'A Hierarchical Hidden Markov Model Framework for Home Appliance Modeling', *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3079–3090, Jul. 2018, doi: 10.1109/TSG.2016.2626389.
[117] D. Egarter, V. P. Bhuvana, and W. Elmenreich, 'PALDi: Online Load Disaggregation via Particle Filtering', *IEEE Trans. Instrum. Meas.*, vol. 64, no. 2, pp. 467–477, Feb. 2015, doi: 10.1109/TIM.2014.2344373.
[118] V. P. Bhuvana and A. M. Tonello, 'Appliance load disaggregation based on maximum a-posterior particle filter (MAP-PF)', in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct. 2017, pp. 165–170, doi: 10.1109/SmartGridComm.2017.8340699.
[119] Z. Guo, Z. J. Wang, and A. Kashani, 'Home Appliance Load Modeling From Aggregated Smart Meter Data', *IEEE Trans. Power Syst.*, vol. 30, no. 1, pp. 254–262, Jan. 2015, doi: 10.1109/TPWRS.2014.2327041.
[120] M. Aiad and P. H. Lee, 'Non-intrusive load disaggregation with adaptive estimations of devices main power effects and two-way interactions', *Energy Build.*, vol. 130, pp. 131–139, Oct. 2016, doi: 10.1016/j.enbuild.2016.08.050.

[121] M. Aiad and P. H. Lee, ‘Unsupervised approach for load disaggregation with devices interactions’, *Energy Build.*, vol. 116, pp. 96–103, Mar. 2016, doi: 10.1016/j.enbuild.2015.12.043.

[122] C. Krull, M. Thiel, and G. Horton, ‘Testing Applicability of Virtual Stochastic Sensors for Non-Intrusive Appliance Load Monitoring’, *Electron. Notes Theor. Comput. Sci.*, vol. 337, pp. 119–134, May 2018, doi: 10.1016/j.entcs.2018.03.037.

[123] A. Cominola, M. Giuliani, D. Piga, A. Castelletti, and A. E. Rizzoli, ‘A Hybrid Signature-based Iterative Disaggregation algorithm for Non-Intrusive Load Monitoring’, *Appl. Energy*, vol. 185, pp. 331–344, Jan. 2017, doi: 10.1016/j.apenergy.2016.10.040.

[124] M. Azaza and F. Wallin, ‘Evaluation of classification methodologies and Features selection from smart meter data’, *Energy Procedia*, vol. 142, pp. 2250–2256, Dec. 2017, doi: 10.1016/j.egypro.2017.12.626.

[125] J. Liao, G. Elafoudi, L. Stankovic, and V. Stankovic, ‘Non-intrusive appliance load monitoring using low-resolution smart meter data’, in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Nov. 2014, pp. 535–540, doi: 10.1109/SmartGridComm.2014.7007702.

[126] T. Sirojan, B. T. Phung, and E. Ambikairajah, ‘Deep Neural Network Based Energy Disaggregation’, in *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, Aug. 2018, pp. 73–77, doi: 10.1109/SEGE.2018.8499441.

[127] M. Aiad and P. H. Lee, ‘Non-intrusive monitoring of overlapping home appliances using smart meter measurements’, in *2017 IEEE Power and Energy Conference at Illinois (PECI)*, Feb. 2017, pp. 1–5, doi: 10.1109/PECI.2017.7935717.

[128] M. Azaza and F. Wallin, ‘Supervised household’s loads pattern recognition’, in *2016 IEEE Electrical Power and Energy Conference (EPEC)*, Oct. 2016, pp. 1–5, doi: 10.1109/EPEC.2016.7771718.

[129] M. Nardello, M. Rossi, and D. Brunelli, ‘A low-cost smart sensor for non intrusive load monitoring applications’, in *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, Jun. 2017, pp. 1362–1368, doi: 10.1109/ISIE.2017.8001444.

[130] H. G. C. P. Dinesh, P. H. Perera, G. M. R. I. Godaliyadda, M. P. B. Ekanayake, and J. B. Ekanayake, ‘Individual power profile estimation of residential appliances using low frequency smart meter data’, in *2015 IEEE 10th International Conference on Industrial and Information Systems (ICIIS)*, Dec. 2015, pp. 140–145, doi: 10.1109/ICIINFS.2015.7399000.

[131] C. Dinesh, B. W. Nettasinghe, R. I. Godaliyadda, M. P. B. Ekanayake, J. Ekanayake, and J. V. Wijayakulasooriya, ‘Residential Appliance Identification Based on Spectral Information of Low Frequency Smart Meter Measurements’, *IEEE Trans. Smart Grid*, vol. 7, no. 6, pp. 2781–2792, Nov. 2016, doi: 10.1109/TSG.2015.2484258.

[132] H. G. C. P. Dinesh, P. H. Perera, G. M. R. I. Godaliyadda, M. P. B. Ekanayake, and J. B. Ekanayake, ‘Residential appliance monitoring based on low frequency smart meter measurements’, in *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Nov. 2015, pp. 878–884, doi: 10.1109/SmartGridComm.2015.7436412.

[133] S. Singh and A. Majumdar, ‘Deep Sparse Coding for Non–Intrusive Load Monitoring’, *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4669–4678, Sep. 2018, doi: 10.1109/TSG.2017.2666220.

[134] V. Singhal, J. Maggu, and A. Majumdar, ‘Simultaneous Detection of Multiple Appliances From Smart-Meter Measurements via Multi-Label Consistent Deep Dictionary Learning and Deep Transform Learning’, *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2969–2978, May 2019, doi: 10.1109/TSG.2018.2815763.

[135] A. Miyasawa, M. Matsumoto, Y. Fujimoto, and Y. Hayashi, ‘Energy disaggregation based on semi-supervised matrix factorization using feedback information from consumers’, in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Sep. 2017, pp. 1–6, doi: 10.1109/ISGTEurope.2017.8260211.

[136] L. Stankovic, V. Stankovic, J. Liao, and C. Wilson, ‘Measuring the energy intensity of domestic activities from smart meter data’, *Appl. Energy*, vol. 183, pp. 1565–1580, Dec. 2016, doi: 10.1016/j.apenergy.2016.09.087.
[137] D. C. Mocanu, E. Mocanu, P. H. Nguyen, M. Gibescu, and A. Liotta, ‘Big IoT data mining for real-time energy disaggregation in buildings’, in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2016, pp. 003765–003769, doi: 10.1109/SMC.2016.7844820.
[138] B. Huang, M. Knox, K. Bradbury, L. M. Collins, and R. G. Newell, ‘Non-intrusive load monitoring system performance over a range of low frequency sampling rates’, in *2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA)*, Nov. 2017, pp. 505–509, doi: 10.1109/ICRERA.2017.8191111.
[139] G. C. Koutitas and L. Tassiulas, ‘Low Cost Disaggregation of Smart Meter Sensor Data’, *IEEE Sens. J.*, vol. 16, no. 6, pp. 1665–1673, Mar. 2016, doi: 10.1109/JSEN.2015.2501422.
[140] G. Zhang, G. Wang, H. Farhangi, and A. Palizban, ‘Residential electric load disaggregation for low-frequency utility applications’, in *2015 IEEE Power Energy Society General Meeting*, Jul. 2015, pp. 1–5, doi: 10.1109/PESGM.2015.7286502.
[141] A. Kipping and E. Trømborg, ‘Modeling and disaggregating hourly electricity consumption in Norwegian dwellings based on smart meter data’, *Energy Build.*, vol. 118, pp. 350–369, Apr. 2016, doi: 10.1016/j.enbuild.2016.02.042.
[142] J. Ponoćko and J. V. Milanović, ‘Forecasting Demand Flexibility of Aggregated Residential Load Using Smart Meter Data’, *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5446–5455, Sep. 2018, doi: 10.1109/TPWRS.2018.2799903.
[143] O. Kimura, H. Komatsu, K. Nishio, and T. Mukai, ‘A prototype tool for automatically generating energy-saving advice based on smart meter data’, *Energy Effic.*, vol. 11, no. 5, pp. 1247–1264, Jun. 2018, doi: 10.1007/s12053-018-9639-3.
[144] F. Luo, G. Ranzi, W. Kong, Z. Y. Dong, S. Wang, and J. Zhao, ‘Non-intrusive energy saving appliance recommender system for smart grid residential users’, *Transm. Distrib. IET Gener.*, vol. 11, no. 7, pp. 1786–1793, 2017, doi: 10.1049/iet-gtd.2016.1615.
[145] J. M. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, ‘Sustainable Homecare Monitoring System by Sensing Electricity Data’, *IEEE Sens. J.*, vol. 17, no. 23, pp. 7741–7749, Dec. 2017, doi: 10.1109/JSEN.2017.2713645.
[146] M. F. Adiatmoko, A. Soeprijanto, M. Syai’in, and R. N. Hananur, ‘Smart meter based on time series modify and constructive backpropagation neural network’, in *2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, Oct. 2017, pp. 147–153, doi: 10.1109/ICITACEE.2017.8257693.
[147] S. R. Arrachman, M. F. Adiatmoko, A. Soeprijanto, M. Syai’in, M. S. A. Sidik, and N. H. Rohiem, ‘Smart meter based on time series modify and extreme learning machine’, in *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro--Mechanical System, and Information Technology (ICACOMIT)*, Oct. 2017, pp. 86–92, doi: 10.1109/ICACOMIT.2017.8253392.
[148] M. Nardello, M. Rossi, and D. Brunelli, ‘An innovative cost-effective smart meter with embedded non intrusive load monitoring’, in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Sep. 2017, pp. 1–6, doi: 10.1109/ISGTEurope.2017.8260242.
[149] K. A. Agyeman, S. Han, and S. Han, ‘Real-Time Recognition Non-Intrusive Electrical Appliance Monitoring Algorithm for a Residential Building Energy Management System’, *Energies*, vol. 8, no. 9, pp. 9029–9048, Sep. 2015, doi: 10.3390/en8099029.
[150] H. Altrabalsi, V. Stankovic, J. Liao, L. Stankovic, and Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, UK, ‘Low-complexity energy disaggregation using appliance load modelling’, *AIMS Energy*, vol. 4, no. 1, pp. 1–21, 2016, doi: 10.3934/energy.2016.1.1.

[151] M. Azaza and F. Wallin, ‘Finite State Machine Household’s Appliances Models for Non-intrusive Energy Estimation’, *Energy Procedia*, vol. 105, pp. 2157–2162, May 2017, doi: 10.1016/j.egypro.2017.03.609.
[152] S. Su *et al.*, ‘Non-intrusive load monitoring of air conditioning using low-resolution smart meter data’, in *2016 IEEE International Conference on Power System Technology (POWERCON)*, Sep. 2016, pp. 1–5, doi: 10.1109/POWERCON.2016.7753952.
[153] N. Pathak, D. Lachut, N. Roy, N. Banerjee, and R. Robucci, ‘Non-Intrusive Air Leakage Detection in Residential Homes’, in *Proceedings of the 19th International Conference on Distributed Computing and Networking*, New York, NY, USA, 2018, pp. 31:1–31:10, doi: 10.1145/3154273.3154345.
[154] A. A. Munshi and Y. A. I. Mohamed, ‘Unsupervised Nonintrusive Extraction of Electrical Vehicle Charging Load Patterns’, *IEEE Trans. Ind. Inform.*, vol. 15, no. 1, pp. 266–279, Jan. 2019, doi: 10.1109/TII.2018.2806936.
[155] A. Shaw and B. P. Nayak, ‘Electric vehicle charging load filtering by power signature analysis’, in *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)*, Feb. 2017, pp. 71–75, doi: 10.1109/ICDMAI.2017.8073488.
[156] S. Welikala, C. Dinesh, M. P. B. Ekanayake, R. I. Godaliyadda, and J. Ekanayake, ‘Incorporating Appliance Usage Patterns for Non-Intrusive Load Monitoring and Load Forecasting’, *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 448–461, Jan. 2019, doi: 10.1109/TSG.2017.2743760.
[157] M. Wurm and V. C. Coroamă, ‘Poster abstract: grid-level short-term load forecasting based on disaggregated smart meter data’, *Comput. Sci. - Res. Dev.*, vol. 33, no. 1, pp. 265–266, Feb. 2018, doi: 10.1007/s00450-017-0374-3.
[158] J. Wang *et al.*, ‘A Two-Step Load Disaggregation Algorithm for Quasi-static Time-series Analysis on Actual Distribution Feeders’, in *2018 IEEE Power Energy Society General Meeting (PESGM)*, Aug. 2018, pp. 1–5, doi: 10.1109/PESGM.2018.8586131.
[159] Y. Xu and J. V. Milanović, ‘Artificial-Intelligence-Based Methodology for Load Disaggregation at Bulk Supply Point’, *IEEE Trans. Power Syst.*, vol. 30, no. 2, pp. 795–803, Mar. 2015, doi: 10.1109/TPWRS.2014.2337872.

SINTEF

Technology for a better society

www.sintef.no