A Machine Learning-based framework and open-source software for Non Intrusive Water Monitoring

Gross, Marie-Philine; Taormina, Riccardo; Cominola, Andrea

DOI
10.1016/j.envsoft.2024.106247

Publication date
2024

Document Version
Final published version

Published in
Environmental Modelling and Software

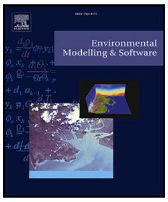
Citation (APA)
Gross, M.-P., Taormina, R., & Cominola, A. (2024). A Machine Learning-based framework and open-source software for Non Intrusive Water Monitoring. *Environmental Modelling and Software*, *183*, Article 106247. https://doi.org/10.1016/j.envsoft.2024.106247

Environmental Modelling and Software 183 (2025) 106247

Contents lists available at ScienceDirect

Environmental Modelling and Software

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



A Machine Learning-based framework and open-source software for Non Intrusive Water Monitoring

Marie-Philine Gross [a,b], Riccardo Taormina [c], Andrea Cominola [a,b,*]

[a] Chair of Smart Water Networks, Technische Universität Berlin, Straße des 17. Juni 135, Berlin, 10623, Germany
[b] Einstein Center Digital Future, Wilhelmstraße 67, Berlin, 10117, Germany
[c] Delft University of Technology, Mekelweg 5, Delft, 2628 CD, The Netherlands

ARTICLE INFO

Software and Dataset link: https://github.com/SWN-group-at-TU-Berlin/PyNIWM, https://www.waterrf.org/research/projects/residential-end-uses-water-version-2

Keywords:
Non Intrusive Water Monitoring
Water end-use classification
Machine learning

ABSTRACT

Recent research highlights the potential of consumption-based feedback for water conservation, emphasizing the need for Non Intrusive Water Monitoring (NIWM). However, existing NIWM studies often rely on small datasets, a pre-selected class of models, and inaccessible software. Here, we introduce PyNIWM, a machine learning-based open-source Python framework for NIWM. PyNIWM enables water end-use classification via (i) data characterization and feature engineering, (ii) water end-use event classification with four machine learning classifiers, and (iii) performance assessment. We demonstrate PyNIWM on a real-world dataset containing around 800,000 labeled end-use events from 762 homes across the USA and Canada. The four PyNIWM classifiers achieve F1 scores above 0.85, indicating high suitability for water end-use classification. However, a tradeoff between accuracy and computational cost exists. Finally, data balancing through oversampling enhances classification of low-represented end-use classes, but does not improve overall classification. We release PyNIWM as an open-source software, aiming for collaborative and reproducible research.

1. Introduction

Water demand-side management (DSM) strategies have gained increasing popularity in urban water management to complement supply-side operations and foster water conservation (Abu-Bakar et al., 2021). As water utilities embrace digital technologies such as advanced metering infrastructure (AMI) (Cominola et al., 2015), customer-centered services are becoming one of the core elements of their digitalization journey (Stewart et al., 2018; Boyle et al., 2022; Daniel et al., 2023). Furthermore, there is growing evidence about the potential of consumption-based feedback and other demand management options to foster water conservation behavior both in the short- and long-term (Cominola et al., 2021). High-resolution data collected at the scale of individual households with a sub-daily sampling frequency can play a pivotal role in delivering detailed information to support demand-side management programs, encourage more sustainable water uses, detect anomalies (e.g., leakages, faulty meters), and potentially providing detailed input to network-scale models of cyber–physical water systems (Taormina et al., 2018). This has motivated recent research targeting detailed understanding of the characteristics of water demands, down to different water end uses (e.g., showering, outdoor usage) (Mazzoni et al., 2022), along with identifying the socio-demographic, technological, climate, and behavioral determinants of water demand and their dynamics over time (Cominola et al., 2023).

Extracting information from single-point digital meters on how household water users use water across many end uses requires advanced and proper analytics techniques, as installing individual meters for each fixture is neither economically convenient nor socially acceptable (Cardell-Oliver et al., 2024). Since the seminal study on *Nonintrusive appliance load monitoring* by Hart (1992), the problem of estimating the contribution of individual end uses to the aggregate household consumption – called energy disaggregation or Non-Intrusive Load Monitoring (NILM) – has been studied for more than 30 years in the electricity field. Several approaches have been developed in the literature to address the NILM problem (Zoha et al., 2012; Schirmer and Mporas, 2022), yielding also the development of open-source toolkits for algorithm testing and benchmarking (Batra et al., 2014) and shared data (e.g., Kolter and Johnson, 2011; Makonin et al., 2013). NILM can be formulated either as a classification problem, where the goal is to label the operating state of individual appliances over time, or a regression problem, where the goal is to estimate the power consumption of each appliance at a given time (Precioso and Gómez-Ullate, 2020).

Conversely, the similar problem of disentangling which end-use activities occur over time in a household is still an open research challenge in the water sector. We here call this problem *Non-Intrusive Water Monitoring* (NIWM). The general aim of NIWM is similar to

[*] Corresponding author at: Chair of Smart Water Networks, Technische Universität Berlin, Straße des 17. Juni 135, Berlin, 10623, Germany.
E-mail addresses: marie-philine.gross@tu-berlin.de (M. Gross), R.Taormina@tudelft.nl (R. Taormina), andrea.cominola@tu-berlin.de (A. Cominola).

https://doi.org/10.1016/j.envsoft.2024.106247
Received 29 January 2024; Received in revised form 21 June 2024; Accepted 11 October 2024
Available online 18 October 2024

NILM in the electricity sector. However water and electricity meter signals of single-family houses are substantially different as for two main aspects. First, the total power load of a household is generally always positive over time due, e.g., to the operation of refrigerators, heating, ventilation, and air conditioning (HVAC), and plugs (Makonin et al., 2013). Several periods of zero water flow, instead, can occur during a day, e.g., when nobody is at home and no programmable fixtures are in operation (Mazzoni et al., 2021). Second, many studies concluded that, while still relevant, concurring water use activities in single family houses account for a relatively small proportion of all water use activities (Attallah et al., 2023). Concurrent events (e.g., a toilet flush occurring simultaneously to irrigation) are primarily observed simultaneously to long events such as outdoor irrigation, but many water use activities occur in isolation (Mazzoni et al., 2022). A household power signal, instead, shows a great deal of activities happening simultaneously (Hart, 1992).

Acknowledging the above characteristics of a typical water consumption signal recorded by a digital meter for single-family households, NIWM is typically tackled in two sequential stages. First, a *disaggregation* phase aims at separating concurring water use events (Bethke et al., 2021). This is followed by *end-use classification*, which assigns an end-use label (e.g., faucet, toilet, shower) to each end-use event resulting from the disaggregation phase (Heydari et al., 2022). Different approaches for residential water end-use data collection and NIWM have been recently proposed in the literature. However, several research and technical challenges remain open. Most early examples of software tackling at least one of the phases of NIWM, e.g., *TraceWizard* (Mayer et al., 1999) and *Autoflow* (Nguyen et al., 2015), are not available open source and require trained analysts and labor-intensive manual processing. More recent studies contributed either open water end-use data or open-source software for NIWM. However, some of them focus only on the early-stage phase of high-resolution water end-use data collection, e.g., via IoT technologies (Di Mauro et al., 2019), or on data collection, processing, storing, and accessibility (Cominola et al., 2018a; Pacheco et al., 2021; Di Mauro et al., 2022), without addressing the final end-use classification in NIWM. Other studies within this group achieve water end-use classification, but they are primarily demonstrated on small-scale datasets comprising only one household (Heydari et al., 2022), a few household from the same city/region (Attallah et al., 2021, 2023; Bastidas Pacheco et al., 2022), or synthetic data (Cominola et al., 2018b). Finally, some machine learning (ML) based approaches to end-use classification rely on a preselected class of models (see, e.g., Vitter and Webber, 2018; Heydari et al., 2022). Overall, these limitations combined with the still limited availability of large-scale open datasets with end-use ground truth labels have so far greatly limited comparative studies and benchmarking of NIWM methods (Di Mauro et al., 2021).

Here, we present the machine learning-based Non Intrusive Water Monitoring framework and software PyNIWM, which tackles the end-use classification phase in NIWM. The framework is composed of three sequential modules that perform (i) data characterization and feature engineering, (ii) water end-use event classification, and (iii) performance assessment. In this first version, we implement four ML algorithms for water end-use classification at the core of PyNIWM. Our approach features automated robust algorithm training, performance assessment via multiple quantitative classification metrics, and evaluation of the computational time required for classifier training.

The contribution of this work is three-fold. First, we enable comparative testing of different ML methods for NIWM and demonstrate PyNIWM on a dataset comprising 800,000 water end-use events, aiming for generalization. Second, we assess both end-use classification accuracy and computational time requirements, thus providing a framework for testing the scalability of different NIWM classifiers. Finally, we release PyNIWM as an open-source software available to researchers and practitioners, aiming for future use, collaborative development, and reproducible research.

2. Problem formulation and PyNIWM framework

The PyNIWM framework for Non Intrusive Water Monitoring we propose here tackles the end-use classification phase of NIWM as a supervised learning problem. We thus here assume that labeled end-use data are available for training from the disaggregation phase (when needed) and this data is organized in a tabular format as input of PyNIWM. Each tuple of the input data table refers to a i-th water end-use event and contains its associated M features $f_i = [f_{i,1}, f_{i,2}, \dots, f_{i,M}]$ (e.g., event duration, water volume, peak flow, flow mode, time-of-day, day-of-week), along with the end use class label c_i for that event. The resulting tuple for an event is thus a vector $e_i = [f_i, c_i]$. Given a set of N water end-use events $E = [e_1, e_2, \dots, e_N]$, each described by its tuple of associated features and class label, PyNIWM solves a supervised learning problem aimed at assigning pre-defined water end-use categories (e.g., shower, tap, toilet, dishwasher) to water end-use events based on their input features (e.g., event duration, water volume, peak flow). This is achieved in PyNIWM by training one or more ML classification algorithms such that:

$$\theta_j^* = \mathrm{argmin}_{\theta}[L(C_e - \hat{C}_j(\theta_j, F))] \tag{1}$$

where $C_e = [c_1, c_2, \dots, c_N]$ is the vector of observed water end-use classes for all events; $\hat{C}_j$ the corresponding vector of end-use classes estimated for all events by the j-th classifier implemented in PyNIWM, which are computed as a function of the classifier parameters θ and the end-use event feature matrix F; $L(C - \hat{C}_j)$ is the training loss accounting for the difference between observations and model estimates computed on the training dataset; and θ_j^* the vector of optimal parameters for the j-th classifier in PyNIWM.

The PyNIWM framework we propose here to solve the problem formulated in Eq. (1) and achieve automated ML-based water end-use event classification is composed of three sequential modules (see Fig. 1). In the first module – *Data Characterization & Feature Engineering* – labeled water end-use event data are pre-processed to remove outliers, perform feature engineering, and build the input database to train and test water end-use classifiers. The second module — *End-Use Classification* represents the core of PyNIWM, where different ML classifiers are implemented to classify end-use events based on the input features engineered in the previous module. In the current version of PyNIWM we implement four classification algorithms, i.e., Artificial Neural Networks, two gradient boosting methods (LightGBM and XGBoost), and a Random Forest classifier. This module can perform hyperparameter tuning with grid-search, stratified k-fold cross validation, and multiple runs for each method, starting from different seeds. Finally, in the *Performance Assessment* module, the performance of the ML algorithms used for end-use classification is comparatively analyzed, both in terms of aggregate and end-use level classification accuracy, as well as computational cost.

Each module of PyNIWM is further detailed in the following sections.

2.1. Data characterization & feature engineering

In the first module of PyNIWM – *Data Characterization & Feature Engineering* – individual water consumption events are processed with a three-fold goal. First, data cleaning: we identify and remove outliers and end-use events with lacking feature values. Possible outliers include, for instance, water use events with unrealistic values of water volume, peak flow, or duration. We rely on Tukey's fences for outlier detection (Tukey, 1977). Accordingly, $f_{i,x}$, which is the value of feature x-th for water end-use event i, is classified as an outlier if:

$$f_{i,x} \notin \left[Q_1 - k\left(Q_3 - Q_1\right), Q_3 + k\left(Q_3 - Q_1\right)\right] \tag{2}$$

where Q_1 is the 25th empirical quartile of all observations for that feature, Q_3 is the 75th empirical quartile, and $k = 1.5$. With this k value Tukey's fences approximate the 99.7% confidence interval defined for

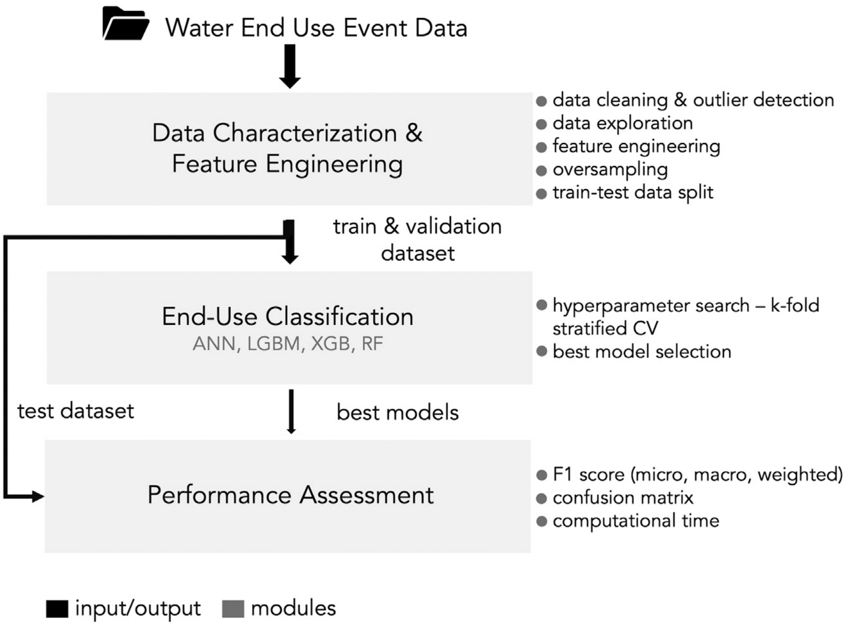


Fig. 1. Flowchart describing the three-step framework for Non Intrusive Water Monitoring implemented in PyNIWM.

normal distributions by a distance of three standard deviations from the mean (Tukey, 1977).

Second, data exploration: we characterize the input dataset of water use events with basic statistics on, e.g., the dataset size in terms of number of water use events, and count the occurrence of each water end-use class to assess whether the dataset is imbalanced. Further, we quantify pair-wise correlation among different water use event features (e.g., duration and water volume) and analyze specific features such as time-of-day or day-of-week indicating when a water end-use event happened to allow preliminary pattern discovery based on visual investigation. By extracting the hour of the day and day of the week from the timestamp, we can explore behavioral patterns for certain appliances. Finally, we jointly analyze the water volume, duration, and peak flow for each end-use category. This provides first insights into water-use behavior and what kind of appliances are used, such as faucets with aerators or high-efficiency toilets. If necessary, this analysis can also be used to split end-use classes into, e.g., old and modern appliances, to potentially increase training performance. While here we assume knowledge of end-use event labels at least for a subset of samples to be used for model training, these statistics can be computed either considering the entire sample of water end-use events, thus ignoring end-use labels, or independently for each end-use class.

Third, feature engineering and train–test split: in this module we implement additional feature engineering (i.e., feature scaling and one-hot encoding for time-related variables such as time-of-day and day-of-week), along with data splitting for model training, validation, and testing to build a suitable dataset for the *End-Use Classification* module. PyNIWM also allows investigating the influence of data imbalance on model performance by optionally balancing the water end-use classes in the training dataset before model calibration. To accomplish this task, we implement the *Synthetic Minority Oversampling Technique* (SMOTE) algorithm (Chawla et al., 2002). SMOTE oversamples the minority classes to yield a balanced dataset. Each new sample $f_{i,\text{new}}$ is computed along the vector connecting the initial sample $f_{i,\text{initial}}$ and one if its k nearest-neighbors $f_{j,\text{initial}}$ as follows:

$$f_{i,\text{new}} = f_{i,\text{initial}} + \lambda \left(f_{j,\text{initial}} - f_{i,\text{initial}} \right). \tag{3}$$

In this study a sample $f_{i,\text{initial}}$ is the vector of features for an end-use event i, λ is a random number between 0 and 1 and $k = 6$ nearest neighbors are computed based on Euclidean distance.

2.2. End-use classification

The first release of the PyNIWM framework presented here includes four state-of-the-art ML classifiers adapted for NIWM. A detailed overview on these methods is provided in the next paragraphs and in Table 1. This module of PyNIWM includes also grid-search hyperparameter search, stratified k-fold cross validation, and runs over multiple seeds to enable robust training of each classifier.

2.2.1. Random forest classifiers

Random Forests (RFs) are a ML learning ensemble technique for supervised learning. Since their first formulation in Breiman (2001), RF have been widely used across various domains, due to their suitability both for regression and classification tasks and ability to approximate complex non-linear relationships. At its core, a Random Forest is a collection of decision trees. Each tree is a non-parametric supervised learning algorithm with a hierarchical structure aimed at partitioning data into homogeneous subsets. The hierarchical structure of a decision tree is composed of *nodes*, which contain a series of control rules which splits input data based on a series of if-then conditions applied to their features, and *branches*, which connect the nodes. Each partitioning operation in a decision tree is based on an information index (e.g., the Gini index), which allows RFs to partition the data in homogeneous groups (Kuhn et al., 2013). The final nodes at the bottom level of the hierarchical structure are called leaves and are used to assign the class level. The depth of a tree increases with the complexity of the relationship the tree needs to approximate and its level of fit. During the training phase of a RF, multiple decision trees are constructed, each using a random subset of the training dataset and a different subset of features. These individual trees independently make predictions, and the ensemble combines their outputs, typically through voting or averaging, to compute the final prediction. This aggregation of predictions results in a model that is robust to overfitting, generalizes well to new data, and can handle high-dimensional datasets, making it suitable for a wide range of applications. Feature importance scores can be calculated for the input variables that are more frequently used or that bring most information to tree branching, facilitating interpretability.

Table 1
Classifiers implemented in PyNIWM for water end-use event classification.

Classifier	Acronym	Reference	Implementation
Random Forest	RF	Breiman (2001)	Pedregosa et al. (2011)
Gradient Boosting methods	LGBM	Ke et al. (2017)	Microsoft Corporation (2017)
	XGB	Chen and Guestrin (2016)	The XGBoost Contributors (2016)
Artificial Neural Networks	ANN	Battaglia et al. (2018)	Chollet et al. (2015)

2.2.2. Gradient boosting methods

Gradient Boosting Methods (GBMs) are powerful ensemble techniques where the predictive power of multiple weak learners is combined sequentially to forge a robust predictive model. While different types of weak learners can be employed, decision trees are predominantly utilized due to their effectiveness in capturing complex patterns in data. The resulting approach differs from other tree-based ensemble methods such as Random Forests, which construct numerous decision trees independently and average the predictions. Conversely, GBMs construct decision trees additively, where each tree is built to correct the errors of the previous trees.

In PyNIWM we integrate two notable GBMs: XGBoost (XGB) and LightGBM (LGBM). XGB, introduced by Chen and Guestrin (2016), builds trees iteratively, aiming to minimize an aggregate score that includes both past and current tree structures. Gradient descent then fine-tunes the leaf values of each tree. On the other hand, LGBM, introduced by Ke et al. (2017), utilizes histogram-based techniques, converting continuous features into discrete bins to expedite training. This algorithm prefers a leaf-wise tree growth, focusing on the most impactful leaf to minimize loss. This growth strategy tends to achieve lower losses than level-wise growth strategies, but at the same time tends to overfit, especially for small datasets.

2.2.3. Artificial neural networks

Artificial Neural Networks (ANNs) are ML models initially developed to mimic the functioning of the human brain. In their basic form, known as the feed-forward neural network or *multi-layer perceptron* (MLP), ANNs consist of a collection of nonlinear processing units, or *neurons*, arranged across multiple cascading interconnected *layers*. The simplest ANNs features an input layer, a single hidden layer – where most of the processing takes place – and an output layer. Each neuron in a given layer receives signals from all neurons of the previous layer, and produces an output via some non-linear activation function (e.g., sigmoidal functions or rectified linear units) of the weighted sum of these inputs. The *weights* associated with the connections between neurons represent the parameters to optimize when *training* the ANN. In supervised ML, ANN training generally involves the minimization of a loss function, e.g., a suitable estimate of the error between its predicted values at the output layer and observed target values in the training data. Variants of stochastic gradient descent are used to perform the minimization process, where the gradients of the weights in the inner layers are computed via the chain rule of derivation (e.g., back-propagation).

ANNs have been widely employed due to their universal approximation capabilities. In the last decade, Deep Learning methods – generally referred to as ANNs with more than one hidden layer, usually incorporating either spatial, temporal, and relational inductive biases (Battaglia et al., 2018; Chollet, 2021) – have revolutionized most fields of science and technology, with substantial contributions in water resources management and engineering (Kratzert et al., 2018; Zounemat-Kermani et al., 2020; Bentivoglio et al., 2022; Garzón et al., 2022). In PyNIWM we implement MLPs with varying number of hidden layers to perform the water end use classification task. We use early-stopping and add dropout layers (Gal and Ghahramani, 2016) to improve the generalization ability of the models, which are all trained with the Adam stochastic optimization algorithm (Kingma and Ba, 2014).

2.3. Performance assessment

In the last module of PyNIWM we assess and compare the performance of the different ML classifiers for water end-use classification using a thorough set of quantitative metrics commonly used for classification problems. In particular, we resort to different multi-class versions of the *F1-score* (F1). For a binary classification problem, F1 is defined as the harmonic mean of *precision* and *recall*:

$$F1\,score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

Precision and Recall are formulated in the two equations below:

$$Precision = \frac{TP}{TP + FP} \tag{5a}$$

$$Recall = \frac{TP}{TP + FN} \tag{5b}$$

Precision is defined as the ratio between true positives (TP), i.e., correctly classified positive instances, and the overall amount of positives predictions returned by the model (including both TP and false positives (FP), i.e., negative instances incorrectly classified as positive cases). Recall quantifies the ratio between the total number of correctly classified positive samples (TP) and the amount of positive cases contained in the dataset (including both TP and false negatives (FN), i.e., positive instances incorrectly classified as negative cases).

When dealing with multi-class classification, aggregate F1 metrics are obtained by calculating them over the entire sample or by combining the individual scores obtained for each class. Three averaging methods are usually employed, yielding three different metrics known as the *macro-*, *micro-*, and *weighted-* F1. The macro-averaged F1 (macro-F1) is computed by simply taking the arithmetic mean of the per-class F1 scores obtained for each class:

$$\text{macro-}F1 = \frac{1}{N}\sum_{i=1}^{N} F1_i \tag{6}$$

where i is the class index and N is the number of classes. This method treats all classes equally regardless of their support values, i.e., the number of actual occurrences of the class in the dataset. Therefore, this metric reflects well the overall performances when working with a balanced dataset, but it can be affected significantly by the performances on small classes.

Micro averaging computes a global average F1 by counting the sums of TP, FP, and FN across the entire dataset and plugging these values in Eqs. (5) and (4). This version of the F1 is useful when reporting an overall classification performance regardless of the class, and it is employed for both balanced and imbalanced datasets. Micro-F1 offers a global overview that might still mask performance issues on minority classes.

Finally, the weighted-averaged F1 (weighted-F1) is calculated by taking the weighted mean of all per-class F1 scores, where the weights refer to the proportion of each class' support $|i|$ relative to the overall cardinality M of the dataset, as follows:

$$\text{weighted-}F1 = \frac{1}{M}\sum_{i=1}^{N} |i| \times F1_i \tag{7}$$

where $M = \sum_{i}^{N} |i|$. The weighted aggregation renders this metric better suited for performance assessment on imbalanced datasets. However, the weighted-F1 may disproportionately reflect the performance of the larger classes, potentially masking poor performance on small ones.

We implement all three F1 formulations in PyNIWM. In addition to these aggregate metrics, we analyze the end-use classification results for each method by visual inspection of its confusion matrix, which reports how many predictions are correct/incorrect for each water end-use class.

Beside the above classification accuracy metrics, we also assess the performance of each classifier by recording its computational time, i.e., the amount of time needed by each algorithm for a complete model training run.

3. Data and experimental settings

In this study, we demonstrate PyNIWM and assess its capability to perform water end-use classification using input data from the event log from the Residential End Uses of Water Study, Version 2 (REU2016; DeOreo et al., 2016). The water end-use sample gathered within the REU2016 study originally comprises water flow data from 762 homes spread across nine cities in the United States and Canada. Residential water consumption was monitored for each home with a single-point smart meter, which recorded water flow data with a sampling resolution of 10 s over a two-week period. Different monitoring campaigns were run sequentially in 9 selected cities in the USA and Canada, hence it took nearly a whole year to gather all data. The authors then disaggregated the recorded flow signal into 13 end-use categories by means of the flow trace analysis tool *Trace Wizard* (DeOreo et al., 1996) in combination with expert evaluation. The resulting dataset contains nearly 3 million labeled water end-use events, each characterized by the following six features: duration, volume, flow peak, mode, time of day, and day of the week. A more detailed summary of its metadata, along with a definition of each water end-use event feature is provided in Tab. S1.

Here we retain approximately 800,000 labeled water end-use events from the data disaggregated in the REU2016 study, after further preprocessing and checking for its completeness and consistency as described in the *Data Characterization & Feature Engineering* step of PyNIWM (see Fig. 1). We then split the dataset and use 75% of the data points to train the ML models embedded in PyNIWM and the remaining 25% for model testing. As the first module in PyNIWM allows for optional data balancing, we create two experimental scenarios for model training: a *balanced* scenario, where we use SMOTE to balance the training dataset and an *imbalanced* scenario, where we use the original processed data without further balancing. Finally, we perform hyperparameter search and model training by first creating a discretized parameter grid for each ML algorithm embedded in PyNIWM for end-use classification. We consider the most relevant parameters for each algorithm in building the parameter grid, e.g., the size and amount of layers for ANNs, and the amount and depth of trees for tree-based RF models. As a result, between 36 and 96 parameter combinations are investigated for each classifier. We implement a stratified k-fold cross-validation (k=5) routine to train each algorithm. Stratified k-fold cross-validation ensures that each fold has the same proportion of the different water end-use classes, thus avoiding inconsistencies among different model training and testing runs. A detailed summary of the parameters we tune via stratified k-fold cross-validation, along with the parameter values/ranges used for grid search is reported in Tab. S2. We execute ten different model training runs, each starting from a different random seed.

In terms of software implementation, we rely on major open-source libraries for coding the PyNIWM algorithms for water end-use classification. More in detail, we choose the Keras implementation for ANNs (Chollet et al., 2015); for the GBM we rely on two implementations, namely LGBM and XGB (Microsoft Corporation, 2017; The XGBoost Contributors, 2016); and for the RF we use the sklearn implementation (Pedregosa et al., 2011). The computational times reported for all model runs in this paper are based on the performance of a workstation with AMD Ryzen 9 3950X CPU, Nvidia GeForce RTX 3090 GPU and 64 GB of RAM.

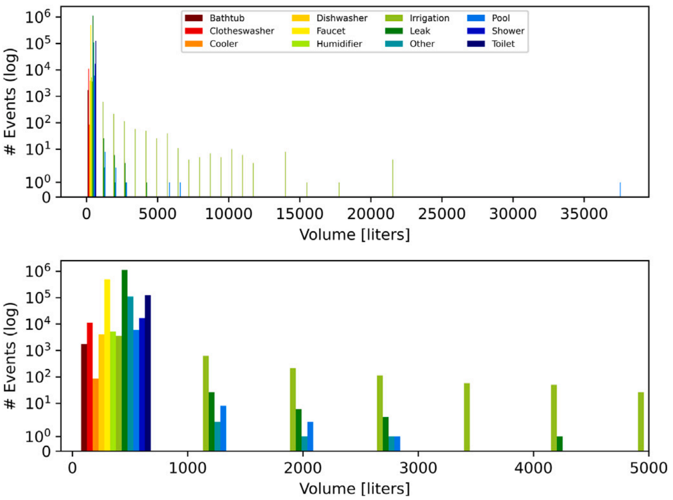


Fig. 2. Water end-use event occurrences by type of end use (color) and associated event volume for the data retained from the REU2016 dataset after outlier exclusion (DeOreo et al., 2016) and exclusion of the *treatment* class that was represented only with one event. The bar plot on the bottom zooms into the one on the top, reducing the range of water end-use event volume between 0 and 5000 liters/event for better visualization.

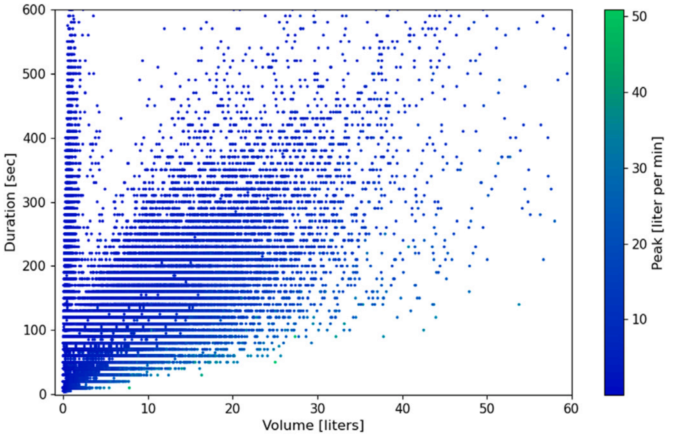


Fig. 3. Correlation analysis of event volume, duration, and peak for faucet water use events.

4. Results

4.1. Data characterization & feature engineering

Data processing and preliminary analysis resulting from the first module of PyNIWM enables gaining a deeper understanding of the structure and characteristics of the dataset, as well as intuitions on which features might be relevant for the end-use classification task. First, Fig. 2 shows that the distribution of end uses across the different classes is highly imbalanced in our dataset. This may be expected, considering some water fixtures are typically used more often throughout the day than others (e.g., faucet). Based on the outcomes of this preliminary analysis, we exclude from further end-use classification two end-use classes from the original dataset, i.e., *treatment* – because it only contains one usage event and, for this reason, is omitted from representation in Fig. 2 – and *leak* due to the inconsistent definition of leak events and their extraction from the smart meter signal during signal disaggregation.

Second, we discover that sub-groups of fixtures within the same end-use class may exhibit different characteristics. In Fig. 3, for instance, the

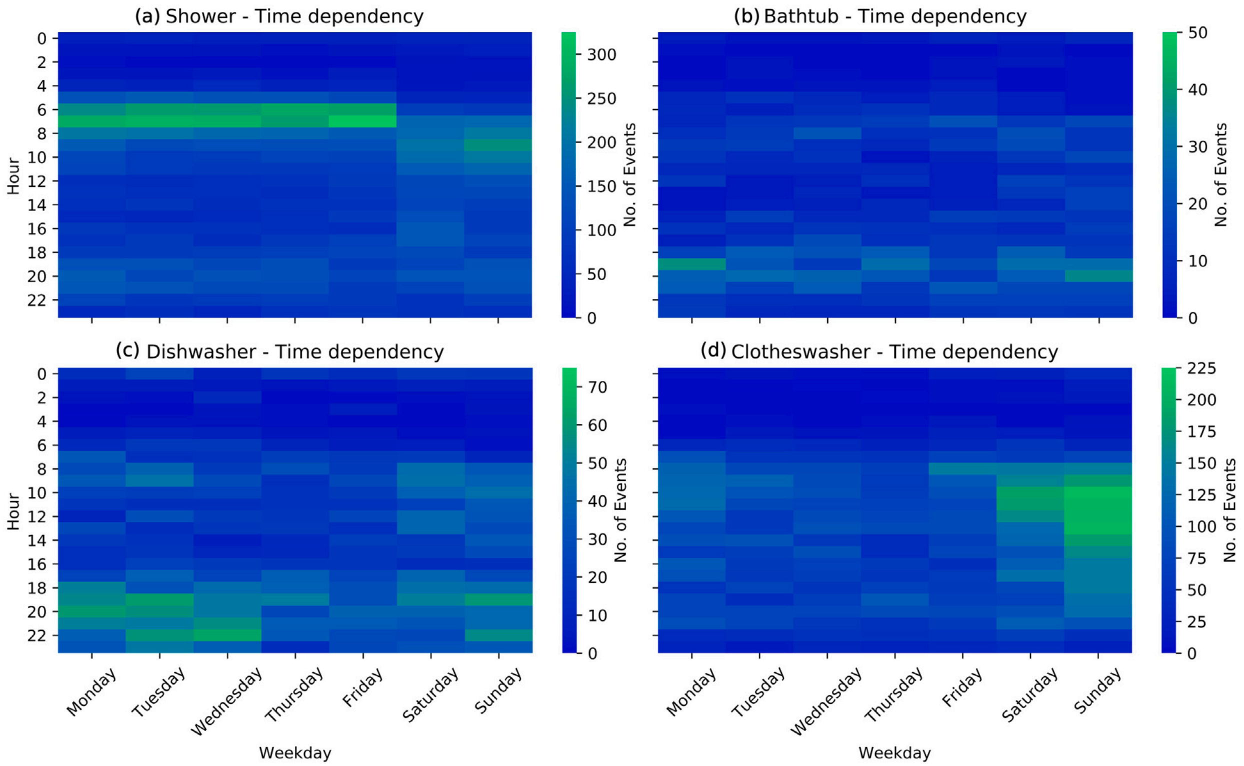


Fig. 4. Time-of-day and day-of-week analysis for (a) shower, (b) bathtub, (c) dishwasher and (d) clotheswasher events.

low peak flow values of most faucet events suggest that faucet aerators are used. Based on the scattered distribution of *faucet* event duration and volume, we split the faucet end-use class into a *high volume-short duration* (HVSD) faucet class and a *low volume-long duration* (LVLD) faucet class. With this division we try to split the class in two separate, distinct classes, so the different faucet use profiles do not overlay.

Finally, time-of-day and day-of-week analysis reveals interesting behavioral patterns for some appliances. For instance a weekday and weekend routine can be distinguished for shower events (Fig. 4(a)). Shower events occur typically in the early morning during weekdays, highlighting typical working hours in northern America from 9 to 17 o'clock. In contrast weekend shower routines are less regular and depict a delayed start time compared to the weekday observations. Bathtubs are mostly counted during late afternoon and evening hours (see Fig. 4(b)). There also is a discernible trend in Fig. 4(d), as most clothes washing events take place during the weekend. Dishwasher events (Fig. 4(c)) are mostly observed during the late afternoon and evening. Also, these routines appear to be more regular on specific days (e.g., Mon–Wed and Sun) and can happen late at night, which suggests the use of programmable devices.

4.2. End-use classification for NIWM

4.2.1. Water end-use classification accuracy

The aggregate water end-use classification performance attained by the PyNIWM classifiers on the test dataset is reported in Fig. 5. Overall, all algorithms achieve micro-F1 and weighted-F1 values close to or slightly above 0.9, indicating high accuracy in water end-use event classification. For all F1 formulations – micro, macro, and weighted – XGB achieves the best scores both in the data imbalanced and data balanced (SMOTE) scenarios. It achieves a micro-F1 of 0.91 when trained on the original data without preliminary data balancing, closely followed by RF, LGBM is less then 1% apart, while ANNs achieve a micro-F1 lower than 0.9.

The performance of all algorithms drops substantially when quantified by macro-F1, with most values of macro-F1 falling below 0.5.

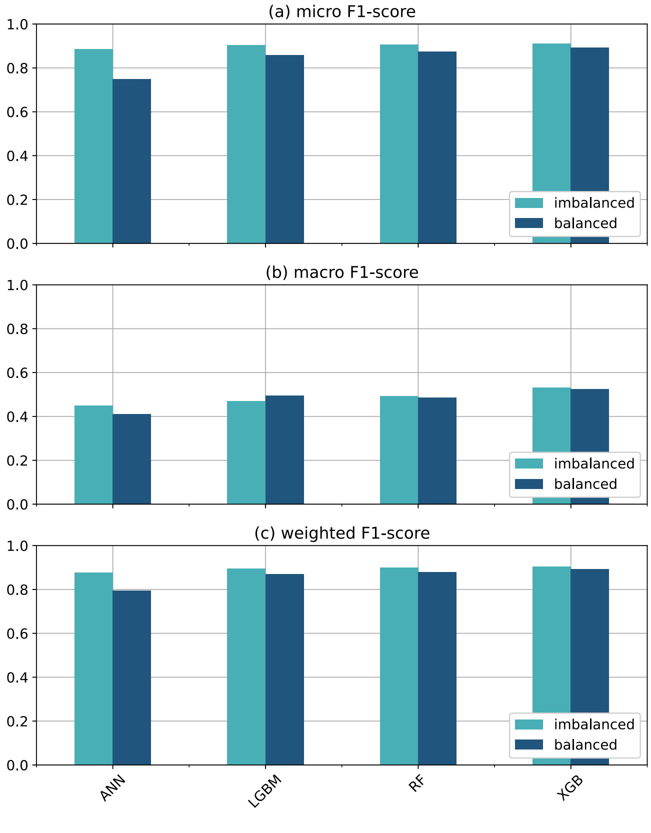


Fig. 5. Performance assessment of the four ML classifiers implemented in PyNIWM for water end-use classification. Three formulations of the F-score are quantified for each algorithm and reported for algorithm testing: micro-F1 (top), macro-F1 (middle), weighted-F1 (bottom). Algorithm performance is assessed for both a scenario with imbalanced data (cyan) and balanced data across water end-use classes (blue).

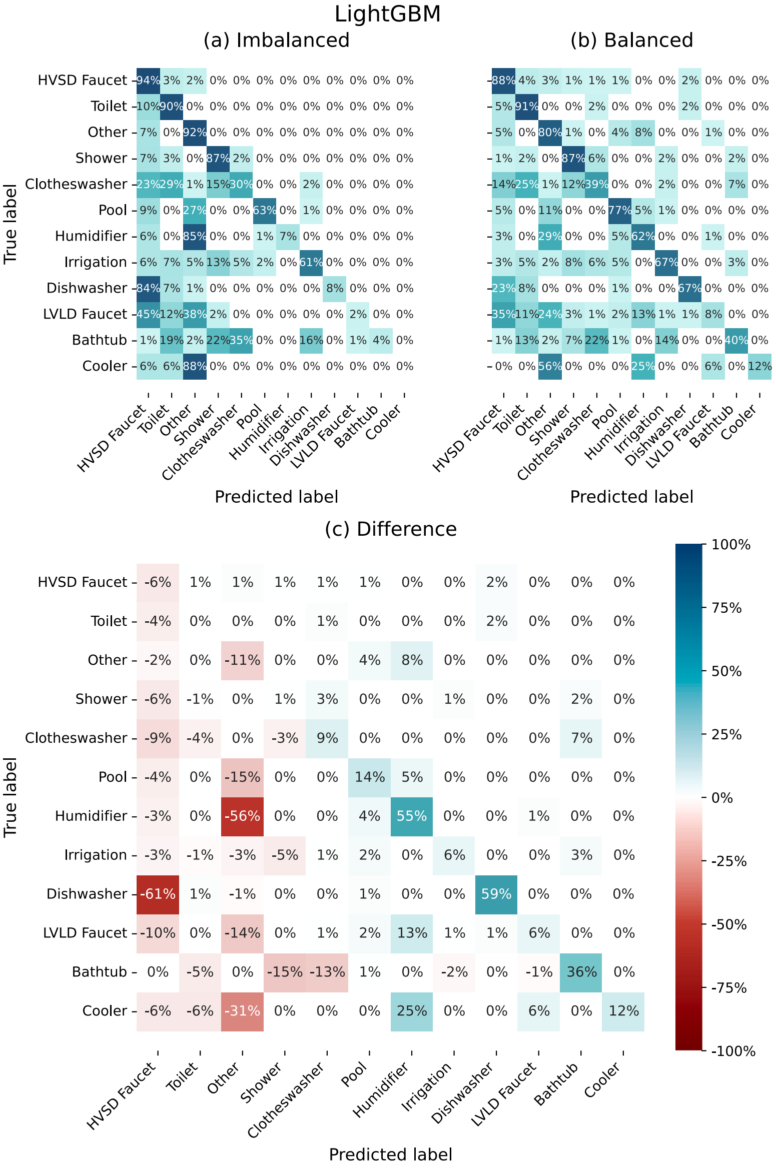


Fig. 6. Confusion matrix obtained for LGBM on imbalanced data (a), balanced data (b), and the difference between the two scenarios (c). In each matrix rows show the actual classes (true labels) and columns show the predicted classes for each water end use. Cell color is proportional to the percentage of events in that cell. Water end-use classes are sorted by decreasing event count (top to bottom on the *y*-axis, left to right on the x-axis).

This suggests that, first, heterogeneous accuracy levels are achieved for different end-use classes and, second, that samples in less represented classes are classified with lower accuracy. Balancing the data, though, does not seem to bring any advantage to the overall algorithm performance at the aggregate level. All PyNIWM models achieve higher classification accuracy when trained on imbalanced data in nearly all cases shown in Fig. 5, except from LGBM when assessed via macro-F1. Marginal performance differences between the two data balancing scenarios are within an interval of 4% for all algorithms except ANNs, for which we observe a drop of nearly 10%. This result is likely due to the knowledge on data structure acquired by the classification model in the training phase over imbalanced data. Data balancing is performed only on the training dataset, while validation and test data is kept imbalanced. The models trained on imbalanced data might thus have an advantage in terms of data structure learning, as the data structure of the testing dataset is comparable to the structure of the training data.

While data balancing via SMOTE does not bring remarkable advantages on aggregate performances as measured by F1, its benefits emerge when analyzing the confusion matrix reporting detailed classification accuracy for individual water end-use classes. Comparing for example the confusion matrices obtained for the LGBM classifier in the data balanced and imbalanced scenarios (Fig. 6(a) and (b)), classification results on unbalanced data achieve a very high True Positive rate for the most represented classes (upper-left area of the confusion matrix), but a very poor classification performance for the minority classes (lower-right area of the confusion matrix). This performance gap is reduced when LGBM is trained on balanced data. In this scenario, LGBM achieves slightly lower scores in the upper half of the confusion matrix, but the predictions for the minority classes improve significantly. Fig. 6(c) highlights the resulting differences in accuracy between the two data balancing scenarios by subtracting the confusion matrix obtained for the data imbalanced scenario from the one obtained for the data balanced scenario. This is especially interesting for the values on the matrix diagonal, which represent the correctly predicted labels (TP). There, a positive value indicates where the balanced model outperforms the imbalanced one and vice versa for negative values. Along the diagonal we can see that the balanced LGBM model improves the prediction for 9 out of 12 water end-use classes, with classification improvements of up to approximately 60% for the dishwasher end-use class, and still above 10% for humidifier, bathtub, pool, and cooler.

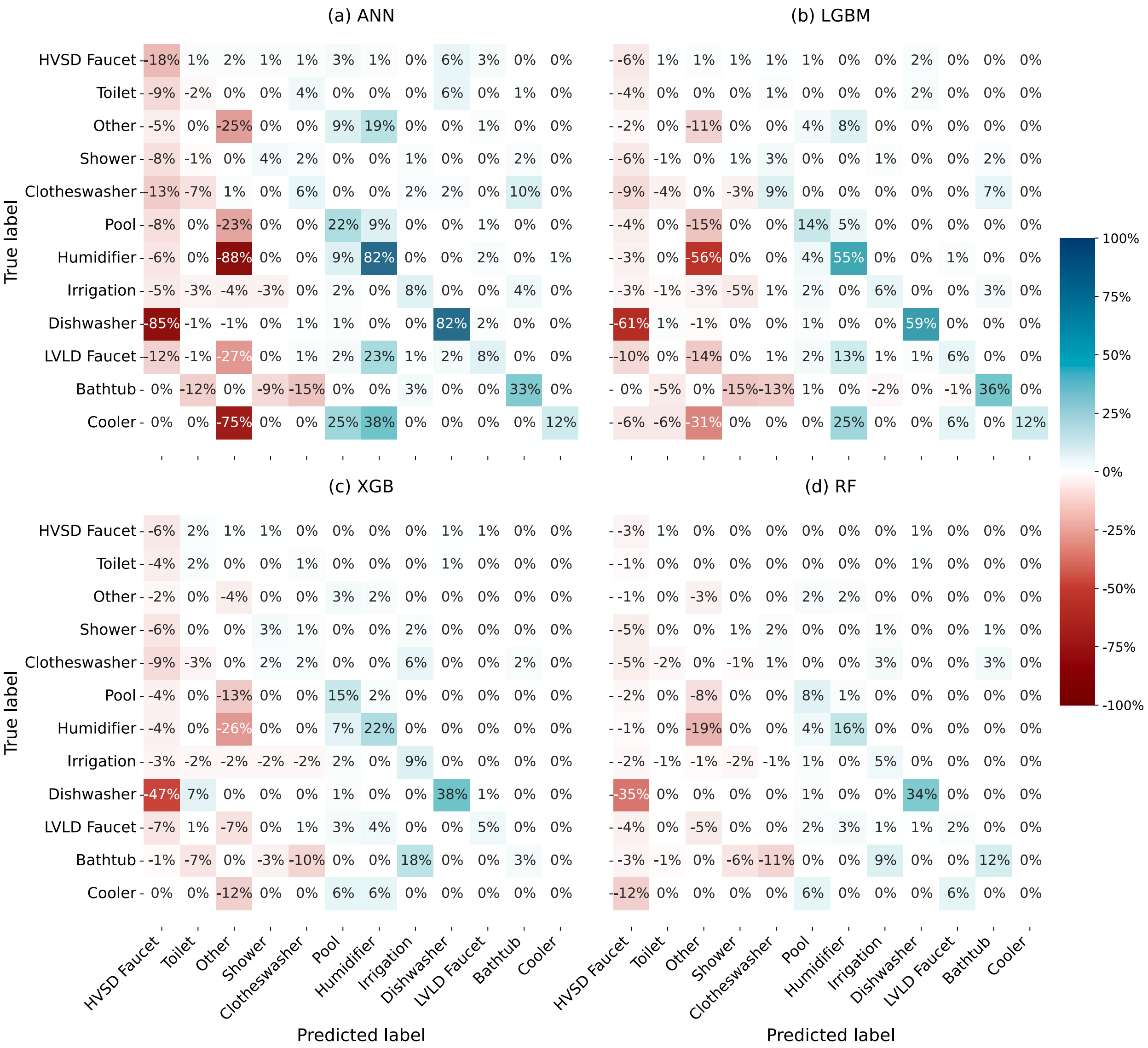


Fig. 7. Difference between the confusion matrix obtained for the balanced scenario (with SMOTE) and imbalanced scenario for each algorithm in PyNIWM.

We see comparable results for the other PyNIWM classifiers in Fig. 7. Data balancing promotes better classification results for the minority classes. Similar to LGBM, ANN, RFs, and XGB benefit most from the balancing for the end-use classes of humidifier, dishwasher, and bathtub by increasing the classification accuracy by up to 82%. Only ANNs show a substantial decrease in classification performance by up to 25% for some of the most represented classes (i.e., faucet and other). Oversampling thus comes with greater decrease in performance on the majority classes for the neural network than for the tree-based methods. In a more detailed analysis of all the confusion matrices (see Figs. 6, S1, S2, S3) we discover that all algorithms struggle to improve their classification performance on specific end uses, such as clotheswasher, shower, and cooler events. This can be attributed to the difference in cardinality of the end-use classes as well as their characteristic features overlapping with those from other end uses, such as faucet and toilet for the clotheswasher, faucet for the shower, and humidifier for the cooler. Misclassification thus happens, but towards similar end uses.

4.2.2. *Computational time for end-use classification*

Though all PyNIWM algorithms except ANNs produce similar results in terms of water end-use classification, computational time can differ substantially across the four methods (see Fig. 8). LGBM obtains nearly the same F1 values than XGB, but in only 1/6 of the time. While the values represented in Fig. 8 indicate that, in the worst case, only a few hours might be needed to train the PyNIWM classifiers on a dataset comprising about 800,000 data points and 6 features, our results are relevant to inform applications on larger datasets. Trade-offs emerge and we can differentiate between algorithms that handle large amounts of data more efficiently (i.e., LGBM and RF) and those that obtain a better classification result (i.e., XGB).

Overall the above findings on water end-use classification accuracy and computational time suggest that all tree-based methods (LGBM, RF, XGB) implemented in PyNIWM yield good results for NIWM, with LGBM providing a good trade-off given its generally accurate classification results and lower computational time than RF and XGB. This confirms that these methods outperform Deep Learning for tabular data (Shwartz-Ziv and Armon, 2022). However, the results presented here should not interpreted as a conclusive statement. A more thorough hyper-parameter search or running the algorithms on a different machine might change the performance ranking of the presented algorithms.

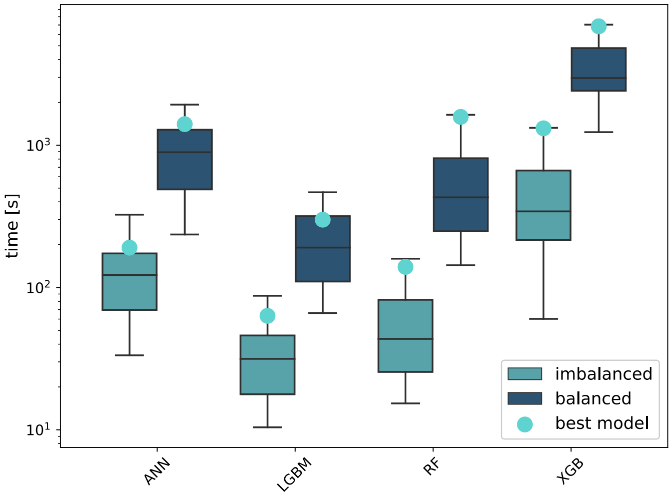


Fig. 8. Computational time for hyper-parameter search with SMOTE.

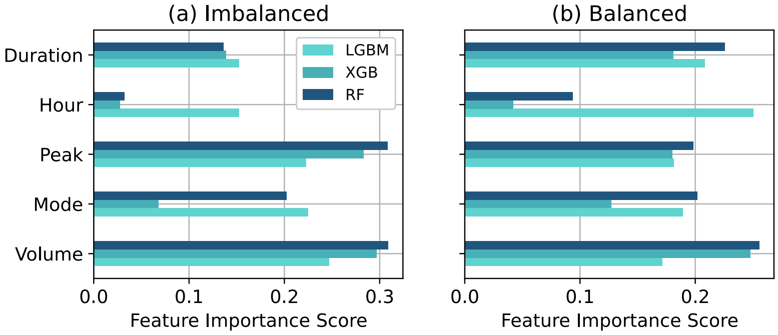


Fig. 9. Feature importance scores of the tree-based methods (RF, XGB, LGBM) for the five most important features for (a) the imbalanced and (b) the balanced dataset.

4.2.3. *Feature importance*

The results from the *Data Characterization & Feature Engineering* module revealed that patterns and differences emerge for particular features across different end-use categories, including event volume, duration, and time of day. We here analyze the feature importance calculated for tree-based classifiers after end-use classification to identify which features the algorithm mainly used to learn patterns in data. For all tree-base methods (RF, XGB, and LGBM) event water volume is the most important feature, followed by flow peak, event duration, and flow mode, but in a different order for each algorithm (see Figs. 9 and S4). The feature importance scores and order for these features are only a little influenced by the balancing of the dataset, which yields a higher importance to the hour-of-day feature. Conversely, the hour-of-day feature achieves a meaningful score higher than only with LGBM in the data-unbalanced scenario. Interestingly, the weekday features are not considered by the algorithms, even if weekday vs weekend patterns emerged from our preliminary analysis. This could be due to the extent of the data collection campaign. As mentioned before, each household was monitored for two weeks and this may not be sufficient for the data to capture representative patterns.

5. Conclusions

In this paper, we present a ML-based Non Intrusive Water Monitoring framework and its implementation PyNIWM, released as an open-source Python software. We formulate residential water end-use classification as a supervised learning problem and develop PyNIWM as a three-module software framework including (i) data characterization and feature engineering, (ii) water end-use event classification, and (iii) performance assessment. PyNIWM includes four machine learning classifiers, namely three tree-based methods – Random Forests and the gradient boosting methods LightGBM and XGBoost – and Artificial Neural Networks.

We test PyNIWM using a real-world dataset containing around 800,000 labeled end-use events collected in 762 homes across nine cities in the USA and Canada, after pre-processing (DeOreo et al., 2016) and comparatively investigate the results achieved by the different PyNIWM classifiers in terms of water end-use classification accuracy, computational time requirements, and performance over balanced/imbalanced data.

Numerical results show that, first, all ML algorithms achieve F1 scores above 0.85, demonstrating high suitability to perform the water end-use classification tasks when proper labeled water end-use data with coupled features are available for model training. The end-use classification accuracy levels achieved by LGBM, RF, and XGB classifiers are less than 1% apart, when assessed by micro- and weighted-F1 on imbalanced data. However, the performance of all algorithms drops substantially when quantified by macro-F1, because different accuracy levels are achieved for different end-use classes, with less represented classes being classified with lower accuracy. Also, computational time can differ substantially across the four methods. LGBM obtains nearly the same classification accuracy than XGB, but in only 1/6 of the time. All in all, our results suggests that some algorithms (LGBM and XGB) can handle large amounts of data more efficiently, while others (RF) obtain slightly better classification result at the price of higher computational requirements. A more complex model such as ANN with multiple hidden layers does not necessarily lead to better results. This trade-off between accuracy and computational cost can guide algorithm choice in practical applications and have implications for scalability. Finally, applying SMOTE to balance the training datasets causes a drop on overall classification performance metrics for all methods, but yields to higher ability to classify low-represented classes. This may become relevant if particular low-represented end uses are prioritized in NIWM to develop customized end use-based water consumption feedback or detect faulty fixtures and leakages.

A direct comparison of our results to similar studies is not possible, as reported metrics often are not consistent between studies and datasets differ significantly in, e.g., number of households, sampling frequency, and reporting period. Cominola et al. (2015) reported that state-of-the-art water end-use classification methods can achieve an accuracy between 74% and 94%, matching the range of our results. Comparing the more detailed classification results between our study and Heydari et al. (2022) reveals a consistently high classification accuracy for the majority classes. In Heydari et al. (2022), though, only the faucet category represents a majority class across end uses and their dataset only comprises one household. This comparison with recent water end-use classification methods further highlights the usefulness of the presented framework. PyNIWM extends the state-of-the-art literature on water end-use disaggregation and classification by leveraging machine learning, enabling comparative analysis of different classifiers, and implementing different data handling features including data balancing via oversampling techniques and robust model training to prevent overfitting.

The main limitation of this study is that currently PyNIWM has only been tested on one dataset which in the realm of water end-use studies is one of the largest. Further, it is based on US data and a different context could significantly change the Data Characterization and Feature Engineering step. From a technical viewpoint, the computational times were only obtained from one machine setup. And lastly, PyNIWM only tackles the second part of NIWM leaving the disaggregation step to the user.

There is large potential for future work to overcome the current limitations of PyNIWM for research and practice. Further collaborative developments fostered by the open-source nature of this project could prioritize practical applications aimed at testing the performance of PyNIWM on datasets from different contexts than the US. Also, hybrid datasets with data from different contexts for training and testing to assess algorithm transferability and foster transfer learning could be beneficial. Also, further investigations to check whether SMOTE undermines the calibration of predictions, meaning the predicted probabilities do not accurately reflect the true likelihood of outcomes are needed. This has been pointed by recent studies in other fields (van den Goorbergh et al., 2022). Software developments could further target stress testing on machines with different computational capabilities, datasets with different sizes, and software extensions with further classification algorithms or data sampling techniques. Finally, implementing in PyNIWM an open-source module for smart meter data disaggregation before end-use classification would make it a complete tool to resolving water end uses from smart meter data to actionable end-use characterization for water demand management.

CRediT authorship contribution statement

Marie-Philine Gross: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Riccardo Taormina:** Writing – review & editing, Writing – original draft, Validation, Supervision, Software, Resources, Methodology, Formal analysis, Conceptualization. **Andrea Cominola:** Writing – review & editing, Writing – original draft, Validation, Supervision, Resources, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to acknowledge TheWater Research Foundation for providing the data from project 4309, the Residential End Uses of Water, Version 2 survey for this study.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.envsoft.2024.106247.

Software and data availability

PyNIWM is available on GitHub as an open-source software framework and can be downloaded here: https://github.com/SWN-group-at-TU-Berlin/PyNIWM. Additional information for needed Python packages and how to use the software can also be found there. The data used in this study was obtained from the Residential End Uses of Water Study, Version 2. This database, project 4309, is managed by The Water Research Foundation and can be accessed here: https://www.waterrf.org/research/projects/residential-end-uses-water-version-2.

References

Abu-Bakar, H., Williams, L., Hallett, S.H., 2021. A review of household water demand management and consumption measurement. J. Clean. Prod. 292, 125872.

Attallah, N.A., Horsburgh, J.S., Bastidas Pacheco, C.J., 2023. An open-source, semisupervised water end-use disaggregation and classification tool. J. Water Resour. Plann. Manag. 149 (7), 04023024.

Attallah, N.A., Rosenberg, D.E., Horsburgh, J.S., 2021. Water end-use disaggregation for six nonresidential facilities in Logan, Utah. J. Water Resour. Plan. Manag. 147 (7), 05021006.

Bastidas Pacheco, C.J., Horsburgh, J.S., Attallah, N.A., 2022. Variability in Consumption and End Uses of Water for Residential Users in Logan and Providence, Utah, US. J. Water Resour. Plann. Manag. 149 (1), 05022014.

Batra, N., Kelly, J., Parson, O., Dutta, H., Knottenbelt, W., Rogers, A., Singh, A., Srivastava, M., 2014. NILMTK: An open source toolkit for non-intrusive load monitoring. In: Proceedings of the 5th International Conference on Future Energy Systems. pp. 265–276.

Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al., 2018. Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261.

Bentivoglio, R., Isufi, E., Jonkman, S.N., Taormina, R., 2022. Deep learning methods for flood mapping: a review of existing applications and future research directions. Hydrol. Earth Syst. Sci. 26 (16), 4345–4378.

Bethke, G.M., Cohen, A.R., Stillwell, A.S., 2021. Emerging investigator series: disaggregating residential sector high-resolution smart water meter data into appliance end-uses with unsupervised machine learning. Environ. Sci. Water Res. Technol. 7 (3), 487–503.

Boyle, C., Ryan, G., Bhandari, P., Law, K.M., Gong, J., Creighton, D., 2022. Digital transformation in water organizations. J. Water Resour. Plann. Manag. 148 (7), 03122001.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.

Cardell-Oliver, R., Cominola, A., Hong, J., 2024. Activity and resolution aware privacy protection for smart water meter databases. Internet Things 25, 101130.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. 16, 321–357.

Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, Association for Computing Machinery, New York, NY, USA, pp. 785–794. http://dx.doi.org/10.1145/2939672.2939785.

Chollet, F., 2021. Deep learning with Python. Simon and Schuster.

Chollet, F., et al., 2015. Keras: Deep learning for humans. URL https://keras.io.

Cominola, A., Ghetti, A., Castelletti, A., 2018a. Building an open high-resolution residential water end-use dataset with non-intrusive metering, intrusive metering, and water use diaries. In: EGU General Assembly Conference Abstracts. p. 13471.

Cominola, A., Giuliani, M., Castelletti, A., Fraternali, P., Gonzalez, S.L.H., Herrero, J.C.G., Novak, J., Rizzoli, A.E., 2021. Long-term water conservation is fostered by smart meter-based feedback and digital user engagement. npj Clean Water 4 (1), 29.

Cominola, A., Giuliani, M., Castelletti, A., Rosenberg, D.E., Abdallah, A.M., 2018b. Implications of data sampling resolution on water use simulation, end-use disaggregation, and demand management. Environ. Model. Softw. 102, 199–212.

Cominola, A., Giuliani, M., Piga, D., Castelletti, A., Rizzoli, A.E., 2015. Benefits and challenges of using smart meters for advancing residential water demand modeling and management: A review. Environ. Model. Softw. 72, 198–214.

Cominola, A., Preiss, L., Thyer, M., Maier, H., Prevos, P., Stewart, R., Castelletti, A., 2023. The determinants of household water consumption: A review and assessment framework for research and practice. npj Clean Water 6 (1), 11.

Daniel, I., Ajami, N.K., Castelletti, A., Savic, D., Stewart, R.A., Cominola, A., 2023. A survey of water utilities' digital transformation: drivers, impacts, and enabling technologies. npj Clean Water 6 (1), 51.

DeOreo, W.B., Heaney, J.P., Mayer, P.W., 1996. Flow trace analysis to access water use. J.-Am. Water Works Assoc. 88 (1), 79–90.

DeOreo, W., Mayer, P., Kiefer, J., 2016. Residential End Uses of Water, Version 2: Executive Report.. In: Water Research Foundation. 4309b.

Di Mauro, A., Cominola, A., Castelletti, A., Di Nardo, A., 2021. Urban water consumption at multiple spatial and temporal scales. A review of existing datasets. Water 13 (1), 36.

Di Mauro, A., Di Nardo, A., Santonastaso, G.F., Venticinque, S., 2019. An IoT system for monitoring and data collection of residential water end-use consumption. In: 2019 28th International Conference on Computer Communication and Networks. ICCCN, IEEE, pp. 1–6.

Di Mauro, A., Venticinque, S., Santonastaso, G.F., Di Nardo, A., 2022. WEUSEDTO—Water end USE dataset and tools: An open water end use consumption dataset and data analytics tools. SoftwareX 20, 101214.

Gal, Y., Ghahramani, Z., 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: International Conference on Machine Learning. PMLR, pp. 1050–1059.

Garzón, A., Kapelan, Z., Langeveld, J., Taormina, R., 2022. Machine learning-based surrogate modelling for urban water networks: Review and future research directions. Water Resour. Res. e2021WR031808.

van den Goorbergh, R., van Smeden, M., Timmerman, D., Van Calster, B., 2022. The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression. J. Am. Med. Informat. Assoc. 29 (9), 1525–1534.

Hart, G.W., 1992. Nonintrusive appliance load monitoring. Proc. IEEE 80 (12), 1870–1891.

Heydari, Z., Cominola, A., Stillwell, A.S., 2022. Is smart water meter temporal resolution a limiting factor to residential water end-use classification? A quantitative experimental analysis. Environ. Res. Infrastruct. Sustain. 2 (4), 045004.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y., 2017. Lightgbm: A highly efficient gradient boosting decision tree. Adv. Neural Inf. Process. Syst. 30.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kolter, J.Z., Johnson, M.J., 2011. REDD: A public data set for energy disaggregation research. In: Workshop on Data Mining Applications in Sustainability (SIGKDD), San Diego, CA, Vol. 25. Citeseer, pp. 59–62.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M., 2018. Rainfall–runoff modelling using long short-term memory (LSTM) networks. Hydrol. Earth Syst. Sci. 22 (11), 6005–6022.

Kuhn, M., Johnson, K., Kuhn, M., Johnson, K., 2013. Classification trees and rule-based models. Appl. Predict. Model. 369–413.

Makonin, S., Popowich, F., Bartram, L., Gill, B., Bajić, I.V., 2013. Ampds: A public dataset for load disaggregation and eco-feedback research. In: 2013 IEEE Electrical Power & Energy Conference. IEEE, pp. 1–6.

Mayer, P.W., DeOreo, W.B., Opitz, E.M., Kiefer, J.C., Davis, W.Y., Dziegielewski, B., Nelson, J.O., 1999. Residential end uses of water. Amer Water Works Assn USA.

Mazzoni, F., Alvisi, S., Blokker, M., Buchberger, S.G., Castelletti, A., Cominola, A., Gross, M.-P., Jacobs, H.E., Mayer, P., Steffelbauer, D.B., et al., 2022. Investigating the characteristics of residential end uses of water: A worldwide review. Water Res. 119500.

Mazzoni, F., Alvisi, S., Franchini, M., Ferraris, M., Kapelan, Z., 2021. Automated household water end-use disaggregation through rule-based methodology. J. Water Resour. Plann. Manag. 147 (6), 04021024.

Microsoft Corporation, 2017. Lightgbm. URL https://lightgbm.readthedocs.io.

Nguyen, K.A., Stewart, R.A., Zhang, H., Jones, C., 2015. Intelligent autonomous system for residential water end use classification: Autoflow. Appl. Soft Comput. 31, 118–131.

Pacheco, C.J.B., Brewer, J.C., Horsburgh, J.S., Caraballo, J., 2021. An open source cyberinfrastructure for collecting, processing, storing and accessing high temporal resolution residential water use data. Environ. Model. Softw. 144, 105137.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Precioso, D., Gómez-Ullate, D., 2020. NILM as a regression versus classification problem: the importance of thresholding. arXiv preprint arXiv:2010.16050.

Schirmer, P.A., Mporas, I., 2022. Non-intrusive load monitoring: A review. IEEE Trans. Smart Grid.

Shwartz-Ziv, R., Armon, A., 2022. Tabular data: Deep learning is not all you need. Inf. Fusion 81, 84–90.

Stewart, R.A., Nguyen, K., Beal, C., Zhang, H., Sahin, O., Bertone, E., Vieira, A.S., Castelletti, A., Cominola, A., Giuliani, M., et al., 2018. Integrated intelligent water-energy metering systems and informatics: Visioning a digital multi-utility service provider. Environ. Model. Softw. 105, 94–117.

Taormina, R., Galelli, S., Tippenhauer, N.O., Salomons, E., Ostfeld, A., Eliades, D.G., Aghashahi, M., Sundararajan, R., Pourahmadi, M., Banks, M.K., et al., 2018. Battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks. J. Water Resour. Plann. Manag. 144 (8), 04018048.

The XGBoost Contributors, 2016. XGBoost. URL https://xgboost.ai/.

Tukey, J.W., 1977. Exploratory data analysis. Addison-Wesley Publishing Company.

Vitter, J.S., Webber, M.E., 2018. A non-intrusive approach for classifying residential water events using coincident electricity data. Environ. Model. Softw. 100, 302–313.

Zoha, A., Gluhak, A., Imran, M.A., Rajasegarar, S., 2012. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. Sensors 12 (12), 16838–16866.

Zounemat-Kermani, M., Matta, E., Cominola, A., Xia, X., Zhang, Q., Liang, Q., Hinkelmann, R., 2020. Neurocomputing in surface water hydrology and hydraulics: A review of two decades retrospective, current status and future prospects. J. Hydrol. 588, 125085.