water

MDPI

Article

AI-Driven Anomaly Detection in Smart Water Metering Systems Using Ensemble Learning

Maria Nelago Kanyama [1,*], Fungai Bhunu Shava [1], Attlee Munyaradzi Gamundani [1] and Andreas Hartmann [2]

[1] Department of Computer Science, Namibia University of Science and Technology (NUST), Private Bag 13388, Windhoek 9000, Namibia; agamundani@nust.na (A.M.G.)
[2] Institute of Groundwater Management, Technical University of Dresden, 01069 Dresden, Germany; andreas.hartmann@tu-dresden.de
* Correspondence: mkanyama91@gmail.com

Abstract

Water, the lifeblood of our planet, sustains ecosystems, economies, and communities. However, climate change and increasing hydrological variability have exacerbated global water scarcity, threatening livelihoods and economic stability. According to the United Nations, over 2 billion people currently live in water-stressed regions, a figure expected to rise significantly by 2030. To address this urgent challenge, this study proposes an AI-driven anomaly detection framework for smart water metering networks (SWMNs) using machine learning (ML) techniques and data resampling methods to enhance water conservation efforts. This research utilizes 6 years of monthly water consumption data from 1375 households from Location A, Windhoek, Namibia, and applies support vector machine (SVM), decision tree (DT), random forest (RF), and k-nearest neighbors (kNN) models within ensemble learning strategies. A significant challenge in real-world datasets is class imbalance, which can reduce model reliability in detecting abnormal patterns. To address this, we employed data resampling techniques including random undersampling (RUS), SMOTE, and SMOTEENN. Among these, SMOTEENN achieved the best overall performance for individual models, with the RF classifier reaching an accuracy of 99.5% and an AUC score of 0.998. Ensemble learning approaches also yielded strong results, with the stacking ensemble achieving 99.6% accuracy, followed by soft voting at 99.2% and hard voting at 98.1%. These results highlight the effectiveness of ensemble methods and advanced sampling techniques in improving anomaly detection under class-imbalanced conditions. To the best of our knowledge, this is the first study to explore and evaluate the combined use of ensemble learning and resampling techniques for ML-based anomaly detection in SWMNs. By integrating artificial intelligence into water systems, this work lays the foundation for scalable, secure, and efficient smart water management solutions, contributing to global efforts in sustainable water governance.

Keywords: anomaly detection; ensemble machine learning; class imbalance; smart water metering networks; SMOTE; SMOTEEN; water efficiency

check for updates

Academic Editor: Bithin Datta

Received: 19 April 2025
Revised: 16 May 2025
Accepted: 19 May 2025
Published: 27 June 2025

Citation: Kanyama, M.N.; Bhunu Shava, F.; Gamundani, A.M.; Hartmann, A. AI-Driven Anomaly Detection in Smart Water Metering Systems Using Ensemble Learning. *Water* **2025**, *17*, 1933. https://doi.org/10.3390/w17131933

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Water is a cornerstone of sustainable development, supporting numerous United Nations Sustainable Development Goals (SDGs), particularly SDG 6, which aims to ensure the availability and sustainable management of water and sanitation for all [1]. Central to achieving this goal are water metering networks, which play a critical role in the global

infrastructure by facilitating the efficient distribution of water—an essential resource for human survival, economic growth, and environmental sustainability [2]. However, like many other complex networked systems, water metering networks are prone to various anomalies, such as leaks, meter malfunctions, and data transmission errors, which can severely impact their reliability and efficiency [3,4]. These disruptions pose substantial challenges, especially in water-scarce regions where precise monitoring and management of water resources are vital.

In response to these challenges, SWMNs have emerged, harnessing the power of the Internet of Things (IoT) to enable real-time monitoring of water usage and distribution [2]. The deployment of SWMNs not only addresses the immediate need for improved water management but also aligns with broader SDG objectives, contributing to environmental sustainability, economic development, and social equity [5]. SWMNs offer significant advantages over traditional water metering systems by enabling real-time monitoring, optimizing resource allocation, and improving water conservation efforts [6]. Unlike conventional meters, which require manual readings and are prone to delays in detecting anomalies, SWMNs facilitate automated data collection, anomaly detection, and predictive analytics, thus reducing operational inefficiencies and water losses [7,8]. Refs. [9,10] contributed to the advancement of SWMNs by simulating and developing a model that raised awareness of their adoption among water utilities. Their work emphasized the need for strategic implementation to enhance decision-making, infrastructure planning, and consumer engagement. Further research by [9] extended this by investigating optimal data acquisition point placements within SWMNs. Their findings underscored that strategic sensor placements are crucial for maximizing network efficiency, ensuring accurate data collection, and minimizing signal loss or redundancy.

Despite these advancements, a significant gap remains in the ability of current systems to automatically detect and mitigate anomalies in real time, and where the anomaly detection is needed, there exists a severe class imbalance in the training datasets. Class imbalance is when a dataset's class distribution of the classes present is not equal and skewed [11]. A class may ultimately have many instances of the total sample called the majority class, and the class with fewer instances is called the minority class. For example, in the water consumption dataset anomaly detection problem, where leakages are just 2% of the total dataset, this is referred to a class imbalance of the order of 100:2. In most cases in anomaly detection, the minority class is more important than the majority class under certain contexts [11]. The application of artificial intelligence (AI) in this domain, specifically for anomaly detection, has made limited progress, primarily due to the scarcity of high-quality, labeled data and the complexity of identifying irregular patterns in water consumption [4]. This paper argues that integrating AI, particularly ML techniques, into SWMNs can significantly enhance the reliability and responsiveness of these systems by enabling real-time anomaly detection and adaptive responses. By leveraging ensemble learning approaches, which combine the predictive strengths of multiple ML models, this research aims to improve anomaly detection capabilities, resulting in more accurate, efficient, and robust monitoring of water distribution networks.

The primary aim of this study is to develop an AI-driven anomaly detection model for SWMNs by leveraging ML techniques and data resampling techniques. Specifically, this paper investigates whether ensemble ML models offer significant improvements over individual ML models in detecting anomalies using various data resampling techniques. The objectives of the study are as follows: (1) To develop a framework for anomaly detection in water consumption data, integrating ML techniques and data resampling techniques to enhance monitoring and decision-making in SWMNs. (2) To address class imbalance in water consumption data by evaluating the effectiveness of various data balancing

techniques, including RUS, SMOTE, and SMOTEENN. (3) To analyze the performance of individual ML models—SVM, RF, DT, and kNN—in detecting anomalies in water consumption data under various data balancing techniques. (4) To develop and evaluate an ensemble ML model using stacking and voting techniques, comparing their performances against individual ML models using key evaluation metrics such as accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (ROC-AUC).

This research reports that AI-driven anomaly detection, particularly using ensemble ML techniques, has the potential to revolutionize the water sector by significantly improving water efficiency. These advancements contribute to sustainable water resource management by enhancing the operational efficiency and resilience of SWMNs, especially in regions facing environmental and resource-related challenges. The rest of this paper is structured as follows: *Section 2* presents the materials and methods, detailing the study area, data collection methods, proposed framework, class imbalance techniques, ML models, performance metrics, and experimental procedures used in the study. *Section 3* discusses the results and findings, providing an in-depth analysis of the performance of the ML models, their comparative evaluation, and validation against existing studies. *Section 4* concludes the study by summarizing key findings and outlining future research directions.

2. Materials and Methods

The study area and data collection process are first detailed, outlining the source, characteristics, and relevance of the dataset. The data acquisition techniques used in this study are then presented, followed by the proposed framework for developing the ML-based anomaly detection model. This framework defines the key parameters and model development phases, ensuring a systematic approach to detecting anomalies in water consumption data. Following this, the study introduces the ML models and data resampling techniques employed to address class imbalance. The impact of these techniques on model performance is systematically analyzed. Additionally, the performance metrics and experimental procedures used to evaluate the models are outlined. Key evaluation measures include accuracy, precision, recall, F1-score, confusion matrix, and AUC-ROC, which provide insights into the models' effectiveness in distinguishing anomalies from normal water consumption patterns. The study follows a structured experimental process to ensure reproducibility and robust validation of the results.

2.1. Study Area

An ensemble ML framework was proposed to detect anomalies across various water meters rather than creating individual models for each household. The study was conducted in Location A, situated in Windhoek, Namibia, with data provided by the City of Windhoek, which manages water metering for billing and operational purposes. This location was chosen due to its unique characteristics, making it an ideal setting for advancing research on SWMNs. One notable feature is the presence of an independent water reservoir, providing a controlled environment to study anomalies and develop robust anomaly detection models. The location's infrastructure, which relies on mechanical water meters, plays a crucial role in bridging the gap between traditional water metering systems and the rapidly evolving landscape of SWMNs. By utilizing historical data from these mechanical meters, the study offers valuable insights into water consumption patterns and behaviors over time. The combination of mechanical meter readings and the independent water reservoir creates a reliable foundation for analyzing real-world data. This setting not only aids in the development of effective anomaly detection models but also opens opportunities to explore how legacy water systems can transition into more advanced

SWMNs. These advancements are critical for enhancing the operational efficiency and reliability of water resource management in regions like Namibia.

2.2. Data Collection Methods

The dataset used in this study comprises raw water consumption records collected over a 6-year period, from January 2017 to December 2022, covering 1375 residential households in Location A, Windhoek, Namibia. Each record contains key attributes, including a unique meter identification number, the reading date, and the total monthly water consumption in cubic meters.

The acquired dataset is presented in Tables 1–6, which provide the descriptions of statistics used in this research. The statistical metrics include count, mean, standard deviation (std), minimum (min), 25th percentile (25%), median (50%), 75th percentile (75%), and maximum (max). A close analysis of the tables reveals a steady increase in the number of active water meters, starting with 808 in 2017 and progressively rising to 1303 by 2022. This growth signifies the ongoing development in Location A, accompanied by an escalating water demand as indicated by the upward trend in the mean water consumption values over the years. Interestingly, the period from January 2017 to January 2020 records the lowest mean water consumption. This decline may be attributed to seasonal factors such as holidays when consumers were away or lapses in water meter readings for those months. In contrast, the years 2021 and 2022 show a marked improvement in water consumption patterns, accompanied by more consistent and reliable data recording practices. These observations highlight not only the evolving water demand in the region but also the enhancements in data collection and management over time.

Given the limitations of monthly data for capturing short-term anomalies, the dataset was temporally augmented to daily resolution using the piecewise cubic Hermite interpolating polynomial (PCHIP) method [12]. The PCHIP was selected over standard spline interpolation because it preserves monotonicity and prevents overshooting between data points, which is essential when modeling physical processes such as water consumption that follow gradual, non-oscillatory patterns. This approach generated a synthetic but realistic daily consumption profile that allowed for the detection of finer-grained anomalies. To facilitate accurate labeling, a long short-term memory autoencoder (LSTM-AE) was employed and further explained in [13]. The LSTM-AE reconstructed normal consumption sequences and flagged deviations based on reconstruction error, allowing for the identification of anomalies such as spikes, prolonged low usage, and sudden changes. This labeling was critical for supervised training and evaluation of the ensemble ML models. The combination of temporal augmentation and neural-based labeling provided a robust foundation for developing a model suited to operational environments where high-frequency data are often unavailable.

Table 1. Description of statistics of water consumption data for 2017.

	JAN_17	FEB_17	MAR_17	APR_17	MAY_17	JUN_17	JUL_17	AUG_17	SEP_17	OCT_17	NOV_17	DEC_17
1	808	808	808	808	808	808	808	808	808	808	808	808
2	8.884901	6.440594	6.460396	6.69802	11.00124	7.90099	6.87995	11.63243	11.53713	12.75371	15.03094	23.64604
3	16.2794	15.14732	10.27555	15.47818	17.16813	12.9908	13.5372	19.92893	18.20954	23.09042	22.40475	115.3365
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	1	2	4	5
6	4	3	4	4	7	6	5	8	8	9	11	12
7	13	9	10	10	16	11	9	15	15.25	17	19	21.25
8	270	311	191	373	272	208	307	333	281	381	310	3110

Table 2. Description of statistics of water consumption data for 2018.

	JAN_18	FEB_18	MAR_18	APR_18	MAY_18	JUN_18	JUL_18	AUG_18	SEP_18	OCT_18	NOV_18	DEC_18
1	1028	1028	1028	1028	1028	1028	1028	1028	1028	1028	1028	1028
2	0.227626	20.65661	11.32977	10.93288	8.238327	11.1216	11.29767	8.965953	13.66926	11.54086	14.63327	10.3142
3	2.266288	39.7629	18.86923	17.55782	11.17806	17.50793	16.64495	11.02629	19.78422	14.49955	17.91407	14.04124
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	2	2	3	3	5	3
6	0	12.5	7	7	6	9	9	7	11	9	11	7
7	0	27	15	14	12	15	15	13	18	15	20	14
8	51	677	312	298	224	342	314	233	366	209	333	199

Table 3. Description of statistics of water consumption data for 2019.

	JAN_19	FEB_19	MAR_19	APR_19	MAY_19	JUN_19	JUL_19	AUG_19	SEP_19	OCT_19	NOV_19	DEC_19
1	1126	1126	1126	1126	1126	1126	1126	1126	1126	1126	1126	1126
2	0.045293	26.50622	11.02575	11.67584	10.18206	9.974245	14.05595	10.18739	13.2167	14.33481	10.73535	12.47336
3	0.709736	33.67401	13.50688	25.35285	12.04723	10.70087	18.20631	12.76431	16.20351	18.03091	11.60181	18.74732
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	6	2	3	3	3	5	4	6	6	4	5
6	0	19	8	8	8	8	12	8	11	12	8	10
7	0	37	16	15	14	14	19	14	18	19	15	16
8	15	633	261	712	247	220	336	220	346	353	214	424

Table 4. Description of statistics of water consumption data for 2020.

	JAN_20	FEB_20	MAR_20	APR_20	MAY_20	JUN_20	JUL_20	AUG_20	SEP_20	OCT_20	NOV_20	DEC_20
1	1224	1224	1224	1224	1224	1224	1224	1224	1224	1224	1224	1224
2	9.714052	11.19608	11.44853	0.896242	24.17484	9.834967	10.00408	10.78023	20.45425	12.8317	16.75163	11.61356
3	12.59281	16.81064	17.42454	5.257695	41.42892	16.96552	11.42815	13.40226	22.42771	22.4398	19.72643	15.57206
4	0	0	0	0	0	0	0	0	0	0	0	0
5	3	3	4	0	9	3	4	4	10	5	6	4
6	8	9	9	0	19	8	8	9.5	18	10	14	9
7	13	15	14	0	31	13	14	14	28	17	23	15
8	284	334	340	114	773	417	246	298	540	605	390	255

Table 5. Description of statistics of water consumption data for 2021.

	JAN_21	FEB_21	MAR_21	APR_21	MAY_21	JUN_21	JUL_21	AUG_21	SEP_21	OCT_21	NOV_21	DEC_21
1	1268	1268	1268	1268	1268	1268	1268	1268	1268	1268	1268	1268
2	12.85489	9.337539	13.85252	14.73344	13.72319	10.9653	13.21767	14.18297	15.74842	13.79968	20.31467	10.55205
3	22.42158	13.37025	18.09838	20.60463	64.03196	15.23938	18.27562	21.99177	18.26471	16.85729	24.59939	14.96504
4	0	0	0	0	0	0	0	0	0	0	0	0
5	4	3	5	5	4	4	5	5	7	5	8.75	4
6	9	8	11	12	8	9	11	12	13	11	16	8
7	16	13	18	19	13	14	17	19	21	18	26	13
8	532	311	339	426	1599	309	328	516	385	302	465	300

Table 6. Description of statistics of water consumption data for 2022.

	JAN_22	FEB_22	MAR_22	APR_22	MAY_22	JUN_22	JUL_22	AUG_22	SEP_22	OCT_22	NOV_22	DEC_22
1	1303	1303	1303	1303	1303	1303	1303	1303	1303	1303	1303	1303
2	41.32694	38.09286	23.51343	26.6769	28.39908	22.10207	26.86416	24.38219	33.0284	28.8089	32.45741	26.82348
3	731.3927	734.125	416.9982	471.3119	502.6794	390.7141	473.3532	430.6239	584.7378	508.1026	573.1599	473.7025
4	0	0	0	0	0	0	0	0	0	0	0	0
5	7	3	4	5	5	4	5	5	7	5	7	5
6	17	9	10	11	12	9	12	10	13	12	14	11
7	29	15	16	19	19	15	19	17	21	20	21	18
8	26,406	24,545	15,053	17,013	18,144	14,092	17,089	15,544	21,110	18,342	20,693	17,100

NB: The monthly water consumption dataset was augmented to daily resolution using the PCHIP method, as detailed in our earlier work [13]. The resulting daily time series

was then labeled using LSTM-AE, which reconstructed normal patterns and identified deviations based on reconstruction error, following the same procedure established in [13].

2.3. Proposed Framework

Based on the literature review and research findings, we propose the framework illustrated in Figure 1, which outlines the development phases of the model adopted in this study. The framework incorporates a structured approach to handling class imbalance, model training, and evaluation, ensuring robust anomaly detection in SWMNs. The following steps highlight the proposed framework:

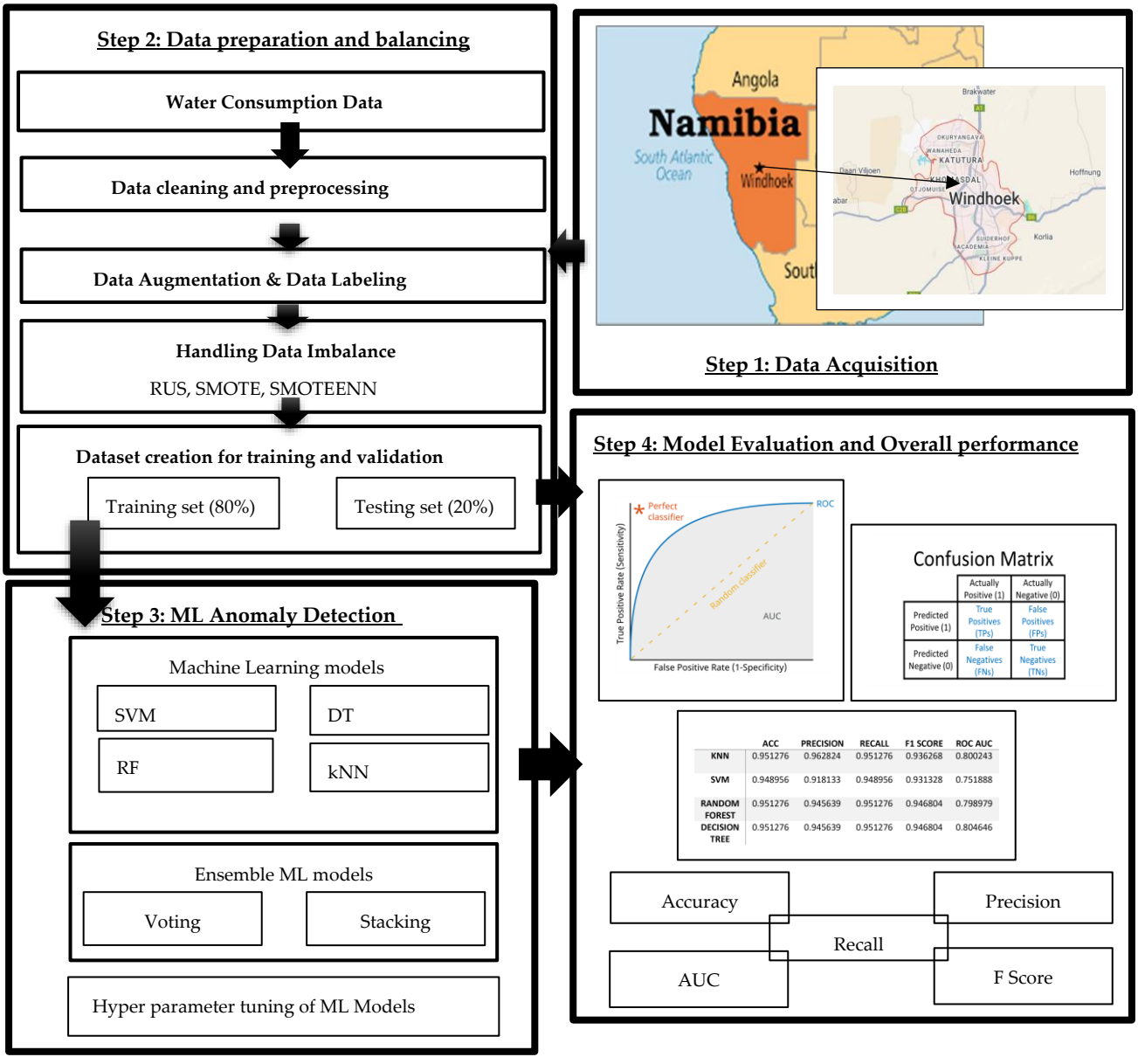


Figure 1. Proposed framework.

2.3.1. Step 1: Data Acquisition

Historical water consumption data were obtained from the City of Windhoek. The records span a 6-year period and cover a diverse range of urban households.

2.3.2. Step 2: Data Preparation and Balancing

The raw dataset consists of monthly water consumption readings collected over a 6-year period (2017–2022) from 1375 households in Windhoek, Namibia. Each record includes the following attributes:

1. Meter ID (unique identifier for each household water meter)
2. Reading Date (timestamp of the monthly reading)
3. Monthly Consumption (in cubic meters)
4. Derived Features: interpolated daily consumption, and anomaly labels

Additional spatial variables such as household location and building type were not available due to privacy constraints. However, the water meters are distributed across several residential zones, capturing a heterogeneous mix of consumption patterns representative of the city's broader water usage behavior. The dataset underwent thorough cleaning, which involved removing duplicate records, addressing missing values, and validating zero water consumption entries. Households with more than 80 percent missing data across the 6-year period were excluded from the analysis. Cases with zero consumption were confirmed through ground-truth validation in collaboration with municipal technicians, who identified meter disconnections, service interruptions, or periods of non-occupancy.

Given the limitation of monthly readings for detecting short-term consumption anomalies, the dataset was temporally augmented to daily resolution using PCHIP interpolation. A LSTM-AE was trained on the daily sequences to reconstruct normal consumption behavior. Observations with high reconstruction error were flagged as anomalous, allowing for more reliable labeling of normal and abnormal cases. Because real-world anomaly instances were limited in the dataset, synthetic anomalies were also generated using the interquartile range method. This statistical approach was well suited to skewed distributions and allows for the simulation of meaningful deviations from typical usage patterns. Two anomaly types were modeled:

- *Spikes*: Single-day values above Q3 + 1.5 × IQR
- *Sudden changes*: Rapid deviations from previous baseline values

NB: Anomaly Class Definitions

➢ *Class 0—Normal Consumption*

This class includes regular water usage patterns that follow expected trends with minimal deviations. These observations represent the baseline behavior, often governed by routine household consumption with no signs of leakage, abnormal spikes, or operational disruptions.

➢ *Class 1—Spike*

Spike anomalies are characterized by sudden, sharp increases in water consumption over a short duration, typically a single time step. These events may be triggered by pipe bursts, unattended outdoor water use, or temporary infrastructure malfunctions. They are transient but potentially costly if not promptly addressed [4].

➢ *Class 2—Sudden Changes*

This class captures abrupt and prolonged deviations in consumption behavior. These changes may reflect structural leakage, shifts in household occupancy, water rationing responses, or behavioral changes such as increased irrigation or conservation practices. Unlike spikes, sudden changes persist over time and often indicate deeper system issues [4].

These two types were chosen as the focus of this study, as they are among the most prevalent in urban water metering systems, as established in our earlier work [4]. Following preprocessing and anomaly labeling, the dataset revealed a strong class imbalance, with

normal observations significantly outnumbering anomalous ones. To address this, three resampling techniques were applied: (1) RUS (2) SMOTE (3) SMOTEENN.

Data Imbalance Handling

After data labeling using the LSTM-AE reconstruction error approach, the resulting dataset exhibited significant class imbalance, which posed challenges for effectively training anomaly detection models. Out of the total number of labeled data points, approximately 92% were classified as "normal", while only 3% were labeled as "spike" and 5% were labeled as "sudden change". This skewed distribution revealed a dominance of normal instances, with the minority anomaly classes significantly underrepresented, potentially leading to biased learning and poor generalization in identifying critical irregularities. To address this, the research employed three class imbalance handling techniques: RUS; SMOTE; and SMOTEENN, a hybrid method combining SMOTE and edited nearest neighbors. These techniques were critical in rebalancing the dataset and ensuring that minority classes were adequately represented during training. The pseudocodes are detailed below (Algorithms 1–4):

Algorithm 1: Dataset Input

Input Dataset:
X ← Feature matrix (interpolated daily water consumption (PCHIP))
y ← Labels generated by LSTM-AE reconstruction error
where:
Class 0 = Normal Consumption
Class 1 = Spike Anomaly
Class 2 = Sudden Change

Algorithm 2: RUS

Input:
X ← feature matric of shape (n_samples, n_features)
y ← corresponding label vector (normal = 0, anomaly = 1 or 2)
Output:
X_resampled, y_resampled←Balanced dataset with equal class representation
Procedure:
1. Identify majority and minority class indices in y.
2. Extract:
X_majority, y_majority←samples where y==majority class
y_minority, y_minority←samples where y==minority class
3. Randomly under-sample X_majority to match the size of X_minority.
4. Concatenate:
X_resampled ← [X_minority; X_sampled_majority]
y_resampled ← [y_minority; y_sampled_majority]
5. Shuffle X_resampled and y_resampled to ensure randomization.
6. Return X_resampled, y_resampled

Algorithm 3: SMOTE

Input:

X ← Feature matrix of shape (n_samples, n_features)

y ← Corresponding label vector (normal = 0, anomaly = 1 or 2)

k ← Number of nearest neighbors for interpolation (default: 5)

Output:

X_augmented, y_augmented ← Dataset including original and synthetic minority samples

Procedure:

1. Identify minority class samples in X and y.
2. For each minority instance xi:
   a. Identify k nearest neighbors from within the minority class.
   b. Randomly select one or more neighbors.
   c. For each selected neighbor xj:
      i. Generate synthetic sample xs:
         $xs = xi + rand(0,1) \times (xj - xi)$
      ii. Append xs to X_synthetic and assign corresponding label to y_synthetic.
3. Concatenate:
   X_augmented ← [X; X_synthetic]
   y_augmented ← [y; y_synthetic]
4. Return X_augmented, y_augmented

Algorithm 4: SMOTEENN

Input:

X ← Feature matrix of shape (n_samples, n_features)

y ← Corresponding label vector (normal = 0, anomaly = 1 or 2)

k ← Number of neighbors for both SMOTE and ENN stages (5)

Output:

X_cleaned, y_cleaned ← Balanced and noise-filtered dataset

Procedure:

1. Apply SMOTE to X and y:
   a. Follow the SMOTE procedure to obtain X_smote and y_smote.
2. Apply ENN to (X_smote, y_smote):
   For each instance xi in X_smote:
      a. Identify k nearest neighbors.
      b. If label of xi differs from majority label among its neighbors:
         Remove xi from X_smote and y_smote.
3. The remaining instances form:
   X_cleaned ← filtered feature matrix
   y_cleaned ← corresponding filtered labels
4. Return X_cleaned, y_cleaned

Each resampled dataset was split into training (80%) and testing (20%) subsets using stratified sampling to preserve class distribution. ML classification was then performed with ensemble strategies such as voting and stacking implemented to improve generalization and model robustness.

NB: All individual ML models (SVM, kNN, DT, RF) were trained and evaluated on datasets resampled using three techniques: RUS, SMOTE, and SMOTEENN. However, ensemble learning models (stacking and voting) were developed using the SMOTEENN-resampled dataset, which yielded the best performance in handling class imbalance.

2.3.3. Step 3: ML Algorithms for Anomaly Detection

Following the data augmentation, anomaly labeling, and data imbalance handling phase, the next stage involved developing ML models to classify normal and anomalous water consumption patterns. Anomaly detection in this context is framed as a supervised learning problem, where labeled consumption data are used to train and evaluate predictive models [14]. In this study, four ML models were selected based on their proven effectiveness in anomaly detection and their adaptability to SWMNs. These models were chosen based on their diverse architectures, varying data requirements, and potential for strong performance, as indicated in prior research [3]. The selection of these models represents a strategic blend of traditional approaches that have been extensively tested in related anomaly detection studies [1,15–23]. These models were also chosen to allow direct comparisons with previous studies, further validating the results of this research. By utilizing these well-established ML models, this study aims to not only benchmark the performance of anomaly detection in SWMNs but also explore how these methods can be adapted and improved for real-time water resource management. The following pseudocodes summarize the working steps for each algorithm (Algorithms 5–10):

Algorithm 5: SVM

Input:
X, y ← Feature matrix and LSTM-labeled target
Resample method ← RUS, SMOTE, or SMOTEENN
Kernel ← RBF
C ← Regularization parameter
Output:
ŷ ← Predicted class labels for X_test
Anomaly_Index ← Distance to decision boundary
Procedure:
1. Apply selected resampling method to (X, y) → (X_bal, y_bal)
2. Standardize X_bal
3. Train SVM using kernel and C on (X_bal, y_bal)
4. Predict class labels ŷ on X_test
5. Compute Anomaly_Index = |decision_function(X_test)|
6. Return ŷ, Anomaly_Index

Algorithm 6: KNN

Input:
X, y ← LSTM-labeled dataset
Resample method ← RUS, SMOTE, or SMOTEENN
k ← Number of neighbors
Output:
$\hat{y}$ ← Predicted class labels
Anomaly_Index ← Mean distance to k neighbors
Procedure:
1. Apply resampling to (X, y) → (X_bal, y_bal)
2. Normalize X_bal and X_test
3. For each sample x in X_test:
a. Compute distance to all points in X_bal
b. Identify k-nearest neighbors
c. Predict $\hat{y}$ by majority voting
d. Anomaly_Index ← Mean distance to neighbors
4. Return $\hat{y}$, Anomaly_Index

Algorithm 7: Random Forest

Input:
X, y ← LSTM-labeled dataset
Resample method ← RUS, SMOTE, or SMOTEENN
n_trees ← Number of decision trees
Output:
$\hat{y}$ ← Class predictions
Anomaly_Index ← Voting confidence or entropy
Procedure:
1. Resample (X, y) → (X_bal, y_bal)
2. Train RF on X_bal with n_trees, using feature sub-sampling
3. For each x in X_test:
a. Predict class from each tree
b. $\hat{y}$ ← Majority vote
c. Anomaly_Index ← 1 − max(vote_distribution)
4. Return $\hat{y}$, Anomaly_Index

Algorithm 8: Decision Tree

Input:
X, y ← LSTM-labeled dataset
Resample method ← RUS, SMOTE, or SMOTEENN
Criterion ← Entropy
Output:
$\hat{y}$ ← Class predictions
Anomaly_Index ← Leaf node purity or depth
Procedure:
1. Resample (X, y) → (X_bal, y_bal)
2. Train DT classifier using specified criterion

Algorithm 8: *Cont.*

3. For each x in X_test:
    a. Traverse tree to a leaf node
    b. Assign ŷ based on majority class in leaf
    c. Anomaly_Index ← Depth of leaf/impurity measure
4. Return ŷ, Anomaly_Index

Algorithm 9: Stacking Ensemble

Input:
    X_train ← Resampled dataset using SMOTEENN
    y_train ← Class labels
    base_models ← {SVM, k-NN, DT, RF}
    meta_model ← Logistic Regression
Output:
    stacking_model ← Trained ensemble
    ŷ← Predicted class labels
Procedure:
1. Train each base_modeli on (X_train, y_train).
2. For each x ∈ X_train:
        a. Predict base outputs pi = base_modeli(x)
3. Form meta-feature vector X_meta ← [p1, p2, p3, p4]
4. Train meta_model on (X_meta, y_train)
5. For each x ∈ X_test:
        a. Get base predictions [p1, . . ., p4]
        b. Predict final label ŷ ← meta_model([p1, . . ., p4])
6. Return ŷ

Algorithm 10: Voting Ensemble

Input:
    X_train ← Resampled dataset using SMOTEENN
    y_train ← Class labels
    base_models ← {SVM, k-NN, DT, RF}
    voting_type ← {'hard' or 'soft'}
Output:
    ŷ← Final prediction
    anomaly_index ← Voting confidence
Procedure:
1. Train each base_modeli on (X_train, y_train)
2. For each x ∈ X_test:
        a. Collect predictions pi from all base models
        b. If voting_type == 'hard':
            ŷ← class with majority votes
            anomaly_index ← ratio of minority votes
        c. If voting_type == 'soft':
            ŷ ← class with highest average predicted probability
            anomaly_index ← standard deviation of probabilities
3. Return ŷ and anomaly_index

2.3.4. Step 4: Key Performance Metrics

As highlighted in [3], relying on a single performance metric is not sufficient for evaluating the effectiveness of ML models in anomaly detection. To obtain a more comprehensive view of model performance, this study employs multiple metrics, including accuracy, precision, recall, F1 score, the confusion matrix, and ROC-AUC [24,25]. These metrics provide a holistic understanding of how well the models detect anomalous water consumption patterns in SWMNs.

The true positive rate (TPR), or sensitivity, measures the model's ability to correctly identify instances of anomalous water consumption, while the false positive rate (FPR) evaluates the model's incorrect classification of normal water consumption as anomalous. The ROC-AUC score [26], which plots the TPR against the FPR, was also employed as a key metric to assess the trade-off between detecting water anomalies and avoiding false alarms. These metrics are critical for assessing the model's reliability, as they directly align with the research objective of enhancing anomaly detection. To assess the impact of data balancing techniques, the performance of individual models under various class imbalance handling techniques was analyzed.

The confusion matrix was also employed (see Table 7); this is a table that visualizes the performance of a classification model by comparing the predicted labels with the actual labels [27]. It provides insights into the types of errors made by the model. For binary classification in anomaly detection, the confusion matrix consists of four components defined as follows:

Table 7. Confusion matrix.

	Predicted Anomalies	**Predicted Normal**
Actual anomalies	TP	FN
Actual normal	FP	TN

True Positive (TP): Anomalous water consumption correctly identified as anomalous.
True Negative (TN): Normal water consumption correctly identified as normal.
False Positive (FP): Normal water consumption incorrectly identified as anomalous.
False Negative (FN): Anomalous water consumption incorrectly identified as normal.
The corresponding performance formulas are as follows:

Accuracy: It measures the percentage of correct predictions made by the model out of the total number of samples in the dataset [28].

$$Accuracy = (TP + TN)/(TP + TN + FN + FP) \quad (1)$$

Recall: It evaluates the ability of a model to correctly identify positive samples from the total number of samples [28].

$$Recall = TP/(TP + FN) \quad (2)$$

Precision: It evaluates the ability of a model to correctly identify positive samples from the total number of samples

$$Precision = FP/(TN + FP) \quad (3)$$

F1 Score: It is used to assess the balance between precision and recall [29,30].

$$F1\ Score = (Precision \times Recall)/(Precision + Recall) \quad (4)$$

AUC (area under the curve) was used to measure the performance of a binary classification model by evaluating its ability to distinguish between positive (anomalous) and negative (normal) instances across varying decision thresholds [26]. The ROC curve plots the true positive rate (sensitivity) against the false positive rate, capturing the trade-offs between correctly identifying anomalies and falsely flagging normal instances. A higher AUC-ROC value indicates better overall discrimination capability, meaning the model more effectively separates anomalous from normal cases. This metric is especially valuable for anomaly detection, where class imbalance is common, as it provides a holistic view of the model's classification performance across all thresholds, independent of the decision boundary [25,26].

By employing these metrics, this study aims to demonstrate how the proposed ensemble model enhances anomaly detection accuracy, contributing to more effective and reliable water resource management.

2.4. Experimental Setup

We have performed all our experiments on the 13th Gen Intel(R) Core (TM) i7-1355U 1.70 GHz with 16 GB RAM and 64-bit Windows OS. The Anaconda Navigator (64 bit) version with Jupyter Notebook 7.0.6 and Python version 3.12.1 ML libraries, such as Sklearn, matplotlib, NumPy, TensorFlow, Sklearn, and Keras, formed our experimental test bed.

3. Results and Discussion

This section presents the outcomes of the proposed anomaly detection framework and explains how each stage contributed to the results. The findings are presented in relation to Tables 1–6 and Figure 1 to ensure transparency and reproducibility. The dataset described in Section 2.2 was interpolated, labeled using LSTM-AE, and then subjected to different resampling and classification strategies to assess the overall effectiveness of the framework. Each phase in this pipeline is directly linked to measurable outputs that validate its contribution to model performance.

(a) Effects of Class Imbalance in Classification

One of the key challenges in anomaly detection tasks is the presence of class imbalance, which can significantly affect the reliability of ML models. The dataset used in this study was heavily skewed toward normal consumption readings, with only a small fraction representing spike and sudden change anomalies. This imbalance, if left unaddressed, can result in models that are biased toward the majority class, leading to poor detection of rare but critical events [31]. To address the inherent class imbalance in the dataset, three resampling techniques—RUS, SMOTE, and SMOTEENN—were employed, and their effectiveness was evaluated using multiple performance metrics. In anomaly detection tasks, relying on a single metric such as accuracy often masks the true performance of resampling techniques, especially in imbalanced datasets [32]. Thus, a combination of metrics, including accuracy, recall, precision, F1-score, and AUC-ROC, was utilized to provide a holistic evaluation. These metrics help reveal how well each technique improves the models' ability to identify rare anomalies without sacrificing performance on the majority class.

Figure 2 illustrates the recall performance of four ML models—DT, kNN, RF, and SVM—under the three resampling techniques. Among the methods, RUS consistently produced the lowest recall scores, with DT achieving 0.6 and SVM slightly better at 0.7. This highlights a key limitation of RUS: the reduction of majority class data significantly impairs the models' ability to identify anomalies effectively. This agrees with the work in [33] in which the average recall metric for RUS was lower than for SMOTE and SMOTEENN. In

contrast, SMOTE showed considerable improvement across all models, with recall scores for kNN and RF reaching 0.9, while SVM achieved 0.8. This improvement underscores the value of synthetic data generation in enhancing the representation of minority class samples and improving anomaly detection.

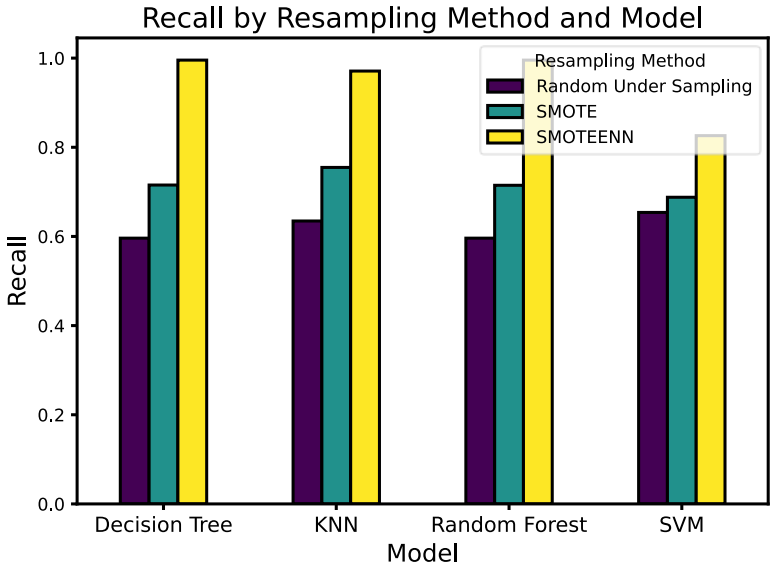


Figure 2. Recall performance.

SMOTEENN emerged as the superior resampling method, achieving perfect recall (1.0) for DT and kNN, and nearly perfect recall values of 0.98 and 0.97 for RF and SVM, respectively. These findings align with those of [34], who reported a recall of 1.00 for RF and kNN and 0.917 for SVM. SMOTEENN combines SMOTE's synthetic data generation with edited nearest neighbors (ENN) to remove noisy and borderline instances from the enriched dataset, which leads to a more defined and balanced dataset [32]. This hybrid technique combines the strengths of SMOTE in generating synthetic minority class samples with edited nearest neighbors (ENN) to remove noisy data points, resulting in a highly refined and well-balanced dataset. The results highlight the critical role of addressing both class imbalance and noise in the data to enhance model performance. By leveraging SMOTEENN, the models demonstrated remarkable improvements in anomaly detection, making it the most effective resampling technique for this study, particularly in SWMNs where imbalanced datasets are prevalent.

Figure 3 illustrates the precision performance of classifiers—DT, kNN, RF, and SVM—when applied to datasets processed with RUS, SMOTE, and SMOTEENN resampling techniques. The results highlight how these resampling strategies influence the classifiers' ability to correctly identify true positive anomalies while minimizing false positives. RUS consistently exhibited the lowest precision across all classifiers, with DT and RF achieving only 0.5 and 0.6, respectively. This underperformance indicates that RUS, while balancing the dataset, sacrifices crucial data points from the majority class, leading to an increased rate of false positives.

In contrast, SMOTE demonstrated a substantial improvement in precision. For example, kNN and SVM achieved precision values of 0.75 and 0.8, respectively, showcasing the effectiveness of generating synthetic minority class samples in improving anomaly identification. However, the most notable improvement was observed with SMOTEENN, which consistently delivered the highest precision across all classifiers, with DT and kNN reaching perfect precision (1.0), a result also reported by [34], further validating the robustness of SMOTEENN in handling class imbalance and improving classification accuracy. RF

and SVM also achieved high precision values of 0.9 and 0.95, respectively. This superior performance highlights SMOTEENN's capability to refine the dataset by not only enhancing the minority class representation but also removing noisy data points, ensuring a more accurate and reliable anomaly detection process.

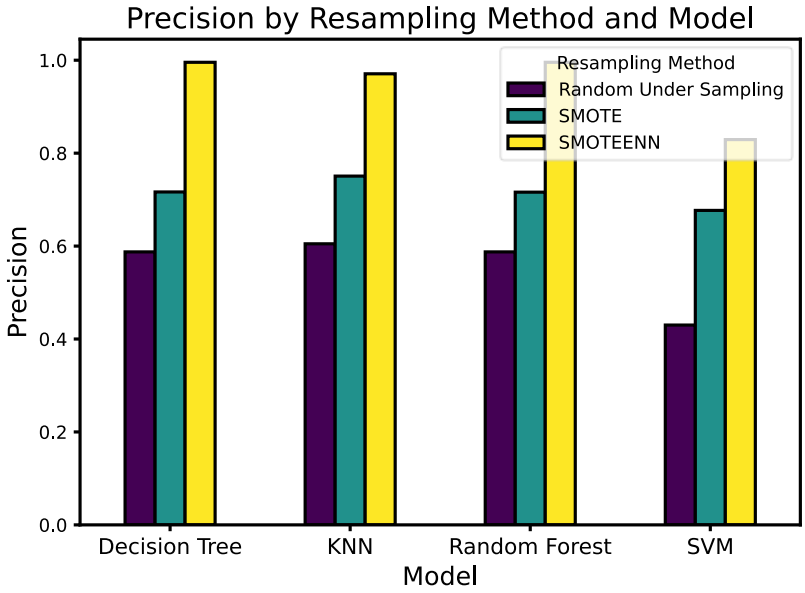


Figure 3. Precision performance.

In Figure 4, RUS shows consistently lower F1 scores, with DT and SVM recording values of 0.5 and 0.6, respectively, indicating its limitations in balancing the dataset while preserving essential information. The loss of critical samples due to RUS compromises its ability to effectively identify anomalies.

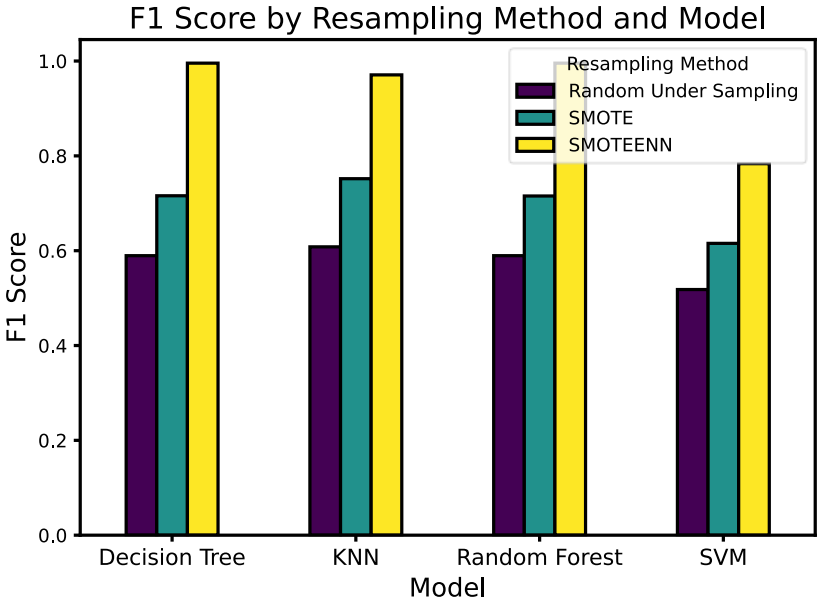


Figure 4. F1 score performance.

SMOTE introduces significant improvements, with kNN and RF achieving F1 scores of 0.8 and 0.85, respectively. However, the most pronounced improvements are observed with SMOTEENN. DT and kNN achieve perfect F1 scores (1.0), while RF and SVM closely follow with scores of 0.9 and 0.95, respectively. These findings are consistent with those reported

in [31,34–36]. The enhanced performance of SMOTEENN demonstrates its efficacy in addressing both class imbalance and data noise, ensuring robust anomaly detection. These results reinforce the superiority of SMOTEENN as the optimal technique for improving model performance in imbalanced datasets.

Figure 5 shows the AUC score for four ML models—DT, kNN, RF, and SVM—evaluated across three resampling techniques: RUS, SMOTE, and SMOTEENN. The AUC score, which measures the ability of models to distinguish between classes, is a critical metric for evaluating performance, particularly in imbalanced datasets.

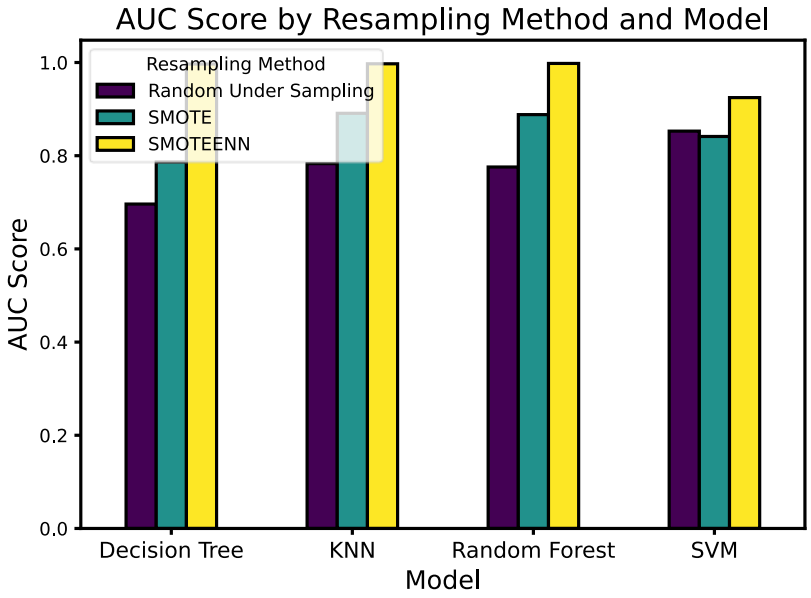


Figure 5. AUC score performance.

RUS demonstrates the lowest AUC scores across all models, with DT scoring approximately 0.7 and SVM achieving 0.8. These results highlight the method's limitations in preserving discriminatory power when reducing the dataset's majority class. SMOTE significantly improves AUC scores, with kNN and RF recording values of 0.9 and 0.95, respectively, underscoring the benefit of synthetic data generation for enhancing model performance.

SMOTEENN emerges as the most effective resampling technique, yielding near-perfect AUC scores for all models. DT and kNN achieve an AUC of 1.0, while RF and SVM closely follow with AUC scores of 0.98 and 0.97, respectively. These results demonstrate SMOTEENN's capability to not only balance the dataset but also refine the data quality, resulting in superior anomaly detection performance across all evaluated models.

Figure 6 highlights the accuracy performance of four ML models—DT, kNN, RF, and SVM—across three resampling techniques: random undersampling (RUS), SMOTE, and SMOTEENN. Accuracy, which measures the proportion of correctly classified instances, is an essential performance metric, particularly for datasets with imbalanced classes.

RUS consistently yields the lowest accuracy values across all models, with DT and kNN scoring approximately 0.65, and RF and SVM marginally better at 0.7. This decline in accuracy reflects the data loss inherent in RUS, where a significant portion of the majority class is removed, compromising the classifier's ability to generalize effectively.

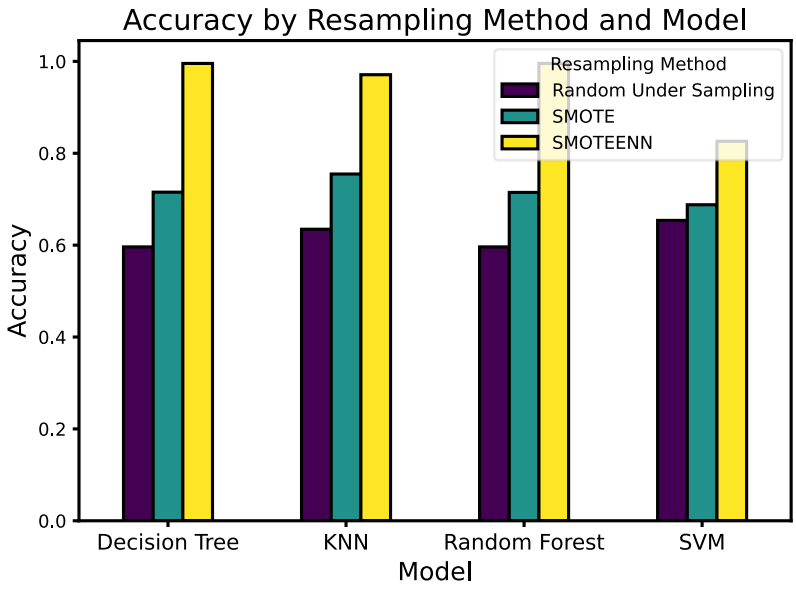


Figure 6. Accuracy.

SMOTE markedly enhances accuracy, with kNN achieving approximately 0.9 and RF nearing 0.93, highlighting the advantage of generating synthetic minority class samples to create a balanced dataset. However, SMOTEENN surpasses both techniques, delivering near-perfect accuracy of 1.0 for DT and kNN. RF and SVM also exhibit significant improvements, achieving accuracy values of 0.98 and 0.97, respectively. The results reaffirm SMOTEENN as the superior resampling method, effectively addressing class imbalance while maintaining data quality and improving model performance. SMOTEENN performance agrees with the work reported in [32] as it outperforms SMOTE in terms of accuracy and mean squared error across all the sample sizes and models.

Table 8 presents the performance metrics—accuracy, F1-score, recall (TPR), and AUC score—of four ML models (RF, DT, kNN, and SVM) under three resampling techniques: RUS, SMOTE, and SMOTEENN. These metrics provide a comprehensive understanding of how well the models address the class imbalance problem in the dataset.

Table 8. Performance metrics.

Method	ML Model	Accuracy	F1 Score	Recall (TPR)	AUC Score
RUS	Random forest	0.596154	0.589492	0.596154	0.775768
	Decision tree	0.596154	0.589492	0.596154	0.696218
	kNN	0.634615	0.60812	0.634615	0.782975
	SVM	0.653846	0.518115	0.653846	0.852747
SMOTE	Random forest	0.714756	0.715368	0.714756	0.888157
	Decision tree	0.715304	0.71586	0.715304	0.786466
	kNN	0.7548	0.751849	0.7548	0.890988
	SVM	0.687877	0.615475	0.687877	0.841265
SMOTEENN	Random forest	0.995541	0.995539	0.995541	0.998008
	Decision tree	0.995541	0.995539	0.995541	0.996743
	kNN	0.971014	0.97079	0.971014	0.99726
	SVM	0.826087	0.783482	0.826087	0.924708

Under RUS, all models exhibit suboptimal performance, with RF and DT showing identical results (accuracy: 0.596, F1-score: 0.589, recall: 0.596, AUC: ~0.77). The kNN model performs slightly better, achieving an accuracy of 0.634, but the improvement is minimal. SVM also performs poorly, with an F1-score of 0.518 and an AUC score of 0.85. The results demonstrate that RUS sacrifices critical majority class data, leading to compromised model performance.

The SMOTE technique significantly improves performance. For instance, RF's accuracy increases to 0.714, and kNN achieves an accuracy of 0.754 with an AUC score of 0.89. This improvement highlights SMOTE's effectiveness in generating synthetic samples to balance the dataset. However, SVM lags with a modest accuracy of 0.687 and an AUC of 0.84, indicating it benefits less from synthetic data compared to other models. These results align with the work done by [11].

SMOTEENN emerges as the most effective resampling method, yielding near-perfect performance for RF and DT (accuracy, recall, and F1-score: ~0.995, AUC: ~0.99). KNN also demonstrates excellent results with an accuracy of 0.971 and an AUC score of 0.997. Even SVM improves substantially under SMOTEENN, achieving an accuracy of 0.826 and an AUC score of 0.924. These results affirm SMOTEENN's ability to enhance both class balance and data quality, combining the strengths of SMOTE and ENNs to refine the dataset effectively. In comparison to the findings of [11], our results demonstrate that SMOTEENN consistently yielded the best performance across all ML models, further validating its effectiveness in handling class imbalance and improving classification accuracy. The consistent superior performance of SMOTEENN across all ML models demonstrated its robustness and effectiveness in handling the class imbalance, regardless of the number of instances per class. This finding is supported by the work in [32].

(b) The Receiver Operating Characteristics (ROC) Curves

Figure 7 presents the ROC curves for four ML models—RF, DT, kNN, and SVM—evaluated across three resampling techniques: RUS, SMOTE, and SMOTEENN. The ROC curves illustrate the models' TPR versus FPR at various threshold levels, offering insights into their discrimination capabilities. The dashed diagonal line represents the performance of a random classifier (AUC = 0.5), serving as a baseline for comparison.

In the RUS scenario, SVM achieves the highest AUC (0.96), followed by RF (0.95) and kNN (0.94). However, DT lags significantly with an AUC of 0.87, a trend also observed in [37]. This indicates that while RUS balances the classes by reducing the majority class, it does so at the expense of information loss, especially affecting simpler models like DT.

The SMOTE resampling technique shows a noticeable improvement for most models, though SVM and RF have slightly lower AUCs (0.94) compared to kNN (0.95). DT still struggles with an AUC of 0.85, highlighting that synthetic oversampling enhances model performance but might introduce noise in simpler models.

Finally, SMOTEENN delivers exceptional results, with RF, DT, and kNN achieving perfect AUC scores of 1.0. SVM also improves to an AUC of 0.97. The combination of SMOTE for oversampling and ENNs for noise removal effectively optimizes the dataset, enabling the models to achieve near-perfect classification performance. This demonstrates the robustness of SMOTEENN in addressing class imbalance and improving model discrimination capabilities for anomaly detection tasks. This observed performance is consistent with the findings [34], further reinforcing SMOTEENN's effectiveness in improving classification outcomes.

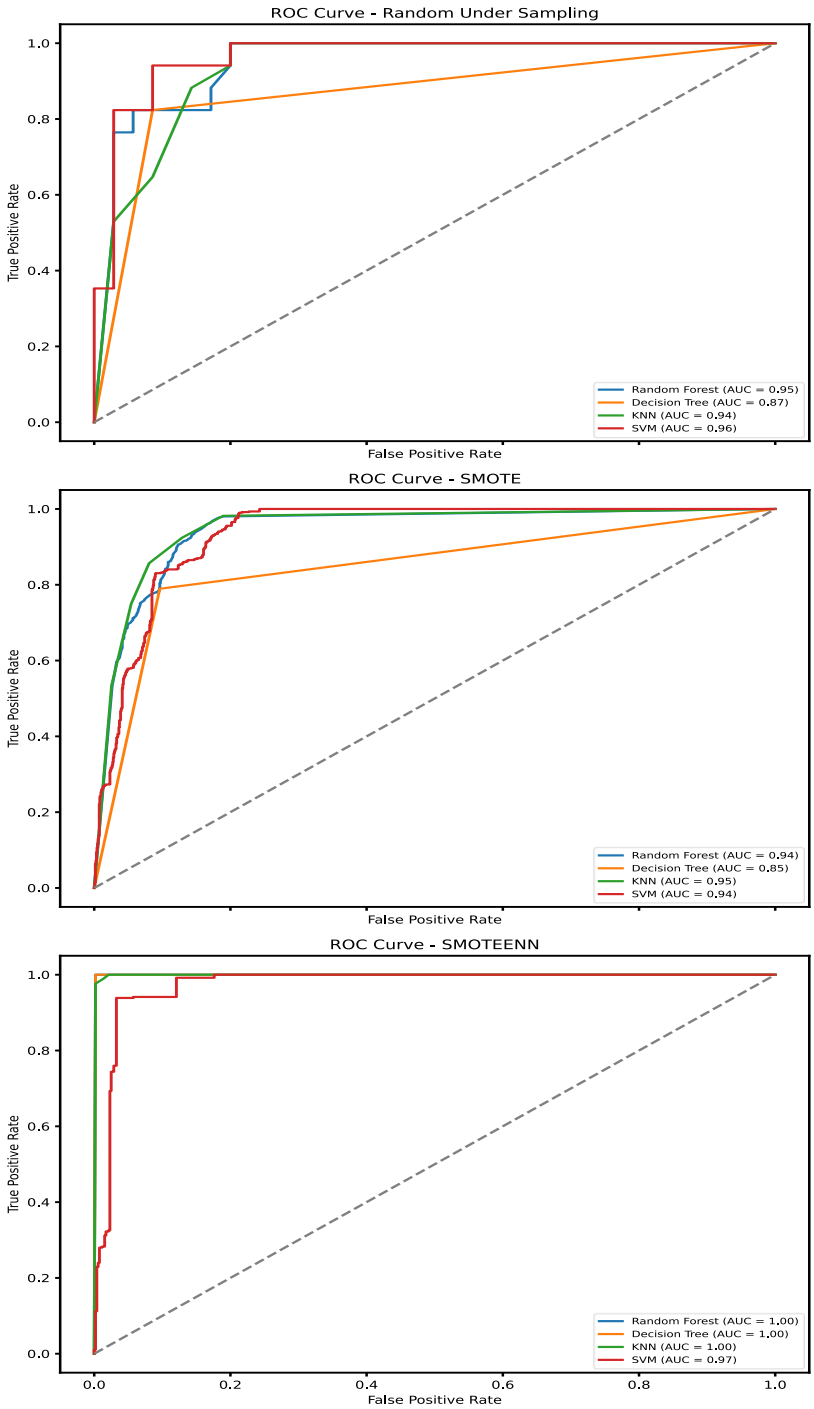


Figure 7. ROC-AUC curves.

(c) Confusion Matrix

Figure 8 shows the confusion matrices for the original, unbalanced dataset. It reveals insights into the performance of various ML models on unbalanced water consumption data, specifically kNN, SVM, RF, and DT classifiers, in detecting normal water consumption, spikes, and sudden changes. Each model exhibits strengths and weaknesses in handling these categories, and a detailed examination of TP, FP, TN, and FN is detailed. The k-NN model demonstrates exceptional accuracy in identifying normal water consumption with 398 true positives and only one false positive. However, it struggles with some false negatives and positives in detecting spikes and sudden changes. Similarly, the SVM model excels in spike detection with perfect true positives and no false negatives but fails to

detect any true positives for sudden changes, indicating a significant gap in its anomaly detection capabilities.

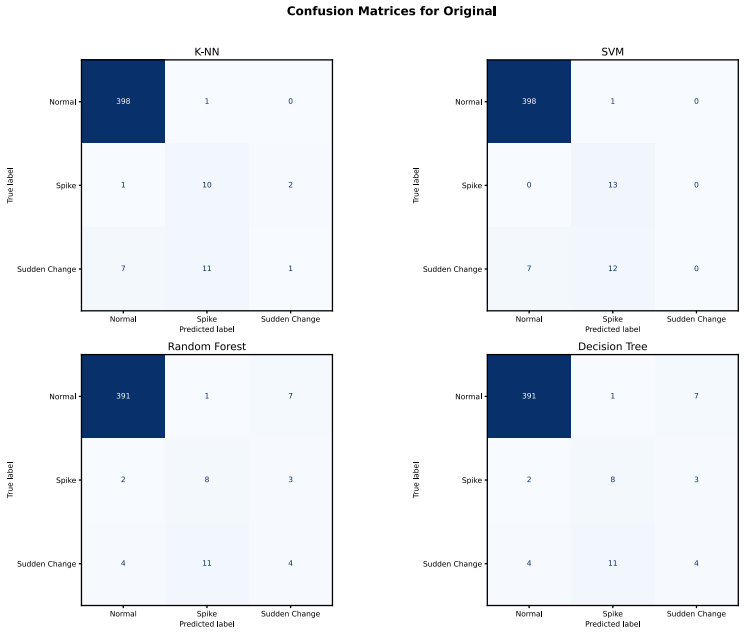


Figure 8. Confusion matrices for original unbalanced water consumption data.

The RF and DT models show a slight decrease in accuracy for normal consumption compared to kNN and SVM, with more false positives. RF, for example, has 391 true positives and eight false positives for normal consumption. It also shows moderate performance in spike detection, with a few false negatives and positives, and mixed results for sudden changes. The DT model exhibits similar behavior to RF, indicating consistent issues in handling anomalies. Both models struggle with distinguishing sudden changes, which is evident from their comparable numbers of TPs, FNs, and FPs. These findings highlight the need for advanced techniques to manage data imbalance and improve anomaly detection, which are discussed in the next sections.

The application of SMOTE to the water consumption dataset significantly impacts the performance of the ML models in detecting anomalies. Figure 9 shows the application of SMOTE on water consumption data with the aim of balancing the dataset. The kNN model shows improved performance in detecting spikes, with 304 TPs and only 67 FNs. However, it continues to struggle with sudden changes, as indicated by the high number of FNs (94) and FPs (80). Similarly, the SVM model exhibits excellent performance in spike detection, achieving perfect TPRs with 371 TPs and no FNs. Nonetheless, it faces challenges in identifying sudden changes, with 49 TPs but a substantial number of FNs (161) and FPs (198).

The RF and DT models also demonstrate enhanced performance with the balanced dataset provided by SMOTE. Both models show similar behavior, with 281 TPs for spikes and 238 TPs for sudden changes. However, they still have a considerable number of FNs and FPs, particularly for sudden changes. For instance, the RF model has 90 FNs and 80 FPs for sudden changes, while the DT model exhibits the same pattern. These results suggest that while SMOTE effectively balances the dataset and improves anomaly detection, especially for spikes, additional techniques or further refinement may be necessary to enhance the detection accuracy for sudden changes. Overall, these findings highlight the importance of using data balancing methods like SMOTE in combination with other strategies to improve the robustness and reliability of anomaly detection in SWMNs.

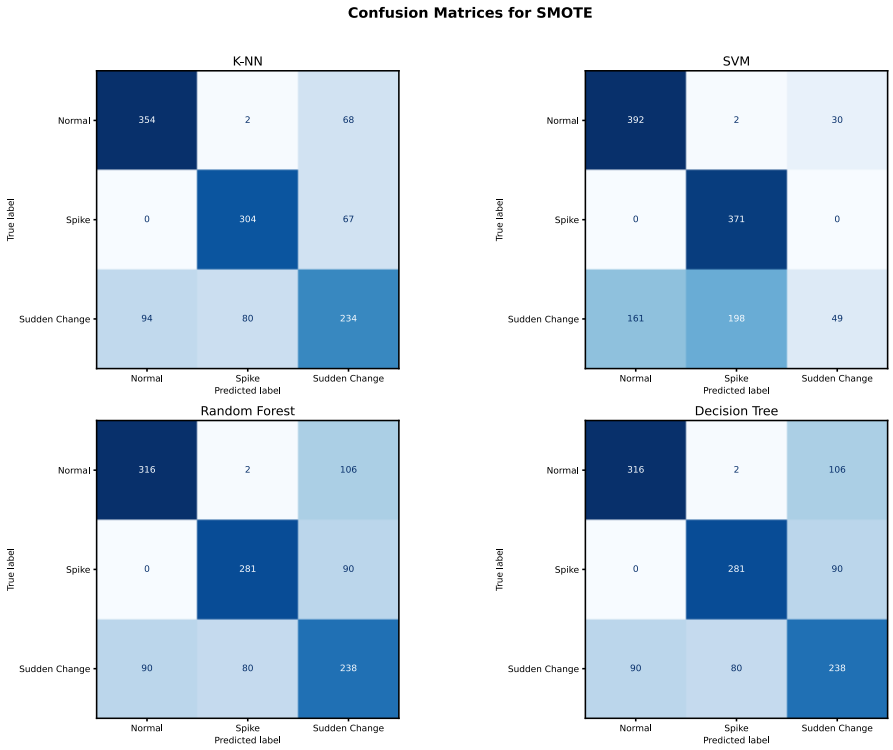


Figure 9. Confusion matrices for SMOTE balanced water consumption data.

The application of RUS to balance the water consumption dataset has a noticeable impact on the performance of the ML models, as illustrated in Figure 10. The kNN model demonstrates a balanced performance, accurately identifying normal water consumption with 9 TPs and no FPs and detecting spikes with 17 TPs and only 1 FN. However, it struggles with sudden changes, resulting in 5 TPs, 6 FNs, and 5 FPs. Similarly, the SVM model excels in spike detection with 18 TPs and no FNs, but it fails to identify any sudden changes correctly, leading to 10 FNs and 6 FPs. The RF and DT models also show improved performance in detecting normal consumption and spikes, but they exhibit similar difficulties in identifying sudden changes. Both models correctly identify 8 TPs for sudden changes but have FNs and FPs, indicating room for improvement.

Overall, RUS effectively balances the dataset and enhances the detection rates of normal consumption and spikes across all models. However, all models continue to face significant challenges in accurately detecting sudden changes, as reflected by the higher numbers of FNs and FPs in this category. These results underscore the complexity of sudden change detection in water consumption data and suggest that while RUS is a beneficial technique for balancing datasets, additional strategies or refinements are required to optimize model performance. Combining data balancing methods with advanced feature engineering or more sophisticated anomaly detection algorithms may lead to more robust and reliable results, ultimately enhancing the efficacy of smart water management systems.

The application of SMOTEENN, a combination of SMOTE and ENN, shows a significant impact on the performance of ML models in detecting anomalies in water consumption data. As shown in Figure 11, the kNN model exhibits excellent performance, accurately identifying normal water consumption with 224 TPs and no FPs or negatives. It also achieves perfect detection for spikes with 197 TPs and no FNs or positives. However, it detects sudden changes with slightly less accuracy, showing 118 TPs, 1 FN, and 3 FPs. The SVM model, while performing well in detecting normal consumption and spikes, shows some difficulty with sudden changes, evidenced by 31 TPs, 62 FNs, and 29 FPs.

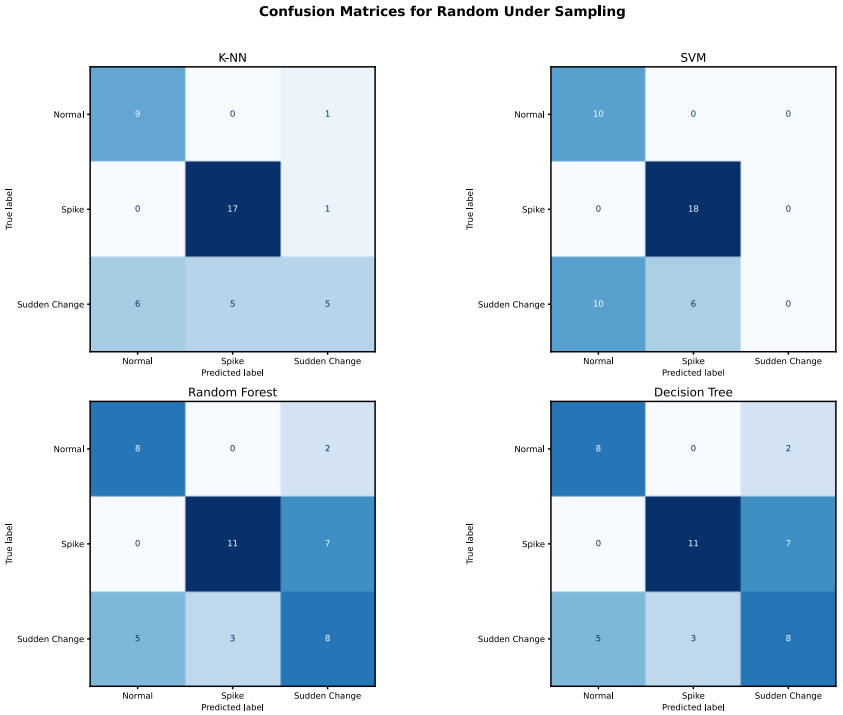


Figure 10. Confusion matrices for random undersampling balanced water consumption data.

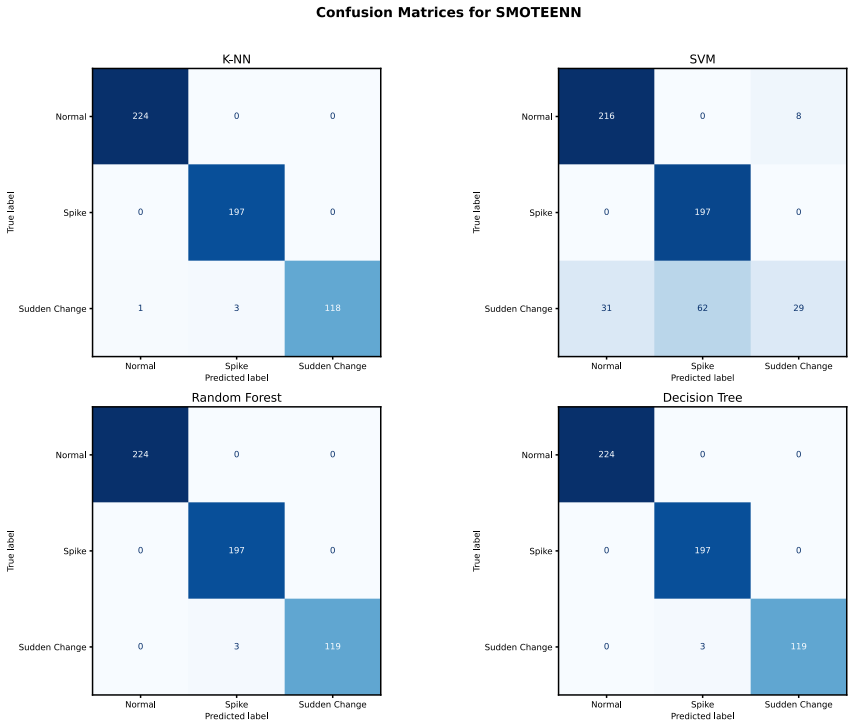


Figure 11. Confusion matrices for SMOTEENN balanced water consumption data.

The RF and DT models display similar performance patterns to those of kNN. Both models correctly identify normal consumption and spikes with 224 and 197 TPs, respectively, without any FPs. For sudden changes, the RF model shows 119 TPs, 3 FNs, and no FPs, while the DT model mirrors this performance. These results suggest that SMOTEENN significantly enhances the models' ability to detect anomalies, particularly in balancing detection accuracy for normal consumption and spikes. However, kNN appears to be the most consistent model overall, given its balanced performance and minimal FPs and negatives

across all categories. Therefore, SMOTEENN was selected to balance the water consumption dataset, as it provides a robust approach to mitigating the challenges of data imbalance and improving the accuracy and reliability of anomaly detection in water consumption patterns. These results align with the findings of [38], where SMOTEENN demonstrated superior performance, further supporting its effectiveness in distinguishing between TP, TN, FP, and FN. Additionally, the findings indicate that RF yielded the best overall performance, reinforcing its suitability for anomaly detection in imbalanced datasets.

(d) Ensemble ML Classifiers

The implementation of ensemble ML classifiers, specifically stacking and voting (hard and soft), provides a robust framework for detecting anomalies and normal water consumption patterns in SWMNs. Figure 12 highlights the comparative performance of these classifiers across key metrics: accuracy, precision, recall (TPR), and F1-score. Stacking demonstrates superior performance across all metrics, leveraging its architecture to combine multiple base classifiers through a meta-classifier. This method effectively learns complex relationships between the predictions of base classifiers, leading to a more refined decision boundary. The stacking classifier's ability to mitigate both variance and bias results in improved generalization and robustness, making it particularly effective for anomaly detection in the imbalanced datasets typical of SWMNs.

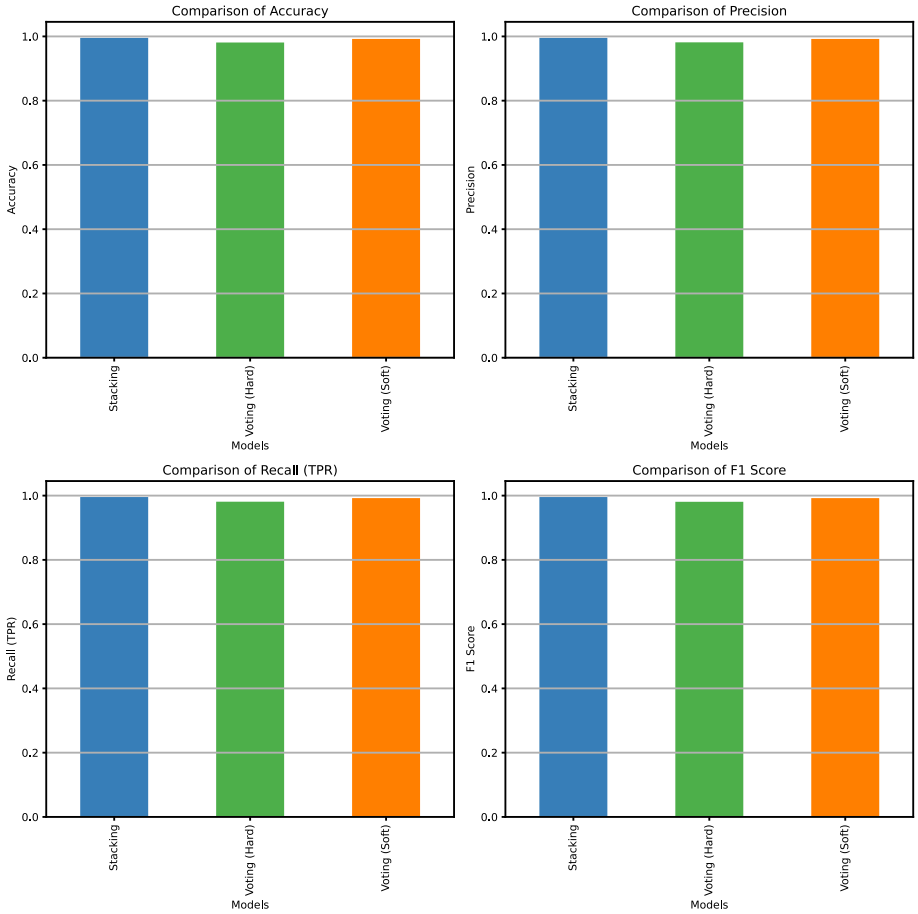


Figure 12. Ensemble ML.

In contrast, voting classifiers exhibit high yet slightly lower performance compared to stacking. Hard voting aggregates predictions based on majority rule, which ensures robustness but may overlook the contributions of stronger base classifiers due to equal weighting. Soft voting, on the other hand, uses probability outputs from base classifiers to

make decisions, offering more flexibility. However, the slightly lower performance metrics for voting suggest that it is less adept at integrating model-specific strengths compared to stacking. Overall, the experimental results underscore the effectiveness of ensemble methods, with stacking emerging as the most reliable due to its adaptability and capability to handle complex datasets. These findings reinforce the value of ensemble approaches in enhancing the accuracy and reliability of anomaly detection systems, a critical requirement for optimizing water resource management in SWMNs.

The performance of the ensemble ML models, stacking, voting (hard), and voting (soft) was evaluated using key metrics: accuracy, precision, recall (true positive rate), and F1 score. In Table 9, the stacking model achieved the highest performance across all metrics, with an accuracy of 99.55% and an F1 score of 99.55%. This exceptional performance is due to the stacking method's ability to combine the strengths of multiple base models and a meta-classifier to make robust predictions. The high recall indicates the model's effectiveness in detecting true anomalies without significant false negatives, making it ideal for applications where both precision and recall are critical. Hard voting model achieved an accuracy of 98.10%, with precision and F1 scores slightly lower than the stacking and soft voting models. The hard voting method determines predictions based on the majority vote of its base models, which is straightforward and computationally efficient. While it performs well, its lower precision and recall indicate that it may not be as effective in handling edge cases or ambiguous data points. The soft voting model achieved strong performance, with an accuracy of 99.22% and an F1 score of 99.22%, slightly lower than the stacking model. Unlike hard voting, soft voting uses the predicted probabilities of base models, allowing it to make more informed predictions. This approach is particularly effective in capturing decision boundaries, making it a strong competitor to the stacking model.

Table 9. Performance metrics for ensemble ML.

Model	Accuracy	Precision	Recall	F1 Score
Stacking	0.995541	0.995540	0.995541	0.995539
Voting (hard)	0.981048	0.981517	0.981048	0.980723
Voting (soft)	0.992196	0.992182	0.992196	0.992172

The stacking model emerged as the most reliable and effective approach for anomaly detection in the context of SWMNs, followed closely by the soft voting model. These results validate the potential of ensemble ML methods in building robust systems for anomaly detection. The results align with the findings presented in [39], where voting and stacking ensemble ML methods demonstrated the best performance. This further supports the effectiveness of ensemble learning techniques in enhancing classification accuracy and handling class imbalance in anomaly detection tasks.

4. Conclusions

This study investigated the practical efficacy of using RUS, SMOTE, and SMOTEENN in addressing class imbalance in ML-based anomaly detection using water consumption data from Location A in Windhoek, Namibia. The research systematically evaluated both individual ML models and ensemble learning approaches under different resampling techniques to determine their effectiveness in imbalanced classification tasks relevant to SWMNs. Findings indicate that class imbalance significantly affects the accuracy and reliability of anomaly detection models in real-world datasets. The results show that SMOTEENN, a hybrid technique that combines oversampling and undersampling, consistently outperforms both SMOTE and RUS across all tested models. When applied with RF, SMOTEENN achieved the highest performance with an accuracy of 99.55% and an AUC score of

0.998. Among the ensemble methods, the stacking ensemble also reached 99.6% accuracy and an F1-score of 99.6%, confirming its robustness in capturing imbalanced anomalies. These quantitative results confirm that SMOTEENN and ensemble learning models significantly enhance predictive performance in imbalanced data settings, reinforcing their practical value in data-scarce environments. Future research should explore additional hybrid strategies that extend beyond SMOTEENN, particularly by combining ensemble learning with active learning or dynamic sampling approaches. Such methods may further improve the generation and selection of synthetic examples, especially near decision boundaries. The evaluation of SMOTEENN and ensemble models across other domains beyond water consumption also presents a valuable avenue for generalizing findings and optimizing model adaptability. Finally, the integration of Explainable AI (XAI) techniques within anomaly detection frameworks is strongly recommended. Improving model interpretability will help ensure that AI-driven water governance tools remain transparent, accountable, and suitable for real-world policy and operational decision-making.

Author Contributions: M.N.K.: conceptualization, data curation, formal analysis, investigation, methodology, validation, visualization, writing original draft, F.B.S.: conceptualization, methodology, supervision, writing—review and editing, A.M.G.: conceptualization, supervision, and editing A.H.: supervision. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge funding for this project by the Southern African Science Service. Centre for Climate Change and Adaptive Land Management (SASSCAL), sponsored by the German. Government through the Federal Ministry of Education and Research (BMBF) with funding number 01LG2091A. Additional funding was provided by the Namibia University of Science and Technology, Faculty of Computing and Informatics.

Data Availability Statement: The datasets used and analyzed during the current study are not publicly available due to confidentiality agreements with the municipal water utility.

Acknowledgments: We extend our gratitude to the City of Windhoek Water Department and the Namibia University of Science and Technology for their invaluable support, research resources, and essential data, which were instrumental to this study.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ramotsoela, D.; Abu-Mahfouz, A.; Hancke, G. A survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study. *Sensors* **2018**, *18*, 2491. [CrossRef] [PubMed]
2. Okoli, N.J.; Kabaso, B. Building a Smart Water City: IoT Smart Water Technologies, Applications, and Future Directions. *Water* **2024**, *16*, 557. [CrossRef]
3. Kanyama, M.N.; Shava, F.B.; Gamundani, A.M.; Hartmann, A. Machine learning applications for anomaly detection in Smart Water Metering Networks: A systematic review. *Phys. Chem. Earth* **2024**, *134*, 103558. [CrossRef]
4. Kanyama, M.N.; Shava, F.B.; Gamundani, A.M.; Hartmann, A. Anomalies identification in Smart Water Metering Networks: Fostering improved water efficiency. *Phys. Chem. Earth* **2024**, *134*, 103592. [CrossRef]
5. Marar, R.W.; Marar, H.W. A reliable algorithm for Efficient Water Delivery and Smart Metering in Water-Scarce Regions. *Asian J. Water Environ. Pollut.* **2024**, *21*, 1–9. [CrossRef]
6. Mankad, U.; Arolkar, H. Smart Water Metering Implementation. In *Smart Trends in Computing and Communications*; Lecture Notes in Networks and Systems; Springer Science and Business Media: Berlin, Germany, 2023; pp. 721–731. [CrossRef]
7. Beal, C. *The 2014 Review of Smart Metering and Intelligent Water Networks in Australia & New Zealand*; Water Services Association of Australia: Docklands, Australia, 2014. [CrossRef]
8. Ogboh, V.C.; Ogboke, H.N.; Obiora-Okeke, C.A.; Nwoye, N.A. Design and Implementation of IoT Based Smart Meter. *Int. J. Eng. Invent.* **2024**, *13*, 158–168. Available online: www.ijeijournal.com (accessed on 20 February 2025).
9. Nyirenda, C.N.; Makwara, P.; Shitumbapo, L. Particle Swarm Optimization Based Placement of Data Acquisition Points in a Smart Water Metering Network. In *Proceedings of SAI Intelligent Systems Conference*; Lecture Notes in Networks and Systems; Springer: Cham, Switzerland, 2018; Volume 16, pp. 905–916. [CrossRef]

10. Shitumbapo, L.N.; Nyirenda, C.N. Simulation of a Smart Water Metering Network in Tsumeb East, Namibia. In Proceedings of the 2015 International Conference on Emerging Trends in Networks and Computer Communications, ETNCC 2015, Windhoek, Namibia, 17–20 May 2015; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2015; pp. 44–49. [CrossRef]
11. Srivastava, J.; Sharan, A. SMOTEEN Hybrid Sampling Based Improved Phishing Website Detection. *TechRxiv* **2022**, preprint. [CrossRef]
12. Abayomi-Alli, O.O.; Damasevicius, R.; Maskeliunas, R.; Abayomi-Alli, A. BiLSTM with Data Augmentation using Interpolation Methods to Improve Early Detection of Parkinson Disease. In Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, FedCSIS 2020, Sofia, Bulgaria, 6–9 September 2020; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 371–380. [CrossRef]
13. Kanyama, M.N.; Shava, F.B.; Gamundani, A.M.; Hartmann, A. Enhancing Anomaly Detection in Smart Water Metering Networks with LSTM-Autoencoder and Data Augmentation Techniques. In *2024 4th International Multidisciplinary Information Technology and Engineering Conference (IMITEC), Vanderbijlpark, South Africa, 27–29 November 2024*; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2024; pp. 20–28. [CrossRef]
14. Oladipupo, T. Types of Machine Learning Algorithms. In *New Advances in Machine Learning*; InTechOpen: Londra, UK, 2010. [CrossRef]
15. Fang, S.; Sun, W.; Huang, L. Anomaly Detection for Water Supply Data using Machine Learning Technique. *J. Phys. Conf. Ser.* **2019**, *1345*, 022054. [CrossRef]
16. Garmaroodi, M.S.S.; Farivar, F.; Haghighi, M.S.; Shoorehdeli, M.A.; Jolfaei, A. Detection of Anomalies in Industrial IoT Systems by Data Mining: Study of CHRIST Osmotron Water Purification System. *IEEE Internet Things J.* **2021**, *8*, 10280–10287. [CrossRef]
17. Iyer, S.; Thakur, S.; Dixit, M.; Katkam, R.; Agrawal, A.; Kazi, F. Blockchain and Anomaly Detection based Monitoring System for Enforcing Wastewater Reuse. 2019. Available online: https://github.com/sreeragiyer/Wastewater-Reuse (accessed on 15 January 2025).
18. Mahmoud, H.; Wu, W.; Gaber, M.M. A Time-Series Self-Supervised Learning Approach to Detection of Cyber-physical Attacks in Water Distribution Systems. *Energies* **2022**, *15*, 914. [CrossRef]
19. Mounce, S.R.; Pedraza, C.; Jackson, T.; Linford, P.; Boxall, J.B. Cloud based machine learning approaches for leakage assessment and management in smart water networks. *Procedia Eng.* **2015**, *119*, 43–52. [CrossRef]
20. Ramotsoela, T.D.; Hancke, G.P.; Abu-Mahfouz, A.M. Behavioural Intrusion Detection in Water Distribution Systems Using Neural Networks. *IEEE Access* **2020**, *8*, 190403–190416. [CrossRef]
21. Ramotsoela, D.T.; Hancke, G.P.; Abu-Mahfouz, A.M. Attack detection in water distribution systems using machine learning. *Hum.-Centric Comput. Inf. Sci.* **2019**, *9*, 13. [CrossRef]
22. Taormina, R.; Galelli, S. Deep-Learning Approach to the Detection and Localization of Cyber-Physical Attacks on Water Distribution Systems. *J Water Resour. Plan. Manag.* **2018**, *144*, 04018065. [CrossRef]
23. Wang, D.; Wang, P.; Zhou, J.; Sun, L.; Du, B.; Fu, Y. Defending water treatment networks: Exploiting spatio-temporal effects for cyber attack detection. In Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM), Sorrento, Italy, 17–20 November 2020; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 32–41. [CrossRef]
24. Luque, A.; Carrasco, A.; Martín, A.; Lama, J.R. Exploring symmetry of binary classification performance metrics. *Symmetry* **2019**, *11*, 47. [CrossRef]
25. De Diego, I.M.; Redondo, A.R.; Fernández, R.R.; Navarro, J.; Moguerza, J.M. General Performance Score for classification problems. *Appl. Intell.* **2022**, *52*, 12049–12063. [CrossRef]
26. Powers, D.M.W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63. [CrossRef]
27. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process Manag.* **2009**, *45*, 427–437. [CrossRef]
28. Powers, D. Evaluation_From_Precision_Recall_and_F-Factor_to_R. *Mach. Learn. Technol.* **2008**, *2*.
29. Vujović, Ž. Classification Model Evaluation Metrics. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 599–606. [CrossRef]
30. Liu, Y.; Zhou, Y.; Wen, S.; Tang, C. A Strategy on Selecting Performance Metrics for Classifier Evaluation. *Int. J. Mob. Comput. Multimed. Commun.* **2014**, *6*, 20–35. [CrossRef]
31. Wongvorachan, T.; He, S.; Bulut, O. A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining. *Information* **2023**, *14*, 54. [CrossRef]
32. Husain, G.; Nasef, D.; Jose, R.; Mayer, J.; Bekbolatova, M.; Devine, T.; Toma, M. SMOTE vs. SMOTEENN: A Study on the Performance of Resampling Algorithms for Addressing Class Imbalance in Regression Models. *Algorithms* **2025**, *18*, 37. [CrossRef]
33. Muaz, A.; Jayabalan, M.; Thiruchelvam, V. A Comparison of Data Sampling Techniques for Credit Card Fraud Detection. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 477–485. Available online: www.ijacsa.thesai.org (accessed on 20 December 2024). [CrossRef]
34. Singh, A.; Ranjan, R.K.; Tiwari, A. Credit Card Fraud Detection under Extreme Imbalanced Data: A Comparative Study of Data-level Algorithms. *J. Exp. Theor. Artif. Intell.* **2022**, *34*, 571–598. [CrossRef]

35. Kim, M.; Hwang, K.B. An empirical evaluation of sampling methods for the classification of imbalanced data. *PLoS ONE* **2022**, *17*, e0271260. [CrossRef]
36. Kudithipudi, S.; Narisetty, N.; Kancherla, G.R.; Bobba, B. Evaluating the Efficacy of Resampling Techniques in Addressing Class Imbalance for Network Intrusion Detection Systems Using Support Vector Machines. *Ing. Syst. d'Inform.* **2023**, *28*, 1229–1236. [CrossRef]
37. Fernando, C.D.; Weerasinghe, P.T.; Walgampaya, C.K. Heart Disease Risk Identification using Machine Learning Techniques for a Highly Imbalanced Dataset: A Comparative Study. *KDU J. Multidiscip. Stud.* **2022**, *4*, 43–55. [CrossRef]
38. Ako, R.E.; Aghware, F.O.; Okpor, M.D.; Akazue, M.I.; Yoro, R.E.; Ojugo, A.A.; Setiadi, D.R.I.M.; Odiakaose, C.C.; Abere, R.A.; Emordi, F.U.; et al. Effects of Data Resampling on Predicting Customer Churn via a Comparative Tree-based Random Forest and XGBoost. *J. Comput. Theor. Appl.* **2024**, *2*, 86–101. [CrossRef]
39. Misr International University (MIU). In Proceedings of the 2024 International Mobile, Intelligent, And Ubiquitous Computing Conference, Cairo, Egypt, 8–9 May 2024; IEEE: Piscataway, NJ, USA, 2024.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.