eng

MDPI

Article

Adaptive Gradient-Boosting-Based Sensor Selection for Energy-Efficient Wireless Sensor Networks (WSNs)

Sara Khalil Ibrahim [1], Rashid S. Jasim [2], Aliaa Saad Aljubair [3], Hussein A. Jasim [1,*], Mohamed Abdulrahman Abdulhamed [1] and Dhuha Habeeb [4]

[1] Computer Sciences Department, College of CSIT, University of Basrah, Basrah 61004, Iraq
[2] Department of Computer Engineering Techniques, Shatt Al-Arab University College, Basra 61014, Iraq
[3] Information Systems Department, College of CSIT, University of Basrah, Basrah 61004, Iraq
[4] Department of Laser and Optoelectronics Technical Engineering, Shatt Al-Arab University College, Basra 61014, Iraq
* Correspondence: hussein.jasim@uobasrah.edu.iq

Abstract

Energy conservation remains a fundamental challenge in wireless sensor networks (WSNs), where battery-powered nodes are often deployed in locations where recharging is impractical. Because spatially correlated sensors frequently report redundant readings, deactivating less informative sensors can extend network lifetime while retaining nearly the full-network classification performance. Prior sensor-selection studies, however, have often evaluated fixed sensor counts on public classification benchmarks without ground-truth labels identifying which sensors are genuinely informative, while estimating energy savings primarily from sensor-count ratios rather than topology-dependent physical energy models. This paper presents an adaptive sensor-selection framework that uses gradient-boosted trees with permutation importance to rank sensors and identify the smallest subset that retains a predefined fraction of full-network classification accuracy. The framework is evaluated using a purpose-built synthetic benchmark with known ground-truth sensor informativeness and an explicit network topology governed by a first-order radio energy model, together with eight independently re-implemented baseline methods evaluated under identical conditions across eight data-partition seeds. At the selected operating point, GBM-Permutation retains 99.3% of the full-sensor classification accuracy (84.6% versus 85.2%) using only 10 of 54 sensors, corresponding to an 82.6% reduction in modeled per-round energy consumption; modeled first-node-dies network lifetime increases from 615 rounds (using all 54 sensors) to 883 rounds (a 1.4-fold increase), distinct from the sensor-count-based Lifetime Extension Factor (LEF) of 5.4. The proposed method significantly outperforms all baseline methods in the primary synthetic evaluation ($p < 0.01$). A redundancy–severity sensitivity analysis and validation on two generic real-world tabular datasets and one real sensor-derived dataset show that the advantage is strongest under moderate redundancy rather than being universal. These findings demonstrate the value of ground-truth-validated and topology-aware evaluation for identifying sensor-selection methods that are effective under energy-constrained WSN conditions.

Keywords: energy conservation; feature selection; gradient boosting; network lifetime; redundancy reduction; sensor selection; wireless sensor networks (WSNs)

Academic Editor: Huanyu Cheng

Received: 28 July 2026
Revised: 8 September 2026
Accepted: 15 September 2026
Published: 21 September 2026

1. Introduction

Wireless Sensor Networks (WSNs) underpin a wide range of monitoring applications, including environmental sensing, industrial automation, precision agriculture, and infrastructure monitoring, by distributing battery-powered nodes that sense, process, and relay data to a central sink [1]. Because these nodes are frequently deployed in locations where battery replacement is impractical or impossible, energy efficiency remains one of the most persistent constraints on WSN design, and radio transmission continues to dominate per-node energy budgets far more than local sensing or computation [2,3].

A substantial body of work addresses this constraint at the protocol level: duty cycling, clustering, and data aggregation reduce energy consumption by limiting when nodes transmit [4]. A complementary, data-driven line of work instead asks which sensors need to transmit at all: since spatially or physically correlated sensors often report highly redundant readings, a classifier trained to identify the most informative subset of sensors can allow the rest to sleep without materially degrading the network's ability to classify or monitor its environment. This sensor-selection paradigm, and its associated Lifetime Extension Factor (LEF = total sensors/sensors used), originates with Barnawi and Keshta [5], who compared Naïve Bayes, MLP, and SVM classifiers for ranking sensor importance, and has continued to be extended with newer feature-selection techniques, most recently minimum-redundancy–maximum-relevance (mRMR) ranking combined with SVM, Naïve Bayes, and k-NN classifiers [6].

Gradient-boosted decision trees and XGBoost specifically [7] are a natural fit for this problem: they provide built-in, non-linear feature-importance ranking, handle heterogeneous sensor data well, and have already demonstrated strong results in adjacent WSN and IoT tasks, including fault diagnosis [8], adaptive radio parameter allocation in LoRaWAN [9], and intrusion detection [10]. However, existing applications of gradient boosting in WSNs have concentrated on these adjacent tasks rather than on sensor selection for energy conservation itself, and the sensor-selection literature has generally evaluated a single, arbitrarily fixed number of retained sensors rather than characterizing the full accuracy–energy trade-off curve [11], nor has it validated selection quality against a known ground truth of which sensors are actually informative—a gap that is structural to the field's reliance on repurposed public classification benchmarks, where no such ground truth exists.

This paper addresses these gaps directly. Rather than reuse an unrelated classification benchmark as a sensor-count proxy, we construct a controlled simulation in which the true informativeness of every virtual sensor is known by design, letting us measure not just classification accuracy but whether a selection method actually recovers the sensors that matter. We combine this with a first-order radio energy model [12] applied to an explicit network topology, so that energy and lifetime claims are grounded in physical transmission cost rather than in sensor-count ratios alone. Our specific contributions are as follows:

Consistent with the position that gradient boosting combined with permutation importance is, on its own, a well-established machine-learning pipeline rather than a novel algorithm, the primary contributions of this paper are the WSN-specific adaptation and the ground-truth-validated evaluation framework built around it (Contributions 2–5 below), with the specific selection method (Contribution 1) presented as the instantiation used to demonstrate that framework. Section 4.7 reports an ablation confirming that permutation importance itself is not statistically distinguishable from a gain-based alternative within the same classifier, directly supporting this framing.

1. An adaptive, gradient-boosting-based sensor-selection method that ranks sensors by permutation importance [13], chosen over impurity-based importance specifically to avoid the well-documented bias of the latter toward high-cardinality features, and

selects the smallest subset retaining a target fraction of full-network accuracy rather than an arbitrarily fixed count. An ablation study (Section 4.7) confirms this specific importance mechanism is not itself the source of the method's advantage over other model families, consistent with the framing above.

2. A redundancy-aware synthetic WSN benchmark with known ground-truth sensor informativeness, enabling direct, quantitative evaluation of selection quality (precision/recall against ground truth) in addition to downstream accuracy, something not possible with existing benchmarks in this literature.
3. A physically grounded energy and network-lifetime analysis using a standard radio energy model [12], replacing the count-only LEF ratio used in prior work with per-topology, per-selection energy and first-node-dies lifetime estimates.
4. A comprehensive empirical comparison against eight baselines spanning filter (mutual information, variance, mRMR [14]), embedded (Random Forest importance, Lasso), unsupervised (PCA, autoencoder reconstruction), and naïve (random) selection families, evaluated under identical conditions across eight random seeds with paired statistical significance testing.
5. A sensitivity analysis across three sensor-redundancy regimes (high, medium, and low), together with supplementary validation on two real-world tabular benchmarks and one real sensor-derived dataset, characterizing the conditions under which the GBM-Permutation approach provides its greatest advantage and those in which it is not statistically distinguishable from simpler alternatives.

The remainder of this paper is organized as follows. Section 2 reviews related work on energy conservation, sensor selection, and gradient boosting in WSNs. Section 3 describes the simulation framework, energy model, and evaluation protocol. Section 4 presents results, including the accuracy–energy Pareto analysis, ablation of selection quality, and redundancy–sensitivity findings. Section 5 discusses limitations and practical implications, and Section 6 concludes.

2. Related Work

2.1. Energy Conservation in Wireless Sensor Networks

Classical approaches to WSN energy conservation operate primarily at the protocol level. Duty-cycling and MAC-layer scheduling reduce idle-listening overhead, clustering and data aggregation reduce the volume of transmitted data, and routing protocols minimize per-hop transmission cost [4]. Nkemeni et al. [3] provide a recent experimental demonstration of this family of techniques, showing that combining distributed computing, hierarchical sensing, and duty cycling can reduce a WSN sensor node's energy consumption by a factor of eight, relative to a single technique in isolation, in a real deployed water-pipeline monitoring system. These protocol-level techniques are complementary to, rather than competitive with, sensor-selection approaches: a network can apply duty cycling and select which sensors participate in a sensing round at all.

2.2. Machine-Learning-Based Sensor Selection

The specific paradigm of ranking and selecting sensors by predictive importance, then quantifying the resulting benefit via a Lifetime Extension Factor, was introduced by Barnawi and Keshta [5], who compared Naïve Bayes, MLP, and SVM classifiers and found that, for a matched LEF, SVM-selected sensors yielded better energy efficiency than the other two. This line of work has continued to evolve with newer feature-ranking techniques: Aljawarneh et al. [6] recently (2024) replaced the classifier-driven ranking with minimum-redundancy–maximum-relevance (mRMR) feature selection, evaluated against SVM, Naïve Bayes, and k-NN classifiers on the Ionosphere and Gas Sensor Array Drift

datasets, and reported consistent LEF improvements as sensor counts were reduced. This paper is the most directly comparable prior work to ours, and in Section 4 we include a re-implementation of mRMR as one of our baseline methods, evaluated under identical conditions to the rest.

A related direction targets sensor/feature redundancy more explicitly at the selection stage rather than treating it as an incidental byproduct of importance ranking. Saha and Pal [15] propose a neural-network-based group-feature selection method with explicit redundancy control, directly motivating the redundancy–severity sensitivity analysis.

2.3. Gradient Boosting and XGBoost in Wireless Sensor Networks and IoT

XGBoost [7] and related gradient-boosting methods have recently been applied to several WSN- and IoT-adjacent tasks. Nagarajan and SVN [8] combine XGBoost with whale optimization for sensor fault diagnosis and classification. Nisar et al. [9] use XGBoost to predict optimal LoRaWAN spreading-factor allocation from network conditions (RSSI, SNR, device coordinates), explicitly optimizing an energy/packet-delivery trade-off via an ns-3-based simulation methodology conceptually similar to the simulation-based approach we adopt here. Ouhmi et al. [10] benchmark six boosting variants, including XGBoost, for WSN intrusion detection under class imbalance. Collectively, this work establishes that gradient-boosted trees are both computationally practical and empirically effective in resource-constrained WSN settings but, as noted above, none of them target sensor selection for energy conservation specifically, which is the gap this paper addresses.

2.4. Reconstruction-Based and Redundancy-Reduction Approaches

A separate line of work reduces energy consumption by using autoencoder-style reconstruction to identify or exploit redundancy, rather than a supervised classification objective. Because these methods are trained purely to reconstruct sensor readings rather than to preserve classification-relevant information, they can allocate representational capacity in ways that are misaligned with downstream task performance—a limitation we return to and empirically demonstrate in Section 4, where a reconstruction-based baseline is shown to prioritize noise sensors that are simply harder to reconstruct, rather than sensors that are actually useful for classification. Recent surveys of the broader green/ML-enhanced WSN landscape [4] and of ensemble-based secure routing approaches [16] confirm that machine-learning methods are now applied across nearly every layer of WSN design (routing, clustering, security, and data reduction), reinforcing the case that sensor-level selection deserves the same rigor of evaluation in multi-seed statistical testing, honest baselines, and ground-truth validation that has become standard elsewhere in the field.

2.5. Research Gap

Across this literature, three gaps recur: (i) sensor-selection studies are evaluated on public classification benchmarks with no ground truth for sensor informativeness, making it impossible to distinguish "selects sensors that happen to classify well" from "selects the sensors that are actually informative"; (ii) energy and lifetime claims are typically derived from sensor-count ratios rather than a physical energy model tied to network topology; and (iii) comparisons across methods are rarely backed by multi-seed statistical testing or fair, identically-evaluated baselines. The representative studies reviewed in this section are summarized in Table 1. The remainder of this paper presents a simulation framework and evaluation protocol designed specifically to close all three gaps.

Table 1. Summary of related work.

Ref.	Year	Method	Dataset/Setting	Energy Model	Limitations
[3]	2023	Duty cycling + hierarchical sensing	Real hardware (ESP32 testbed)	Yes (measured)	Protocol-level; not sensor selection
[4]	2025	Systematic review (no new experiments)	N/A	—	Confirms ML breadth across WSN layers
[5]	2016	NB/MLP/SVM ranking + LEF	WSN sensor dataset	No	Originated LEF; count-ratio only, no ground truth
[6]	2024	mRMR + SVM/NB/k-NN	Ionosphere; Gas Sensor Drift	No	Closest prior work; no ground truth, no energy model, single run
[8]	2025	XGBoost + Whale Optimization	WSN fault-diagnosis data	No	Fault diagnosis, not sensor selection
[9]	2025	XGBoost for spreading-factor allocation	ns-3 LoRaWAN simulation	Yes (simulated)	LoRaWAN parameters, not sensor-level selection
[10]	2026	6 boosting variants + GA feature select	WSN-DS intrusion dataset	No	Intrusion detection, not energy conservation
[15]	2024	NN group-feature selection	General FS benchmarks	No	Motivates redundancy analysis; not WSN energy-specific
[16]	2025	Ensemble selection + Type-2 fuzzy	Simulated WSN routing	Partial (routing-level)	Routing layer, not sensor layer

N/A, not applicable.

3. Methodology

3.1. Overview

The framework comprises four components addressing the identified limitations: (1) a synthetic wireless-sensor benchmark with predefined ground-truth sensor informativeness, (2) a first-order radio energy model applied to an explicit network topology, (3) an adaptive sensor-selection procedure based on gradient-boosting permutation importance, and (4) an evaluation protocol incorporating eight independently re-implemented baselines, repeated-seed statistical testing, redundancy–sensitivity analysis, and validation on real datasets, as illustrated in Figure 1.

3.2. Simulated Sensor Network Benchmark

Public classification benchmarks repurposed as sensor-selection testbeds (e.g., Forest CoverType, Ionosphere, Gas Sensor Array Drift) have no ground truth for which features are genuinely informative versus redundant versus noise—a method's selections can only be judged by downstream accuracy, which conflates selection quality with classifier capacity. We instead generate a benchmark with a known generative structure.

Six latent environmental factors $z \in \mathbb{R}^6$ drive a 7-class classification problem. Class labels y are drawn from a fixed, uneven class-prior distribution w = [0.36, 0.21, 0.18, 0.09, 0.07, 0.05, 0.04] (chosen to mirror the class imbalance typical of environmental-monitoring data), and each class c is associated with a fixed mean vector $\mu_c \in \mathbb{R}^6$ (drawn once from $\mathcal{N}(0, 1.6^2)$, seed-fixed for reproducibility). Given a sampled label, $z = \mu_y + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$.

Fifty-four virtual sensors are generated from z in three groups:

- Group A: direct (6 sensors): one low-noise sensor per latent factor, $x = z_k + \mathcal{N}(0, 0.3^2)$. These are the ground-truth "clean, non-redundant" sensors.
- Group B: redundant (30 sensors): five additional, noisier sensors per latent factor, $x = z_k + \mathcal{N}(0, \sigma^2)$, $\sigma \sim$ Uniform(0.4, 1.1). These simulate spatially co-located sensors that duplicate signals already present in Group A.

- Group C: noise (18 sensors): $x \sim \mathcal{N}(0, 1)$, statistically independent of z and y. These simulate irrelevant or faulty sensors.

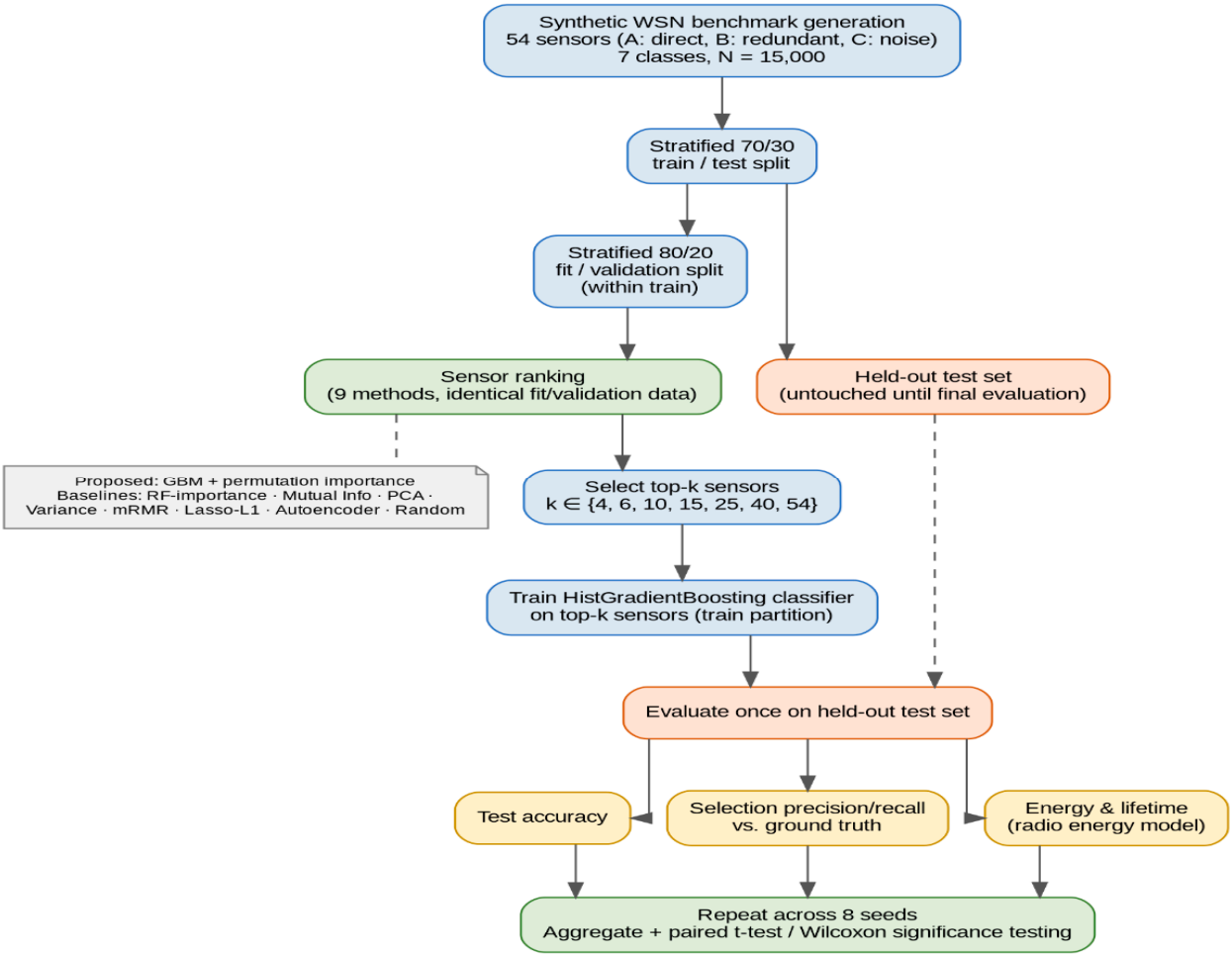

Figure 1. Overview of the experimental pipeline: synthetic benchmark generation, stratified splitting, sensor ranking (9 methods), top-*k* selection, classification, and multi-seed evaluation. Blue boxes denote data-preparation and model-training steps, green boxes denote sensor ranking and multi-seed aggregation, orange boxes denote the held-out test partition and its single final evaluation, and yellow boxes denote the reported evaluation metrics; the dashed lines indicate the held-out test-data path and the list of ranking methods.

Ground truth is defined as informative = Group A ∪ Group B (36 of 54 sensors); Group C (18 sensors) is defined as ground-truth non-informative. This lets us report, for any selection method, both downstream accuracy and selection precision/recall against a known answer that is not possible with unlabeled benchmarks. N = 15,000 samples were generated per experiment (a seed-42 generator instance was shared across all method comparisons, so every method sees identical data).

A single validation run confirms that the benchmark's difficulty is realistic rather than trivial: a Random Forest trained on all 54 sensors reaches 85.6% test accuracy; trained on the 18 noise sensors only, it falls to 34.3% (chance level, matching the 35.2% majority-class baseline); and when trained on the 6 direct sensors alone, it reaches 84.8%, confirming that Group A carries nearly all classification-relevant signals and that Groups B and C are, by design, largely redundant or irrelevant to the task, as shown in Figure 2.

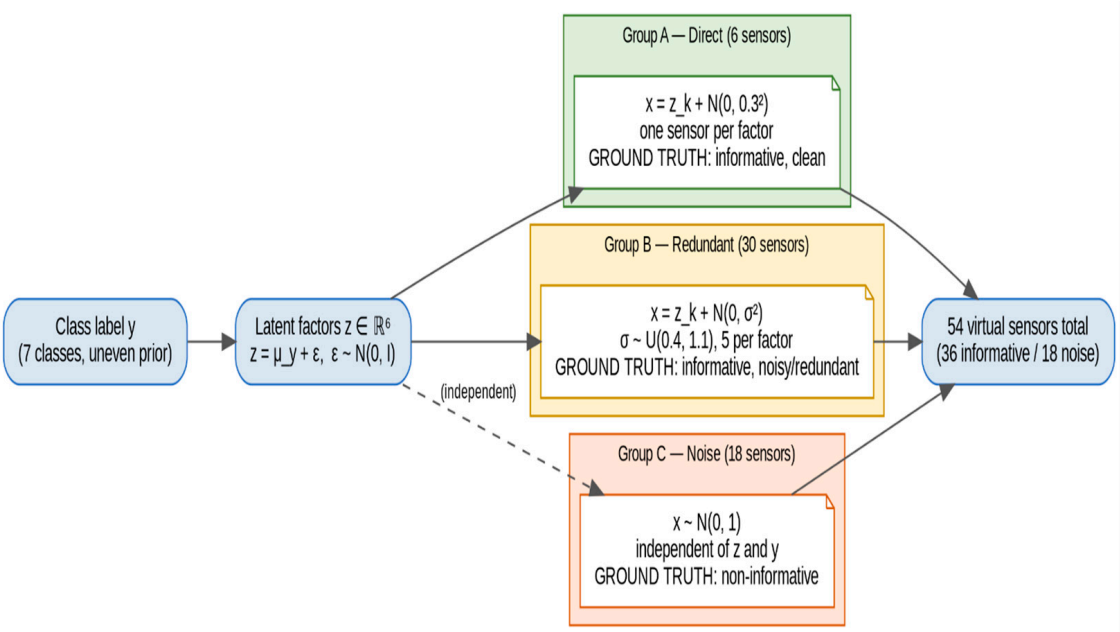

Figure 2. Generative structure of the synthetic sensor benchmark: six latent factors produce three ground-truth sensor groups (A: direct, B: redundant, C: noise). Green, yellow, and orange panels denote sensor Groups A (direct), B (redundant), and C (noise), respectively; blue boxes denote the class label, the latent factors, and the resulting set of 54 virtual sensors; the dashed arrow indicates that Group C is generated independently of the latent factors.

3.3. Network Topology and Energy Model

Each of the 54 sensors is treated as a node in a physical deployment: node positions are drawn uniformly at random in a 100 m × 100 m field (seed-fixed, shared across all methods and sensor counts so that energy differences reflect which sensors are chosen, not topology variation), with a fixed sink at the field's edge midpoint, as shown in Figure 3.

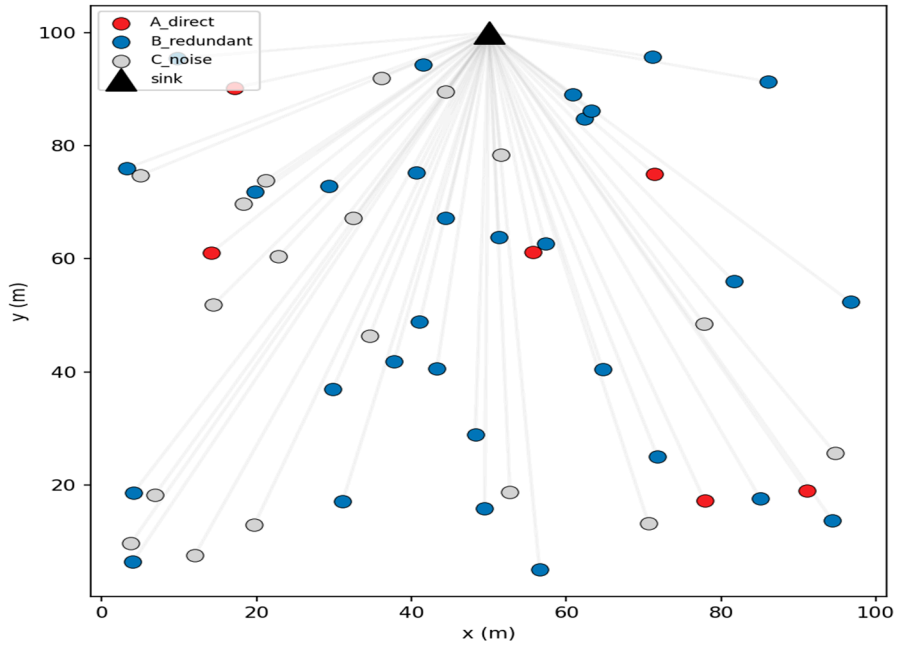

Figure 3. Simulated WSN topology.

Energy consumption is modelled using the standard first-order radio communication model [12]. Transmitting an l-bit packet over distance d costs

$$E_{tx}(l, d) = \begin{cases} l \cdot E_{elec} + l \cdot \varepsilon_{fs} \cdot d^2, & \text{if } d < d_0 \\ l \cdot E_{elec} + l \cdot \varepsilon_{mp} \cdot d^4, & \text{if } d \geq d_0 \end{cases} \quad (1)$$

with crossover distance $d_0 = \sqrt{(\varepsilon_{fs}/\varepsilon_{mp})}$ and receiving costs $E_{rx}(l) = l \cdot E_{elec}$. We use the standard literature constants $E_{elec} = 50$ nJ/bit, $\varepsilon_{fs} = 10$ pJ/bit/m^2, and $\varepsilon_{mp} = 0.0013$ pJ/bit/m^4, a packet size of 4000 bits, and an initial per-node energy budget of 0.5 J (typical for a small sensor mote). For a given selection of k active sensors, per-round network energy is calculated as the sum of the direct transmission energy from each active sensor to the sink and the corresponding reception energy at the sink. Network lifetime is defined as the number of transmission rounds until the first active sensor exhausts its battery, assuming that each active sensor transmits one packet per round. Because energy depends on each selected sensor's specific distance to the sink, two methods selecting the same count of sensors are not guaranteed identical energy cost—a physical nuance the naive count-only LEF ratio cannot capture, though we report LEF (=54/k) alongside the modeled energy for continuity with prior work [5,6].

3.4. Proposed Sensor-Selection Method

We rank sensors using scikit-learn's HistGradientBoostingClassifier (scikit-learn version 1.8.0, Python 3.12), a histogram-based gradient-boosting implementation, trained on the fit partition, with sensors ranked according to permutation importance measured on a held-out validation partition. We use permutation importance rather than impurity-based (gain) importance specifically because impurity-based measures are known to be biased toward high-cardinality/high-variance features [13], which causes exactly this failure mode in a variance-based baseline. Permutation importance is computed with 3 repeats on a subsample of up to 1500–3000 validation points for computational tractability.

Rather than fix an arbitrary sensor count, we evaluate the full sweep $k \in \{4, 6, 10, 15, 25, 40, 54\}$ and define the recommended operating point as the smallest k in the sweep whose mean test accuracy is at least 98% of the full-54-sensor accuracy—a simple, reproducible knee-point rule rather than a post-hoc visual read of the curve. Applied to our results (Section 4), this rule selects $k = 10$.

3.5. Baseline Methods

All baseline methods were independently re-implemented and evaluated under conditions identical to those used for the proposed approach, including the same data partitions, downstream classifier, and sensor-count sweep; no performance values were taken directly from previously published implementations.

- Random Forest importance: (200 trees; impurity-based ranking) the classical embedded tree baseline.
- Mutual Information: (mutual-info-classif) a model-free filter method.
- PCA loadings: sensors ranked by cumulative |loading|-weighted contribution to the components explaining 95% of standardized variance; unsupervised.
- Variance thresholding: sensors ranked by raw (unstandardized) variance; unsupervised, included specifically to test susceptibility to the noise-inflates-variance failure mode.
- mRMR: (minimum-Redundancy–Maximum-Relevance [14]) incremental selection maximizing relevance (mutual information with the label) minus mean redundancy (mutual information with already-selected sensors); this is the same algorithmic family used by the closest directly comparable prior work [6], computed here on a 3000-sample subsample for tractability.

- Lasso-L1: L1-regularized multinomial logistic regression (saga solver, C = 0.1); sensors ranked by mean |coefficient| across classes. An embedded, linear-family alternative to tree-based importance.
- Autoencoder (reconstruction-based): an undercomplete MLP autoencoder (12-unit bottleneck, ReLU, Adam, early stopping) trained to reconstruct its own standardized input; sensors ranked by the L2 norm of their encoder input weights. Included as an analog to the Stacked/Sparse-Autoencoder sensor-selection approaches common in this literature, to test whether a purely reconstruction-driven (label-blind) objective aligns with classification-relevant importance.
- Random: a seeded random permutation of sensor order; sanity-check floor.

3.6. Evaluation Protocol

For each of 8 random seeds, data are split 70/30 into train/test (stratified), and the training partition is further split 80/20 into fit/validation (stratified) for ranking methods that require a held-out set. Ranking models are fitted using the fit partition, while the validation partition is used for validation-based importance estimation and operating-point selection; after the sensor count is fixed, the final classifier is trained on the complete training partition restricted to the selected top-k sensors and evaluated once on the untouched test partition. This yields, per seed: test accuracy, selection precision/recall against ground truth, modeled energy per round, modeled network lifetime, and LEF, for every (method, k) combination—9 methods $\times$ 7 sensor counts $\times$ 8 seeds = 504 runs in the main experiment. Statistical comparisons between GBM-Permutation and each baseline at the fixed operating point are performed using paired t-tests and Wilcoxon signed-rank tests across the eight matched data partitions, with each pair sharing the same train/test split within a seed.

3.7. Redundancy–Sensitivity Design

To test whether conclusions depend on how "redundant" Group B sensors are, we parameterize their noise range and re-run the comparison (GBM-Permutation method plus five representative baselines: Random Forest, mRMR, Lasso, Variance, Random) under three regimes, holding all other generative parameters fixed: high redundancy ($\sigma \sim$ Uniform(0.05, 0.35), near-duplicate sensors), medium redundancy ($\sigma \sim$ Uniform(0.4, 1.1), the main-experiment setting), and low redundancy ($\sigma \sim$ Uniform(0.9, 1.6), noisy/weakly-correlated sensors). Each regime is run across 3 seeds and a reduced sweep $k \in \{6, 10, 25, 54\}$, given the additional computational cost of a third experimental axis.

3.8. Real-Data Generalization Check

As a supplementary check that the findings are not specific to the synthetic generative design, we repeat the method comparison on two established real-world tabular benchmarks obtained from the datasets bundled with scikit-learn: the Breast Cancer Wisconsin diagnostic dataset (569 samples, 30 features, binary classification) and the Digits dataset (1797 samples, 64 features, 10-class classification), both accessed via scikit-learn's bundled datasets, each swept over its own dataset-appropriate sensor counts (Section 4.6) rather than the {4, 6, 10, 15, 25, 40, 54} sweep used for the synthetic benchmark. Neither benchmark has engineered sensor-redundancy structure, so this check specifically probes whether the proposed method's advantage is conditional on the kind of redundancy the synthetic benchmark was designed to contain.

3.9. Implementation and Reproducibility

All experiments use Python 3.12, scikit-learn 1.8.0, NumPy 2.4.4, and pandas 3.0.2. All random seeds, generative parameters, and hyperparameters are fixed and reported above; code is organized as a set of standalone, seed-parameterized scripts (data generation, energy

model, selection methods, and experiment runners) to support independent re-execution. For reproducibility, all experiments use scikit-learn's HistGradientBoostingClassifier rather than the XGBoost package; the reported results therefore refer specifically to the scikit-learn histogram-based gradient-boosting implementation, while direct validation using XGBoost remains an important avenue for future work. The synthetic benchmark and network simulation parameters are summarized in Table 2, and the model hyperparameters and evaluation protocol are summarized in Table 3.

Table 2. Synthetic benchmark and network simulation parameters.

Parameter	Value	Notes
	Class structure	
Number of classes	7	Uneven prior: 0.36/0.21/0.18/0.09/0.07/0.05/0.04
Latent factors (K)	6	$z = \mu_y + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$
Samples (N)	15,000	Shared generator instance (seed 42) across all methods
	Sensor groups (54 total)	
Group A—direct	6 sensors	$x = z_k + \mathcal{N}(0, 0.3^2)$; ground truth: informative, clean
Group B—redundant (main)	30 sensors (5/factor)	$x = z_k + \mathcal{N}(0, \sigma^2)$, $\sigma \sim U(0.4, 1.1)$
Group B—high-redundancy regime	same 30 sensors	$\sigma \sim U(0.05, 0.35)$
Group B—low-redundancy regime	same 30 sensors	$\sigma \sim U(0.9, 1.6)$
Group C—noise	18 sensors	$x \sim \mathcal{N}(0,1)$, independent of z and y
	Network topology	
Field size	100 m × 100 m	Node positions ~ Uniform, seed-fixed
Number of nodes	54	One per virtual sensor
Sink position	Field edge, x-midpoint	Fixed across all methods/seeds
	Radio energy model (first-order)	
E_{elec}	50 nJ/bit	Electronics energy, TX and RX
ε_{fs}	10 pJ/bit/m^2	Free-space model (d < d0)
ε_{mp}	0.0013 pJ/bit/m^4	Multi-path model (d ≥ d0)
Packet size	4000 bits	
Initial node energy	0.5 J	Typical small sensor mote

Table 3. Model hyperparameters and evaluation protocol.

Parameter	Value	Notes
	Proposed method	
Classifier	HistGradientBoostingClassifier	scikit-learn 1.8.0
max_iter	60	
Ranking method	Permutation importance	3 repeats, subsample ≤ 1500–3000 validation points
	Baseline hyperparameters	
Random Forest	200 trees	Impurity-based importance
Lasso-L1	solver = saga, C = 0.1	Mean |coef.| across classes
Autoencoder	1 hidden layer, 12 units	ReLU, Adam, early stopping
mRMR	8 bins, 3000-sample subsample	Incremental relevance − mean redundancy
PCA	95% variance threshold	|Loading|-weighted ranking

Table 3. *Cont.*

Parameter	Value	Notes
	Data splitting	
Train/test split	70%/30%	Stratified
Fit/validation split	80%/20%	Stratified, within training partition
	Sweep and repetition	
Sensor counts (k)	{4, 6, 10, 15, 25, 40, 54}	Main experiment
Sensor counts (k), sensitivity analysis	{6, 10, 25, 54}	Reduced sweep
Seeds, main experiment	8	Seeds 0–7
Seeds, sensitivity analysis	3 per regime	3 regimes × 3 seeds
Recommended operating point rule	smallest k with ≥ 98% of full accuracy	Applied result: k = 10
	Statistical testing	
Paired significance tests	Paired t-test; Wilcoxon signed-rank	Paired on seed, at k = 10

4. Results

This section evaluates the GBM-Permutation approach through a comprehensive comparison with eight baseline sensor-selection methods. The results include accuracy, sensor recovery, statistical significance, robustness to redundancy, energy efficiency, and validation on real-world datasets.

4.1. Accuracy vs. Number of Selected Sensors

Figure 4 plots test accuracy against the number of selected sensors for all nine methods, averaged over eight seeds.

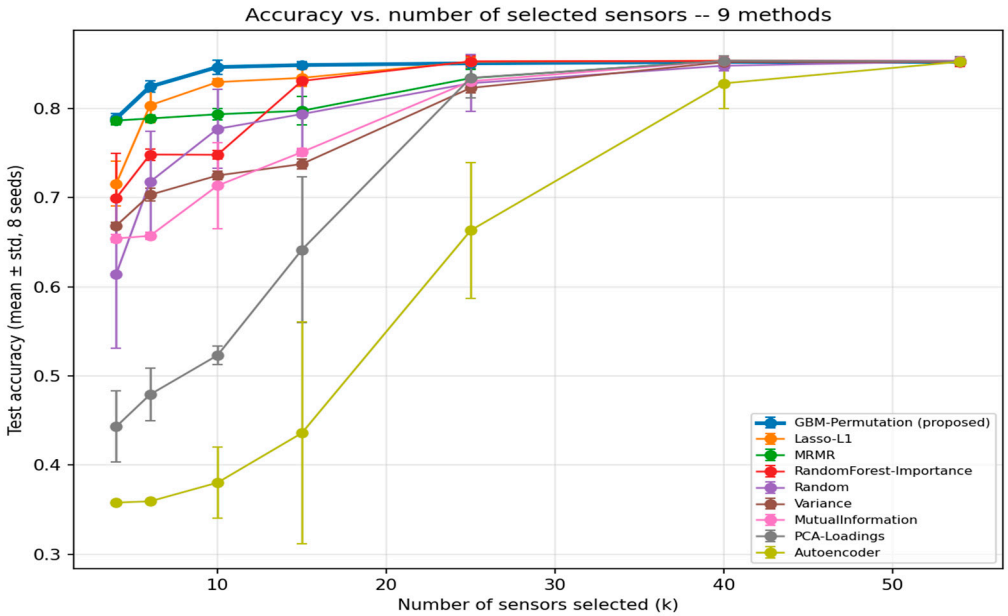

Figure 4. Accuracy vs. number of selected sensors of nine methods.

Figure 4 and Table 4 compare the classification accuracy of nine sensor-selection methods as the number of selected sensors increases. Results are averaged over eight independent runs, with error bars representing one standard deviation. The GBM-Permutation method consistently achieves the highest accuracy in the low-sensor region, obtaining 78.88% at k = 4 and 84.64% at k = 10, outperforming most competing approaches. This indi-

cates that the GBM-Permutation ranking strategy effectively identifies the most informative sensors using a limited subset. As k increases, the performance gap gradually narrows, and the methods converge to approximately 85.2–85.4% at $k = 40$, suggesting that retaining a large proportion of the sensors provides only a limited additional classification benefit under the present benchmark design. The relatively small error bars also demonstrate the stability of the GBM-Permutation method across different runs. Overall, the results indicate that the GBM-Permutation approach provides a favorable trade-off between classification accuracy and sensor reduction, with potential benefits for energy-aware wireless sensor network operation.

Table 4. Test accuracy vs. number of selected sensors (mean, 8 seeds).

Method	$k = 4$	$k = 6$	$k = 10$	$k = 15$	$k = 25$	$k = 40$	$k = 54$
GBM-Permutation	0.7888	0.8244	0.8464	0.8487	0.8506	0.8522	0.8523
Lasso-L1	0.7156	0.8035	0.8296	0.8343	0.8524	0.8535	0.8532
MRMR	0.7865	0.7887	0.7935	0.7974	0.834	0.8536	0.8531
Random	0.6143	0.7176	0.7771	0.7937	0.8283	0.848	0.853
RandomForest-Importance	0.6996	0.7483	0.748	0.8309	0.8529	0.8533	0.8519
Variance	0.6685	0.7031	0.7248	0.7377	0.8231	0.852	0.8522
MutualInformation	0.6542	0.6571	0.7134	0.751	0.8306	0.8536	0.8536
PCA-Loadings	0.4431	0.4791	0.5231	0.6415	0.8343	0.8535	0.8529
Autoencoder	0.3579	0.3592	0.3802	0.4359	0.6633	0.8282	0.8524

4.2. Recommended Operating Point, Energy, and Lifetime

Based on the validation-based operating-point criterion defined in Section 3.4, the recommended operating point is $k = 10$, representing the smallest sensor subset that achieves at least 98% of the full-sensor validation accuracy. As shown in Figure 5, selecting 10 sensors yields 84.6% test accuracy, corresponding to 99.3% of the maximum achievable accuracy (85.2% with 54 sensors). Beyond this point, increasing the number of selected sensors provides only marginal accuracy improvements while substantially increasing energy consumption. According to the network topology and radio energy model, this operating point reduces the per-round network energy consumption by 82.6%. The modeled first-node-dies network lifetime increases from 615 rounds at $k = 54$ to 883 rounds at $k = 10$, a 1.4-fold increase; the LEF of 5.4 (=54/10) is a separate, sensor-count-based ratio and should not be read as the modeled lifetime ratio, which depends on the physical energy model rather than sensor count alone. The resulting accuracy–energy trade-off indicates that the GBM-Permutation method can retain most of the predictive performance while substantially reducing modeled communication energy, highlighting its potential for energy-constrained WSN operation.

Because Section 3.4 fixes the sensor-count sweep across methods for comparability, a natural question is whether the reported energy and lifetime figures follow from the sensor count alone (LEF = 54/k is identical for every method at a given k) rather than from which specific sensors are chosen. They do not: across the nine methods at $k = 10$, modeled per-round energy varies by 10.9% of the mean, and first-node-dies lifetime varies by 34.8% of the mean (43.7% between the best method, RandomForest-Importance at 933.8 rounds, and the worst, PCA-Loadings at 650.1 rounds), purely from the selected sensors' differing distances to the sink. LEF cannot distinguish these cases; the physical energy model can. The reported lifetime should be interpreted as a model-based first-node-dies estimate under a single-hop, non-aggregating deployment with a fixed active set between re-ranking cycles; practical mechanisms such as active-set rotation, multi-hop routing, and in-network aggregation could produce different lifetime values in a real deployment.

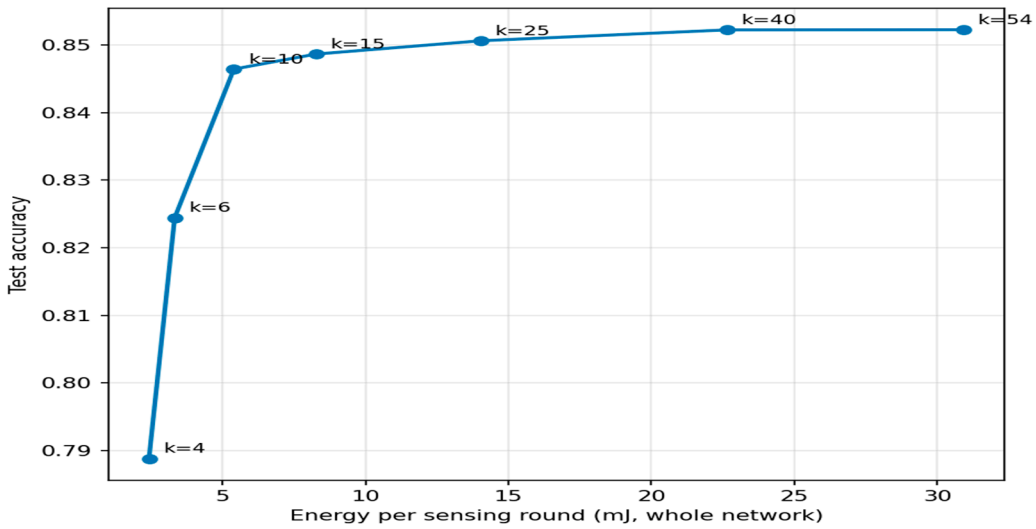

Figure 5. Accuracy–energy trade-off.

4.3. Selection Quality: Recovery of Ground-Truth Informative Sensors

Because the benchmark provides an explicit ground-truth clean sensor set, the selection quality of each method can be evaluated by measuring how many of the six Group A sensors are recovered within the selected subset, as shown in Figure 6 and Table 5.

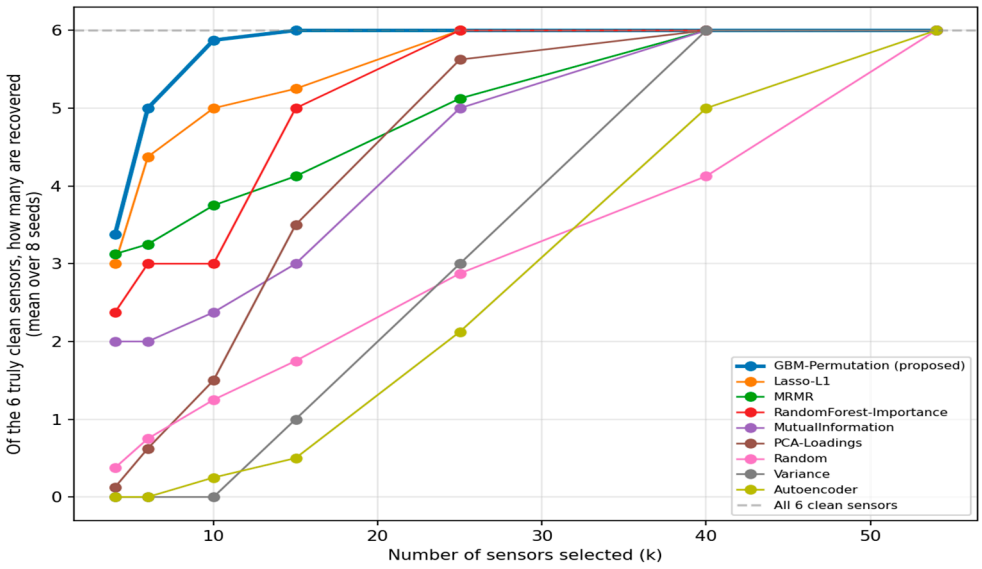

Figure 6. Recovery of cleanest ground-truth sensors.

Table 5. Recovery of the 6 ground-truth clean sensors (mean of 6, 8 seeds).

Method	k = 4	k = 6	k = 10	k = 15	k = 25	k = 40	k = 54
GBM-Permutation	3.38	5	5.88	6	6	6	6
Lasso-L1	3	4.38	5	5.25	6	6	6
MRMR	3.12	3.25	3.75	4.12	5.12	6	6
Random Forest-Importance	2.38	3	3	5	6	6	6
Mutual Information	2	2	2.38	3	5	6	6
PCA-Loadings	0.12	0.62	1.5	3.5	5.62	6	6
Random	0.38	0.75	1.25	1.75	2.88	4.12	6
Autoencoder	0	0	0.25	0.5	2.12	5	6
Variance	0	0	0	1	3	6	6

Section 3.2 defines the ground-truth informative set as Group A ∪ B (36 of 54 sensors), and precision/recall against this definition are reported here directly rather than only the finer Group-A-specific metric above. At $k = 10$, seven of the nine methods, including GBM-Permutation, achieve a precision of 1.000 and a recall of 0.278 against the broader 36-sensor informative set; the two exceptions are Random (precision 0.725) and Autoencoder (precision 0.025), both of which include noise sensors among their top-10 selections. This reflects a ceiling effect rather than a lack of differentiation: because $k = 10$ is smaller than the 36-sensor informative set, recall against this coarse definition is mechanically capped at $k/36 = 0.278$ regardless of method quality, and most methods already achieve perfect precision, so the coarse metric saturates immediately at small k. This is precisely why the finer six-sensor recovery metric above is reported as the primary selection-quality measure in the low-k regime; the two metrics are complementary, not substitutes for one another.

The GBM-Permutation method achieves the highest recovery of the six ground-truth clean sensors in the low-sensor regime, identifying an average of 5.88 of the six sensors at $k = 10$ and all six at $k = 15$. In contrast, Lasso-L1 recovers five sensors at $k = 10$, while MRMR, Random Forest Importance, and Mutual Information recover considerably fewer. Variance-based ranking performs particularly poorly, recovering none of the six ground-truth clean sensors at $k = 10$, indicating that high feature variance does not necessarily correspond to high classification relevance. These findings explain the superior classification accuracy by prioritizing the most informative sensors early; GBM-Permutation method achieves high predictive performance using substantially fewer sensors, thereby supporting efficient and energy-aware wireless sensor network operation.

4.4. Statistical Significance

To assess whether the observed performance differences were consistent across data partitions, paired statistical tests were conducted at the fixed operating point using eight matched experimental runs. As summarized in Table 6, the GBM-Permutation method significantly outperforms all competing approaches according to both the paired t-test and the Wilcoxon signed-rank test. The smallest improvement is observed over Lasso-L1, with a mean accuracy gain of 1.69 percentage points ($p = 0.00046$), while substantially larger gains are achieved over MRMR, Random Forest Importance, Mutual Information, PCA Loadings, and Autoencoder. All Wilcoxon p-values are 0.00781, confirming consistent superiority under a non-parametric test. These results indicate that the performance advantage of GBM-Permutation is consistent across the evaluated data partitions rather than being attributable solely to a favorable partition or initialization. Collectively, these results provide evidence of the robustness of the GBM-Permutation sensor-selection strategy under the evaluated experimental conditions.

Table 6. Paired significance at $k = 10$ (N = 8 seeds).

Baseline	Mean Acc.	Mean Diff.	t-Test p	Wilcoxon p
Lasso-L1	0.8296	+0.0169	0.00046	0.00781
MRMR	0.7935	+0.0529	<0.0001	0.00781
Random	0.7771	+0.0693	0.00522	0.00781
Random Forest-Importance	0.748	+0.0984	<0.0001	0.00781
Variance	0.7248	+0.1216	<0.0001	0.00781
Mutual Information	0.7134	+0.1330	0.00007	0.00781
PCA-Loadings	0.5231	+0.3233	<0.0001	0.00781
Autoencoder	0.3802	+0.4662	<0.0001	0.00781

4.5. Sensitivity to Redundancy Severity

Repeating the comparison under high-, medium-, and low-redundancy regimes shows that the performance advantage is not uniform across redundancy conditions (Figures 7 and 8).

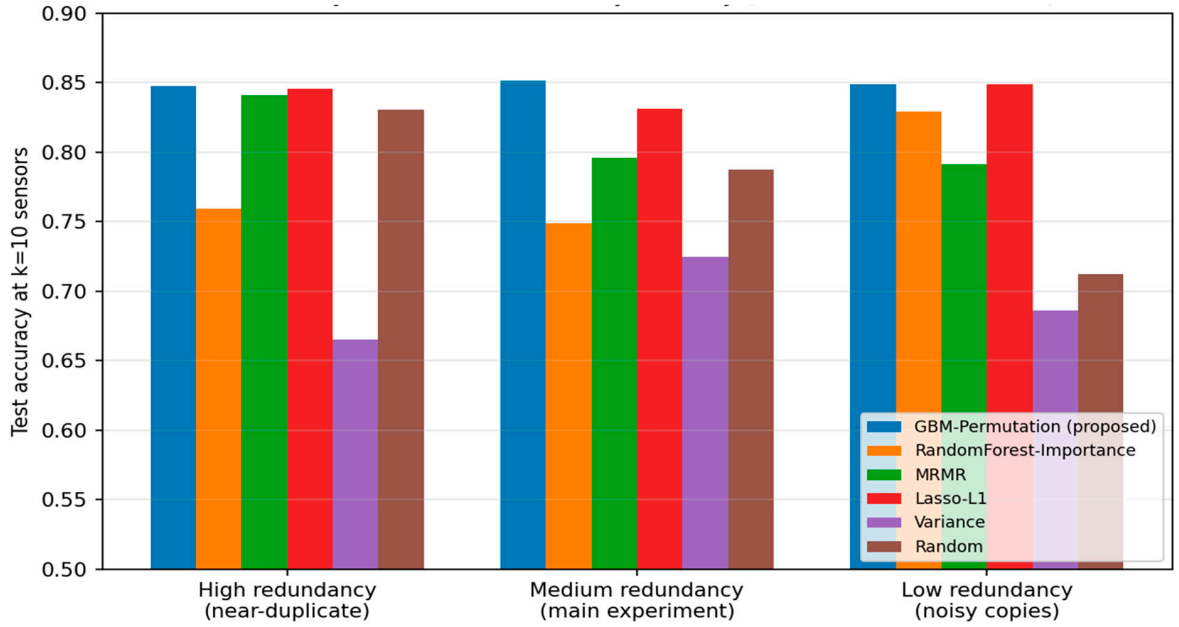

Figure 7. Sensitivity to sensor-redundancy severity (k = 10, mean of 3 seeds).

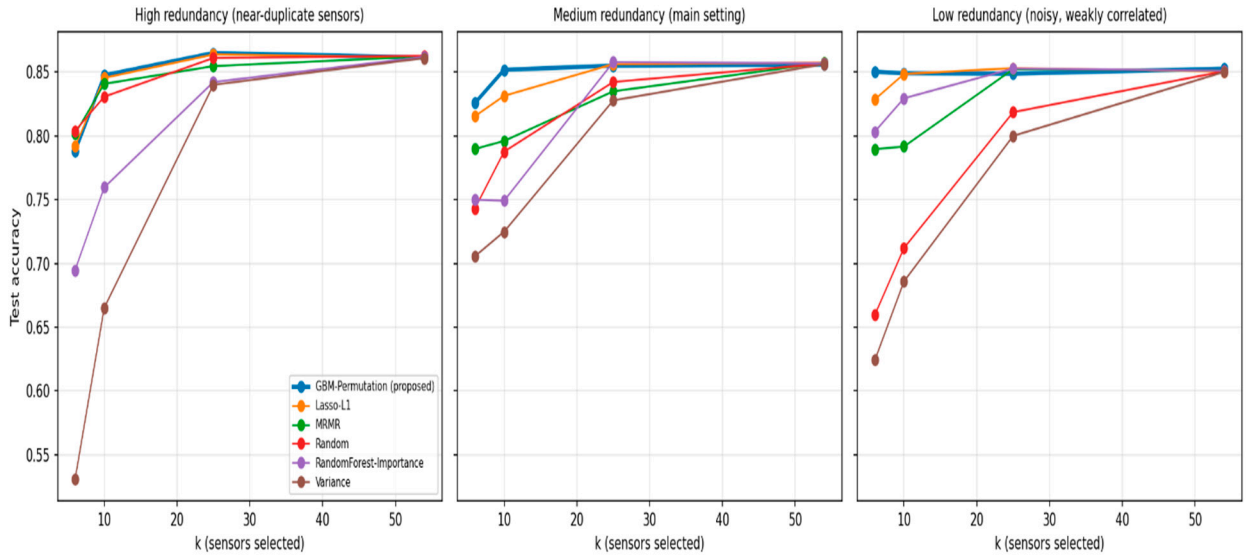

Figure 8. Test accuracy versus the number of selected sensors across redundancy regimes.

In deployment terms, the high-redundancy regime corresponds to densely instrumented settings such as multiple co-located sensors on a single industrial asset, where the simpler and cheaper Lasso-L1, mRMR, or even random selection are adequate since they are not significantly different from GBM-Permutation method there. The medium-redundancy regime corresponds to typical heterogeneous WSN deployments with partial, imperfect sensor correlation, where GBM-Permutation method's advantage is largest and justifies its additional computational cost (Section 4.8). The low-redundancy regime (sparse, weakly correlated sensors) continues to favor the GBM-Permutation method and Lasso-L1 significantly over simpler heuristics.

The robustness of the GBM-Permutation method was further evaluated under three sensor redundancy regimes: high, medium, and low redundancy (Figures 7 and 8 and Table 7). The GBM-Permutation method consistently maintains high classification accuracy across all scenarios, achieving 84.76%, 85.19%, and 84.92% in the high-, medium- and low-redundancy settings, respectively. Statistical analysis shows that its performance is significantly better than Random Forest Importance and Variance across all regimes ($p < 0.05$). Compared with Lasso-L1, GBM-Permutation exhibits a significant advantage only under the medium-redundancy condition, while the differences are not statistically significant in the high- and low-redundancy scenarios. These results suggest that the two methods perform similarly under very high and very low redundancy, whereas GBM-Permutation provides a clearer advantage under the intermediate redundancy regime represented by the main synthetic setting. Overall, these results indicate that the proposed approach maintains competitive performance across the evaluated redundancy levels, while its relative advantage depends on the degree of sensor redundancy.

Table 7. Accuracy by redundancy regime at $k = 10$.

Method	High-Redund.	p (vs. Prop.)	Medium-Redund.	p (vs. Prop.)	Low-Redund.	p (vs. Prop.)
GBM-Permutation	0.8476	—	0.8519	—	0.8492	—
Lasso-L1	0.8456	0.875 (ns)	0.8315	0.028 *	0.8487	0.626 (ns)
MRMR	0.8409	0.603 (ns)	0.7961	0.004 *	0.7917	0.001 *
Random Forest-Importance	0.7596	0.016 *	0.7492	<0.0001 *	0.8296	0.024 *
Random	0.8308	0.503 (ns)	0.7877	0.115 (ns)	0.7121	0.012 *
Variance	0.6651	0.003 *	0.7249	0.0002 *	0.6859	0.001 *

* $p < 0.05$ vs. GBM-Permutation; ns, not significant; 3 seeds/regime—indicative, not as powered as Section 4.4.

4.6. Generalization to Real-World and Non-Simulated Data

The following two subsections evaluate the proposed method's generalizability beyond the synthetic benchmark: first on two generic (non-sensor) tabular datasets (Section 4.6.1), then on a genuine sensor-derived dataset (Section 4.6.2).

4.6.1. Generalization to Real (Non-Simulated) Datasets

To assess whether the observed behavior extends beyond the synthetic WSN benchmark, experiments were conducted on the Breast Cancer Wisconsin and Digits datasets, which contain real-world tabular features but no explicitly engineered sensor-redundancy structure. Because these datasets have far fewer total features than the 54-sensor synthetic benchmark, each was evaluated on its own dataset-appropriate sensor-count sweep (Breast Cancer: $k \in \{3, 5, 10, 15, 30\}$; Digits: $k \in \{5, 10, 20, 32, 64\}$) rather than the main sweep of Section 3.4, so that the largest swept value always equals the full feature count of that dataset. As shown in Figure 9, the GBM-Permutation method remains competitive across both datasets, although its superiority is less pronounced than in the synthetic benchmark. On the Breast Cancer dataset, all methods achieve comparable performances, with accuracies ranging from 93.6% to 95.1% at k = 10, indicating limited differences in feature ranking effectiveness. On the Digits dataset, the GBM-Permutation method demonstrates an advantage when only a small number of features are selected, achieving 69.1% accuracy at $k = 5$, while maintaining competitive performance as additional features are included. These findings are consistent with the redundancy analysis and suggest that the advantage of GBM-Permutation is more pronounced when meaningful feature redundancy is present, while the method remains competitive on the evaluated real-world tabular datasets.

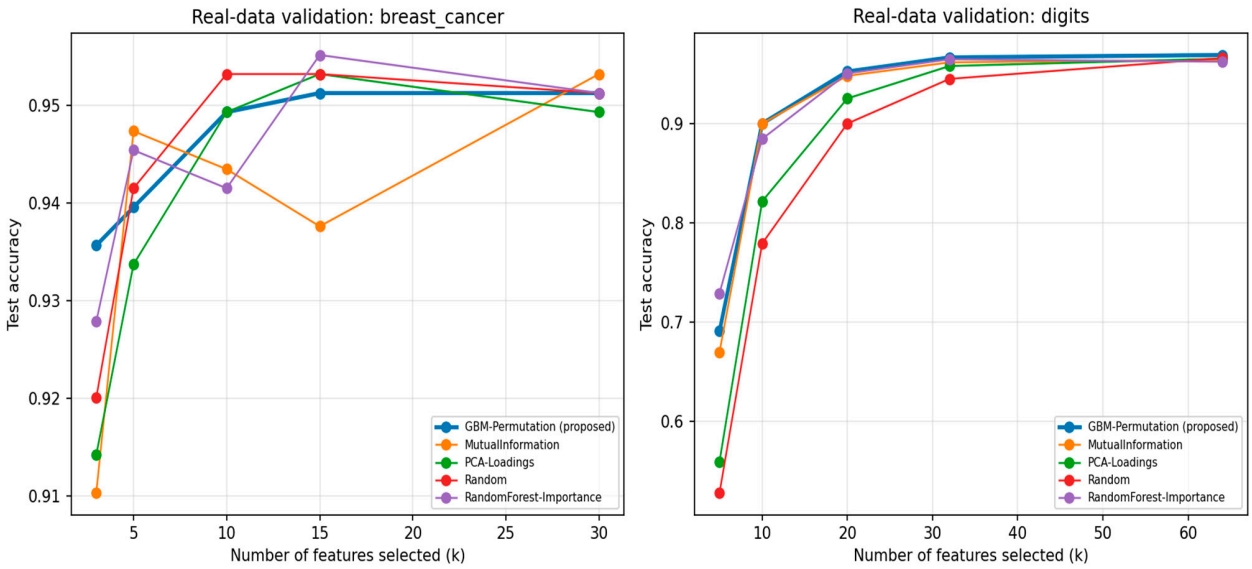

Figure 9. Real-data validation: test accuracy vs. number of selected features on breast-cancer (**left**) and digits (**right**) datasets, by selection method.

4.6.2. Validation on Real Sensor-Derived Data (Gas Sensor Array Drift)

As a further validation using real sensor-derived data, the method comparison was repeated on the Gas Sensor Array Drift dataset (13,910 samples, 128 sensor-derived features from 16 physical gas sensors × eight derived statistics each, six gas classes), the closest directly comparable prior work, evaluated over $k \in \{10, 20, 40, 60, 90, 128\}$ across five seeds. The results are shown in Figure 10 and summarized in Table 8 curacy and significance.

The dataset's 128 features (16 sensors × eight correlated derived statistics) structurally resemble a high-redundancy sensor deployment. Consistent with that interpretation, the GBM-Permutation method is not significantly different from Random ($p = 0.075$) or Lasso-L1 ($p = 0.207$) here—closely matching the high-redundancy regime result in Section 4.5 (Table 7), where the same two comparisons were also non-significant. The observed real-data result is consistent with the high-redundancy simulation regime, providing empirical support for the redundancy–sensitivity finding while also showing that the advantage of GBM-Permutation is not universal.

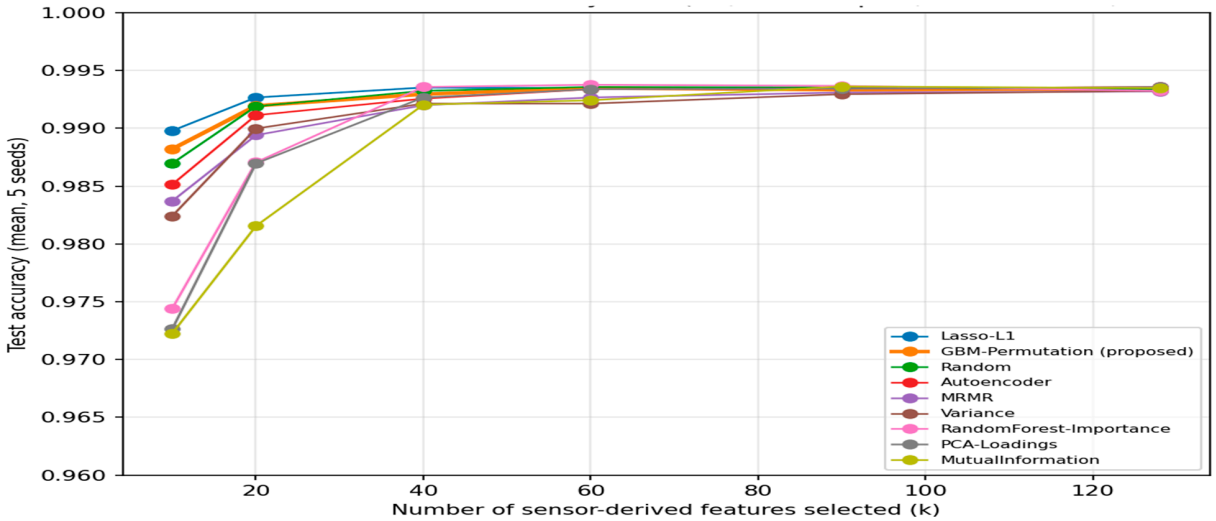

Figure 10. Real-data validation on the Gas Sensor Array Drift dataset (13,910 samples, 128 features, 6 classes).

Table 8. Accuracy and significance at $k = 10$, Gas Sensor Array Drift (mean, 5 seeds).

Method	Accuracy	*p*
Lasso-L1	98.98%	0.207 (ns)
GBM-Permutation	98.82%	—
Random	98.70%	0.075 (ns)
Autoencoder	98.51%	0.167 (ns)
MRMR	98.37%	0.025 *
Variance	98.24%	0.008 *
RandomForest-Importance	97.44%	0.003 *
PCA-Loadings	97.26%	0.009 *
MutualInformation	97.23%	<0.0001 *

* $p < 0.05$ vs. GBM-Permutation; ns, not significant.

4.7. Ablation Study

The ablation examines two design choices: ranking sensors using permutation importance rather than an alternative importance measure, and selecting the operating point adaptively rather than fixing the number of retained sensors. The following two ablations isolate the individual contribution of each.

4.7.1. Ranking-Mechanism Ablation

Two additional ranking mechanisms were implemented using the identical fitted HistGradientBoosting model used by the GBM-Permutation method, isolating the effect of the importance-extraction method from the choice of classifier family (unlike comparisons against Random Forest or Lasso, which necessarily change both at once). "Gain-based" sums split gain across every tree in the ensemble; "Train-set permutation" computes permutation importance on the training data itself rather than the held-out validation split used by the GBM-Permutation method. The results are summarized in Table 9.

Table 9. Ranking-mechanism ablation at $k = 10$ (same classifier, 8 seeds).

Ranking Mechanism	Accuracy	*p*
Permutation (held-out validation)	84.64%	—
Gain-based (same fitted model, no permutation)	84.33%	0.539 (ns)
Permutation importance, training set (no held-out split)	84.94%	0.291 (ns)

p-values are relative to permutation importance computed on the held-out validation split (the GBM-Permutation default); ns, not significant.

These results indicate that the performance advantage over the cross-family baselines is primarily associated with the gradient-boosting model family rather than with permutation importance providing a statistically significant advantage over gain-based importance within that family. The held-out validation split is nonetheless retained as the default design because computing importance on the training set risks an optimistic bias in general—the model can appear to depend on a feature simply because it fit that feature's noise, an effect that held-out evaluation is specifically designed to avoid—even though the two did not differ significantly in this particular benchmark.

4.7.2. Adaptive vs. Fixed Sensor Count

Using the sensor-count sweep reported in Table 4, Table 10 quantifies the energy savings achieved by the adaptive operating-point rule relative to two representative fixed sensor counts.

Table 10. Adaptive operating point vs. fixed sensor counts.

k	Accuracy	% of Full Accuracy	Energy/Round	LEF
10 (adaptive)	84.6%	99.3%	0.0054 J	5.4
25 (fixed)	85.1%	99.8%	0.0140 J	2.16
40 (fixed)	85.2%	100.0%	0.0226 J	1.35

Relative to the adaptive $k = 10$ operating point, using $k = 25$ increases accuracy by 0.5 percentage points at 2.6 times the modeled energy cost, while $k = 40$ increases accuracy by 0.6 percentage points at 4.2 times the modeled energy cost, illustrating the energy–accuracy trade-off associated with larger fixed sensor counts.

4.8. Computational Cost and Deployment Feasibility

Wall-clock time and peak memory (via Python's tracemalloc, five repeats) were measured for the ranking step of all nine methods on the full training data (n = 15,000, d = 54). The results are shown in Figure 11 and summarized in Table 11.

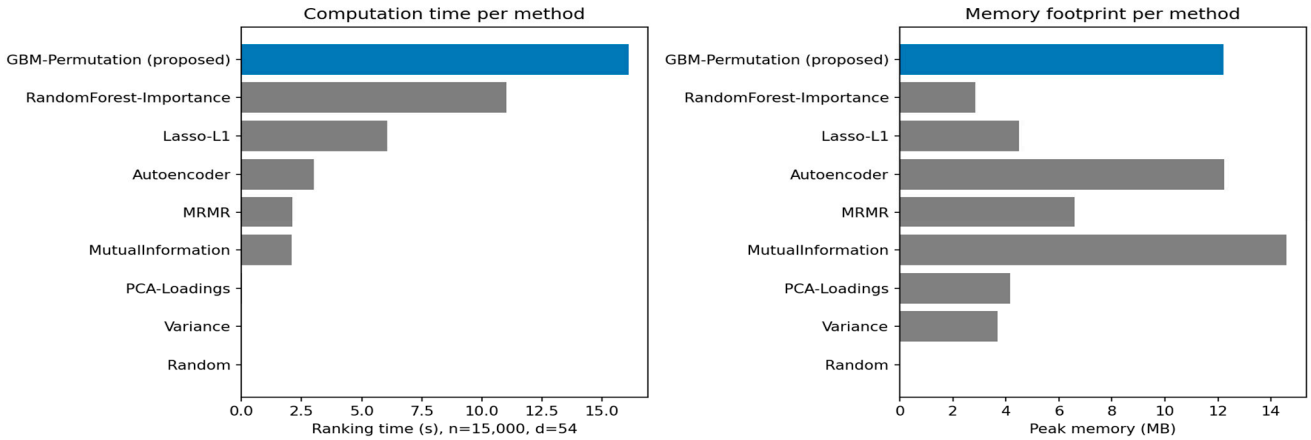

Figure 11. Measured ranking time and peak memory per method. The proposed GBM-Permutation method is shown in blue and the baseline methods in grey; the bars for PCA-Loadings, Variance, and Random are close to zero at this scale.

Table 11. Computational cost and theoretical complexity per method.

Method	Time (s)	Memory (MB)	Complexity
GBM-Permutation	16.09	12.22	$O(T \cdot n \cdot d) + O(R \cdot d \cdot m)$
RandomForest-Importance	11.01	2.85	$O(T \cdot n \cdot d \cdot \log n)$
Lasso-L1	6.07	4.51	$O(I \cdot n \cdot d)$
Autoencoder	3.01	12.24	$O(E \cdot n \cdot d \cdot h)$
MRMR	2.11	6.58	$O(d^2 \cdot n_{sub})$
MutualInformation	2.09	14.59	$O(d \cdot n \log n)$
PCA-Loadings	0.011	4.16	$O(d^2 \cdot n + d^3)$
Variance	0.0015	3.70	$O(n \cdot d)$
Random	~0	~0	$O(d)$

Among the nine evaluated methods, GBM-Permutation incurs the highest measured ranking time. This motivates a layered deployment architecture rather than any change to the algorithm. Layer 1, represented by a base station or gateway, can perform sensor ranking periodically using recently collected network measurements; the maximum measured ranking time and memory footprint in this study remain modest for the assumed gateway-level computing environment. The sensor nodes do not perform the ranking computation; instead, each node receives an active/inactive control decision from the gateway before the

relevant sensing period, thereby limiting the computational burden imposed on resource-constrained devices. The computational cost measured above is therefore incurred entirely in the layer where hardware constraints do not apply.

4.9. Neutral-Classifier Validation

Because GBM-Permutation ranks sensors using HistGradientBoosting and the original evaluation also uses HistGradientBoosting as the downstream classifier, the comparison could potentially favor the method through evaluator–selector alignment. To rule this out, the full (method, k) comparison was repeated using two classifiers structurally unrelated to the ranking mechanism: k-NN (distance-based) and SVM with an RBF kernel (margin-based), with features standardized for both (unlike HistGradientBoosting, which does not require this). The results are shown in Figure 12.

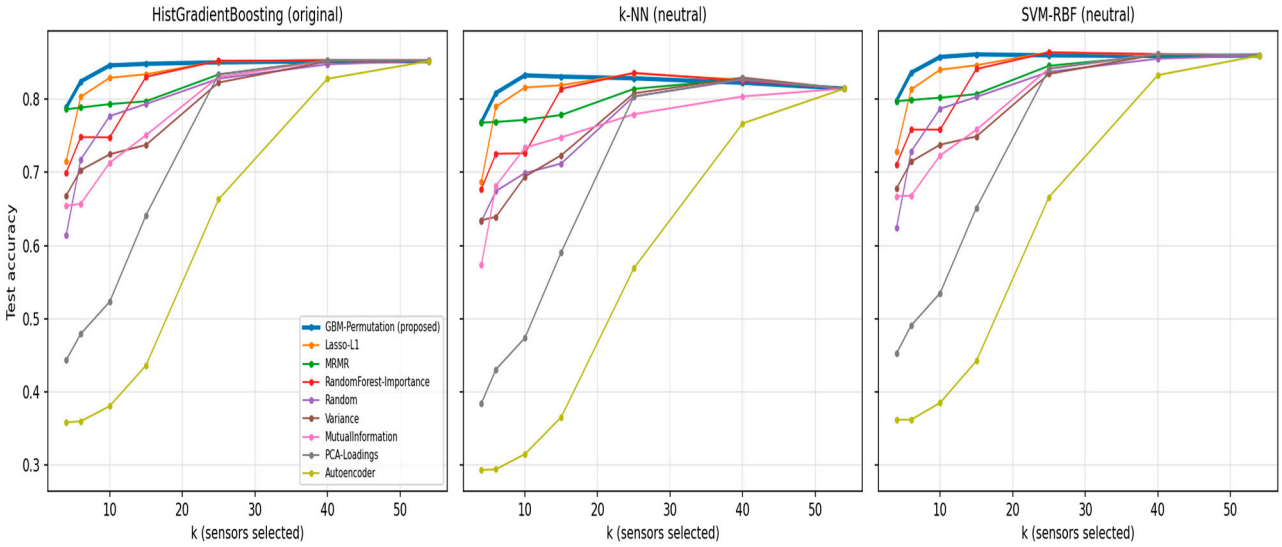

Figure 12. Method comparison under the original classifier and two neutral classifiers.

At k = 10, the relative ranking of the nine methods is preserved under both neutral classifiers, with Spearman ρ = 1.0000 and Kendall τ = 1.0000 relative to the original HistGradientBoosting-based evaluation. GBM-Permutation also retains a statistically significant advantage over all baselines under both neutral classifiers. Detailed accuracies and significance values are listed in Table 12.

Table 12. Accuracy and significance at k = 10 under neutral classifiers (8 seeds).

Method	k-NN Acc.	p (k-NN)	SVM-RBF Acc.	p (SVM)
GBM-Permutation	83.29%	—	85.81%	—
Lasso-L1	81.63%	0.0012 *	84.05%	0.0002 *
MRMR	77.19%	<0.0001 *	80.22%	<0.0001 *
Random	73.39%	0.0029 *	78.72%	0.0035 *
RandomForest-Importance	72.62%	<0.0001 *	75.86%	<0.0001 *
Variance	69.91%	<0.0001 *	73.80%	<0.0001 *
MutualInformation	69.43%	<0.0001 *	72.32%	<0.0001 *
PCA-Loadings	47.36%	<0.0001 *	53.48%	<0.0001 *
Autoencoder	31.45%	<0.0001 *	38.47%	<0.0001 *

* $p < 0.05$ vs. GBM-Permutation.

4.10. Robustness to Independent Data Regeneration

The main experiment in Section 3.6 uses a single seed-fixed benchmark instance (seed 42) and varies the train/test partition across eight seeds. To confirm results do not depend on this specific fixed draw, the full comparison was repeated with the benchmark. An additional robustness experiment independently regenerated the benchmark for each of eight seeds, generating new latent-factor samples and sensor-noise realizations while keeping the underlying class geometry fixed, thereby providing independent samples from the same generative distribution. The method ranking at $k = 10$ is essentially unchanged (GBM-Permutation 85.4%, Lasso-L1 83.6%, MRMR 78.9%, RandomForest-Importance/Random ≈ 76.9%, MutualInformation 74.5%, Variance 71.0%, PCA 58.1%, Autoencoder 35.4%). One notable change in ordering: RandomForest-Importance and Random, which were separated by roughly three percentage points under the original shared-generator design (74.80% vs. 77.71%), converge to statistically indistinguishable accuracy under independent regeneration (both ≈76.9%), indicating that their relative ordering in the original design was itself sensitive to the specific fixed draw. Holm–Bonferroni correction was applied across all eight pairwise comparisons, and paired effect sizes (Cohen's dz) with 95% confidence intervals were computed for both the original and the independently-regenerated design. The results are shown in Figure 13 and summarized in Table 13.

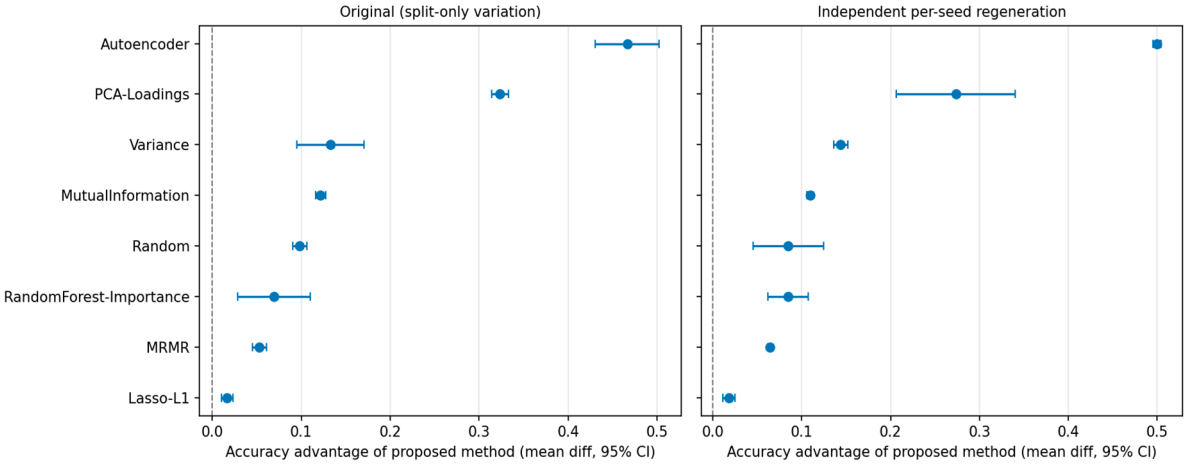

Figure 13. Mean accuracy differences between GBM-Permutation and each baseline, with 95% confidence intervals (the effect sizes reported in Table 13), under both experimental designs.

Table 13. Holm-corrected significance and effect sizes; independently-regenerated data ($k = 10$, 8 seeds).

Baseline	Mean Diff.	95% CI	Cohen's dz	Holm-Adj. p
Autoencoder	0.500	[0.495, 0.505]	89.4	<0.0001
PCA-Loadings	0.273	[0.206, 0.340]	3.4	0.0001
Variance	0.144	[0.136, 0.152]	15.0	<0.0001
MutualInformation	0.109	[0.106, 0.113]	27.0	<0.0001
Random	0.085	[0.045, 0.124]	1.8	0.0015
RandomForest-Importance	0.085	[0.062, 0.107]	3.1	0.0001
MRMR	0.064	[0.062, 0.067]	19.0	<0.0001
Lasso-L1	0.018	[0.011, 0.025]	2.3	0.0007

All pairwise comparisons remain statistically significant after multiplicity correction under both the original design and the independently regenerated benchmark design. Effect sizes range from large (Lasso-L1, dz ≈ 2.3) to very large (Autoencoder, dz ≈ 89);

because Cohen's dz is the mean paired difference divided by its between-seed standard deviation, and several comparisons (e.g., against Autoencoder) show a large, highly consistent accuracy gap across all eight seeds with near-zero between-seed variance in that gap, the denominator shrinks the statistic toward these large values even though the underlying accuracy differences themselves are modest in absolute terms. Tightening the experimental design if anything strengthened several effects rather than revealing them to be artifacts of a small, shared-generator sample.

4.11. Baseline Diagnostics

Standardizing the data before variance ranking forces the feature variances to approximately 1.0, making variance-based ranking ineffective; therefore, the variance baseline uses the original unstandardized features throughout the experiments; this is the only version of this method capable of ranking anything, consistent with standard practice. The autoencoder baseline was tested across bottleneck sizes {6, 12, 20, 30}: accuracy at $k = 4$ ranges from 0.36 to 0.63, confirming genuine hyperparameter sensitivity; the originally reported bottleneck (12 units) was retained rather than tuned to the best-observed value (six units, which happens to equal this benchmark's true latent dimensionality—information unavailable to a real deployment). The apparent non-monotonicity of Random-Forest-Importance between k = 6 (74.83%) and $k = 10$ (74.80%) is not statistically meaningful (paired t-test, $p = 0.89$).

The more substantive finding concerns why random selection outperforms several "informed" methods at k = 10. We measured how many of the benchmark's six underlying latent factors each method's top-10 selection actually covers. The results are shown in Figure 14.

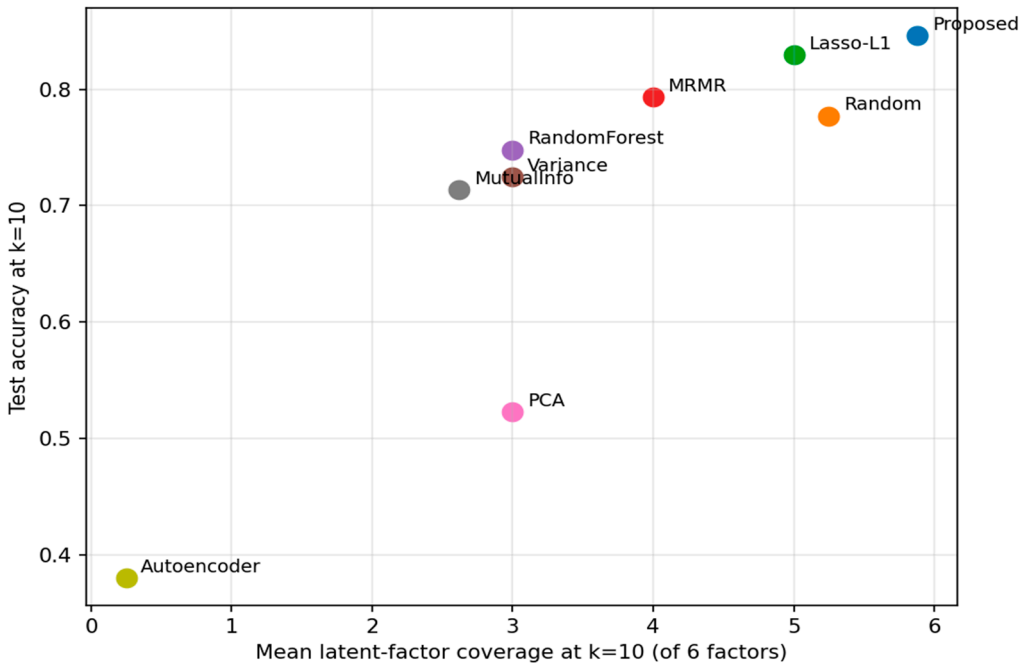

Figure 14. Latent-factor coverage vs. accuracy at $k = 10$ (mean, 8 seeds). Each marker color identifies one selection method, as labeled next to the marker.

Marginal, per-feature ranking methods (RandomForest-Importance, Variance, PCA, Mutual Information) concentrate their top-10 picks onto only three of six latent factors on average, leaving 3–4 factors entirely unrepresented in the selection; because the classification task depends on all six latent factors, incomplete factor coverage substantially constrains the achievable classification accuracy even when the selected individual sensors

are of high quality. Despite including some non-informative sensors, random selection covers an average of 5.25 of the six latent factors at $k = 10$, illustrating how broader factor coverage can outweigh marginal feature-ranking quality under strong redundancy. The GBM-Permutation method (5.88 of 6) and Lasso-L1 (5.00 of 6) achieve both broad factor coverage and sensor quality, which is the actual mechanism behind their advantage—not merely "better per-sensor importance scores." Factor coverage accounts for nearly the entire accuracy ranking in Table 4—a generalizable finding about a failure mode of marginal feature-ranking methods under sensor redundancy, rather than a weakness specific to how the baselines were implemented.

5. Discussion

The experimental results indicate that the GBM-Permutation approach provides its greatest benefit in the low-sensor operating region, where effective sensor selection is particularly important for reducing communication energy consumption. By recovering an average of 5.88 of the six ground-truth clean sensors at $k = 10$, the GBM-Permutation method achieves 99.3% of the full-network accuracy while using only 10 of 54 sensors, outperforming conventional ranking methods that rely on variance, PCA, or correlation-based criteria. Unlike purely marginal or unsupervised ranking criteria, permutation importance estimates the effect of perturbing each sensor on predictive performance, which can improve the identification of classification-relevant sensors in the presence of redundancy. This advantage is most evident under the medium-redundancy condition, where GBM-Permutation achieves statistically significant improvements over the principal baselines considered, whereas under high and low redundancy, its differences from simpler approaches such as Lasso-L1 are not statistically significant. The comparison with existing feature-selection techniques indicates that GBM-Permutation offers a favorable classification–sensor-reduction trade-off under the present evaluation framework; however, these results should not be interpreted as evidence of universal superiority because performance depends on the dataset structure, redundancy characteristics, and evaluation protocol. From a practical perspective, the results suggest that selecting an appropriate operating point from the accuracy–sensor trade-off curve can be as important as selecting the ranking algorithm itself, since most of the achievable accuracy is retained with only a small subset of sensors, leading to substantial reductions in energy consumption and significant network lifetime extension. Several limitations should be acknowledged. The primary evaluation uses a controlled synthetic benchmark with a known ground truth rather than measurements from a deployed WSN; the gradient-boosting implementation is based on scikit-learn's HistGradientBoostingClassifier rather than XGBoost; the redundancy–sensitivity analysis uses fewer repetitions than the main experiment; and the energy model accounts for communication energy but excludes sensing energy, MAC-layer overhead, retransmissions, routing effects, and topology changes. Although the real-data experiments show competitive performance, the largest advantage of GBM-Permutation occurs under meaningful sensor redundancy; future work should therefore evaluate the framework using real WSN datasets collected under diverse spatial, temporal, and deployment conditions.

Additional robustness and diagnostic analyses further support the main findings. The ablation study indicates that the observed advantage is associated primarily with the gradient-boosting model family rather than permutation importance alone. Evaluation with neutral classifiers and independently regenerated benchmark data produces consistent method rankings and statistically significant differences after multiplicity correction. The latent-factor analysis further explains why some marginal ranking methods underperform at small sensor counts, while the real sensor-derived validation provides additional support for the observed dependence on redundancy.

6. Conclusions

This paper presented an adaptive sensor-selection framework based on gradient-boosted permutation importance and evaluated it against eight independently re-implemented baselines under identical conditions across eight data partitions using a purpose-built benchmark with predefined ground-truth sensor informativeness. At the selected operating point, GBM-Permutation retains 99.3% of the full-sensor classification accuracy using 10 of 54 sensors, corresponding to an 82.6% reduction in modeled per-round communication energy and a 1.4-fold increase in modeled first-node-dies lifetime (883 vs. 615 rounds)—distinct from the sensor-count-based LEF of 5.4 reported alongside it; in the primary synthetic evaluation, the method significantly outperforms all evaluated baselines at this operating point. The diagnostic analyses further show why the method performs well: it recovers the ground-truth clean sensor core more effectively than competing rankings at low sensor counts, and that a common unsupervised alternative (variance thresholding) performs worse than random selection at small sensor counts—a cautionary, generalizable finding independent of the GBM-Permutation method specifically. A redundancy–severity sensitivity analysis showed this advantage is not universal: it is statistically indistinguishable from a simple Lasso-L1 baseline at both redundancy extremes, and it is clearest in the moderate-redundancy regime that most realistic deployments likely resemble. Validation on two real tabular datasets showed that the method remains competitive, although its advantage is narrower than in the synthetic benchmark; validation on the real sensor-derived Gas Sensor Array Drift dataset likewise showed no significant advantage over Lasso-L1 at the evaluated operating point, consistent with its advantage being tied specifically to the kind of sensor-level redundancy the synthetic benchmark was designed to contain.

Additional robustness and diagnostic analyses further support these conclusions, including ablation, neutral-classifier evaluation, independent benchmark regeneration, computational-cost analysis, and validation on real sensor-derived data. An ablation study confirming the source of the method's advantage is the model family rather than the permutation-importance mechanism specifically: this includes validation under two classifiers structurally unrelated to the ranking mechanism, preserving the exact method ranking; robustness to fully independent data regeneration with multiplicity-corrected, large-effect-size significance; a measured computational-cost analysis motivating a two-layer cloud/edge deployment architecture; and external validation on a real sensor-derived dataset whose outcome was correctly predicted in advance by the redundancy–sensitivity analysis.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/eng7090491/s1.

Author Contributions: Conceptualization, H.A.J. and R.S.J.; methodology, H.A.J.; software, H.A.J.; validation, H.A.J., S.K.I., A.S.A., M.A.A. and D.H.; formal analysis, H.A.J.; investigation, H.A.J.; resources, S.K.I., A.S.A., M.A.A. and D.H.; data curation, H.A.J.; writing—original draft preparation, H.A.J.; writing—review and editing, S.K.I., R.S.J., A.S.A., M.A.A. and D.H.; supervision, H.A.J., R.S.J., A.S.A., M.A.A. and D.H.; project administration, H.A.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data supporting the findings of this study are synthetic and script-generated (Section 3.2), with no privacy, ethical, or proprietary restriction on sharing. The generation

code, sensor-selection and baseline implementation code, and the complete row-level experimental results underlying every table and figure are openly available in the Supplementary Data and code package accompanying this submission, together with a README specifying exact reproduction commands. The Breast Cancer Wisconsin and Digits datasets are the versions bundled with scikit-learn. The Gas Sensor Array Drift Dataset is publicly available through the UCI Machine Learning Repository. All datasets used in the experiments are publicly accessible and require no proprietary data access.

Acknowledgments: The authors would like to acknowledge CSIT College—University of Basrah and Shatt Al-Arab University College for providing academic support and research facilities during the preparation of this work.

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

References

1. Habeeb, I.J.; Jasim, H.A.; Hashim, M.H. Balance energy based on duty cycle method for extending wireless sensor network lifetime. *Bull. Electr. Eng. Inform.* **2023**, *12*, 3105–3114. [CrossRef]
2. Krawiec, J.; Wybraniak-Kujawa, M.; Jacyna-Gołda, I.; Kotylak, P.; Panek, A.; Wojtachnik, R.; Siedlecka-Wójcikowska, T. Energy Footprint and Reliability of IoT Communication Protocols for Remote Sensor Networks. *Sensors* **2025**, *25*, 6042. [CrossRef] [PubMed] [PubMed Central]
3. Nkemeni, V.; Mieyeville, F.; Tsafack, P. Energy consumption reduction in wireless sensor network-based water pipeline monitoring systems via energy conservation techniques. *Future Internet* **2023**, *15*, 402. [CrossRef]
4. Adu-Manu, K.S.; Amoako, E.; Engmann, F. Advancements in Machine Learning-Enhanced Green Wireless Sensor Networks: A Comprehensive Survey on Energy Efficiency, Network Performance, and Future Directions. *J. Sens.* **2025**, *2025*, 5242517. [CrossRef]
5. Barnawi, A.Y.; Keshta, I.M. Energy Management in Wireless Sensor Networks Based on Naive Bayes, MLP, and SVM Classifications: A Comparative Study. *J. Sens.* **2016**, *2016*, 6250319. [CrossRef]
6. Aljawarneh, M.; Hamdaoui, R.; Zouinkhi, A.; Alangari, S.; Abdelkrim, M.N. Energy optimization for wireless sensor network using minimum redundancy maximum relevance feature selection and classification techniques. *PeerJ Comput. Sci.* **2024**, *10*, e1997. [CrossRef] [PubMed]
7. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2016; pp. 785–794.
8. Nagarajan, B.; Svn, S.K. An intelligent fault node diagnosis and classification approach using XGBoost with whale optimization in wireless sensor networks. *Peer-To-Peer Netw. Appl.* **2025**, *18*, 44. [CrossRef]
9. Nisar, F.; Amin, M.; Touseef Irshad, M.; Hadi, H.J.; Ahmad, N.; Ladan, M. XGBoost-driven adaptive spreading factor allocation for energy-efficient LoRaWAN networks. *Front. Commun. Netw.* **2025**, *6*, 1665262. [CrossRef]
10. Ouhmi, S.; Khalid, H.; Marwan, M.; Silkhi, H.; Temghart, A.A. Improved boosting-based machine learning algorithms for network intrusion detection in wireless sensor network. *IAES Int. J. Artif. Intell.* **2026**, *15*, 2278–2289. [CrossRef]
11. Pande, S.; Khamparia, A.; Gupta, D. Feature selection and comparison of classification algorithms for wireless sensor networks. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *14*, 1977–1989. [CrossRef]
12. Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*; IEEE: Piscataway, NJ, USA, 2000; p. 10.
13. Strobl, C.; Boulesteix, A.-L.; Zeileis, A.; Hothorn, T. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinform.* **2007**, *8*, 25. [CrossRef] [PubMed]
14. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [CrossRef] [PubMed]
15. Saha, A.; Pal, N.R. Group-feature (Sensor) selection with controlled redundancy using neural networks. *Neurocomputing* **2024**, *610*, 128596. [CrossRef]
16. Ambareesh, S.; Chavan, P.; Supreeth, S.; Nandalike, R.; Dayananda, P.; Rohith, S. A secure and energy-efficient routing using coupled ensemble selection approach and optimal type-2 fuzzy logic in WSN. *Sci. Rep.* **2025**, *15*, 38. [CrossRef] [PubMed]