

Effectiveness of Data Sampling Strategies for Maritime AIS Analytics: A One-Day Gulf of Mexico Study

Dhruv Kothari^{1*}, Mehek Kothari¹

Received September 13, 2026

Accepted October 20, 2026

Electronic access October 31, 2026

Automatic Identification Systems (AIS) generate millions of vessel positions per day, so training machine learning (ML) models on complete datasets is expensive. Prior AIS reduction work has evaluated trajectory simplification on downstream tasks; however, no reviewed study has compared multiple competing ping-level sampling strategies across multiple downstream ML tasks and anomaly-label definitions within one shared framework. We benchmark six sampling strategies across vessel classification (VC), fishing detection (FD), pooled anomaly detection (AD), and anomaly detection by vessel type (ADT) using 18 behavioral features from one day of Gulf of Mexico AIS traffic. Each strategy was tested at target sampling rates of 1%, 5%, 10%, 25%, 50%, and 100% across five seeds; lag-dependent features were recomputed from each retained sample before vessel-level aggregation. Full-data F1 scores were 0.593 for VC, 0.549 for FD, 0.898 for AD, and 0.516 for ADT, compared with majority-class baselines of 0.000–0.084. Sampling effectiveness varied by task and rate: Temporal reached 95.3% VC retention at 50%, several strategies reached at least 95% AD retention at 50%, and no strategy reached 95% FD or ADT retention below 100%. Removing the two timing features did not reduce Temporal sampling's F1 retention overall, with retention maintained or higher in 19 of 20 reduced-data task-rate conditions and lower in one. In the ADT second-model sensitivity analysis, strategy rankings also varied across classifiers, indicating that no single sampling strategy can be treated as universally best.

Keywords: Automatic Identification System (AIS), Anomaly Detection, Machine Learning (ML), Vessel Classification, Data Sampling, Maritime Data

Introduction

Maritime analytics rely on Automatic Identification System (AIS), a transponder-based tracking system mandated by the International Maritime Organization for certain vessels, including ships of 300 gross tonnage and above on international voyages¹. Vessels can broadcast dynamic data as frequently as every 2 to 10 seconds, depending on vessel speed and course. The U.S. MarineCadastre.gov AIS dataset contained approximately 3.1 billion vessel-position records in 2024, or about 8.5 million records per day on average², creating a challenge that limits accessibility and operational deployments for smaller research institutions. Vessels traveling at relatively constant speed and course can generate many near-identical AIS positions, creating substantial redundancy³. This redundancy can increase storage and computational costs, motivating methods for intelligent sampling that reduce AIS data while preserving information needed for classification and anomaly detection.

A key question is how to choose which AIS pings to keep. The simplest approach is random sampling, but there are al-

ternatives that allow us to favor important vessel types, keep more data near ports, preserve rare vessel types, retain pings where vessels change speed and direction, or keep fewer pings over time.

Previous research has shown that AIS data is a valuable data source for understanding vessel behavior. Kroodsma et al.⁴ used global AIS data to map fishing activity across more than half of the world's oceans, demonstrating that AIS can be used for large-scale fisheries monitoring. Other studies have trained machine learning models, such as Random Forests, to classify vessel types using AIS data⁵, but these studies usually focus on improving accuracy rather than reducing the amount of training data needed. Researchers have also used unsupervised techniques such as Isolation Forest⁶ to identify anomalous vessel behavior without labeling. In this study, anomaly detection is evaluated using two anomaly label definitions, one generated across all vessel types and one generated separately within each vessel type.

Beyond these studies, a substantial body of work has proposed AIS trajectory simplification and compression methods^{7–11}. Additional approaches use multi-objective or handling-pattern-aware criteria^{12,13}. These studies evaluate compression quality using geometric criteria such as com-

¹ Emerald High School, Dublin, California, USA

* Corresponding author. Email: reachdhruv5@gmail.com

pression ratio and trajectory shape preservation, rather than any downstream machine-learning task. Additional studies have applied machine learning to AIS-based vessel classification and fishing-activity detection^{14–16}, to anomaly detection using both classical and deep-learning methods^{17–21}, and to trajectory and arrival-time prediction^{22–26}; none of these examines whether a sampling or reduction strategy’s effectiveness generalizes across more than one of these task families, or across more than one way of defining ground truth for the same task. Most relevant to the present study, He, Ruan, and Tong²⁷ recently proposed an AIS route-simplification method and evaluated it against two downstream task families, trajectory prediction and anomaly detection, reporting that simplified data reduced training time and improved accuracy on both. This shows that a single AIS simplification method’s effect on downstream ML can differ across task families, but it does not compare competing reduction strategies against one another, and it does not test a shared vessel classification and anomaly detection task set of the kind evaluated here. The closest precedent for comparing multiple reduction methods at all, Ferreira et al.²⁸, compares several existing compression algorithms by their effect on one downstream task, unsupervised trajectory clustering, rather than the six ping-level sampling strategies and four vessel-level tasks evaluated in this study.

Outside the maritime field, other data selection research explores methods for choosing smaller, representative training subsets while maintaining model performance^{29,30}. These include coreset, gradient-matching, and proxy-based subset-selection methods^{31–33}. Other approaches use submodular selection, example-forgetting behavior, or early-training importance^{34–36}. However, whether a given sampling approach transfers consistently across different downstream AIS tasks and different ground truth definitions remains an open question.

Previous studies have generally evaluated data-reduction methods within limited downstream-task settings. This study therefore examines two questions: whether a sampling strategy that works well for one task also works well for other tasks, and whether the results change with the anomaly-label definition, particularly when anomalies are generated by a model rather than from independently labeled data.

To answer these questions, we used one day of AIS data from the Gulf of Mexico and evaluated four machine learning tasks using the same set of 18 behavioral features for each vessel. We compared six different sampling strategies at six target sampling rates ranging from 1% to 100% of the training data. For a fair comparison, within each task, all six sampling strategies used the same vessel-level train/test split, feature definitions, model configuration, and fixed ground-truth labels where applicable. Among the studies reviewed here, none compared multiple competing AIS sampling strategies

across multiple downstream machine-learning tasks within one shared evaluation framework, which is the gap this study addresses.

Methods

Research Design

This study uses an experimental benchmarking design, applying six data sampling strategies to publicly available AIS telemetry data and comparing their downstream performance across four machine learning tasks.

Dataset and Preprocessing

Public AIS data were obtained from NOAA MarineCadastreregov² for October 6, 2024, covering Gulf of Mexico (18°N–31°N, 97°W–80°W). The raw dataset contains 2,895,679 AIS position records from 4,770 unique vessels. We remove records with sensor errors (Speed Over Ground greater than 50 knots, or a null SOG value), remove invalid MMSI identifiers outside the ITU range 100,000,000–999,999,999, and remove vessels with fewer than 30 total pings. Table 1 reports the rows and vessels removed at each of these steps. After these filters, the dataset contains 2,887,826 rows from 4,439 vessels.

Table 1 Rows and vessels removed at each preprocessing step.

Step	Rows before	Rows after	Rows removed	Vessels before	Vessels after	Vessels removed
Raw input	—	2,895,679	—	—	4,770	—
SOG > 50 kn or null	2,895,679	2,890,725	4,954	4,770	4,762	8
Invalid MMSI < 30	2,890,725	2,890,725	0	4,762	4,762	0
pings/vessel	2,890,725	2,887,826	2,899	4,762	4,439	323

An earlier version of this table additionally reported 2,577,223 pings from 4,026 vessels after removing records without a valid previous SOG, COG, or timestamp value, computed once globally before sampling. We traced this step’s larger-than-expected row and vessel removal (310,603 rows and 413 vessels) to COG being independently null on 10.63% of all rows before any lag is taken, which caused the dropna step to remove far more than one row per vessel and to eliminate entirely any vessel whose pings fell within that null-COG population. Because lag-dependent features are now recomputed separately for each sampling condition rather than once globally (see Feature Engineering), this global step is no longer part of the pipeline, and the 4,439-vessel cleaned dataset in Table 1 above is the input to the train/test split below.

The data are then sorted by vessel and timestamp. Each ping’s numeric AIS vessel-type code is mapped into one of

seven groups: fishing (30); towing/tug (31, 32, 52); pleasure craft (36, 37); passenger (60–69); cargo (70–79); tanker (80–89); and other for missing or unmatched codes. Each vessel was then assigned the group occurring most frequently across its pings.

Feature Engineering

We compute 18 behavioral features for each vessel from the retained AIS pings. To isolate the effect of sampling on task performance, static vessel properties such as length and width are excluded because they do not change with ping-level sampling. The speed features (6) are the mean, standard deviation, median, maximum, range of SOG, and fraction of pings where the vessel is stationary ($\text{SOG} < 0.5$ knots). The speed change features (2) are the mean absolute change and standard deviation of change in SOG between consecutive pings, capturing the size and frequency of speed changes. The course-change features (3) are the mean absolute change, standard deviation of change, and maximum absolute change in COG between consecutive pings, capturing the size and sharpness of course changes, with correction for the $0^\circ/360^\circ$ wraparound. The location features (5) are the mean and standard deviation of Haversine distance (the distance between two latitude/longitude points on the Earth's surface) to the nearest of eight reference ports (Houston/Galveston, New Orleans, Tampa, Mobile, Port Arthur, Corpus Christi, Pensacola, and Miami), the fraction of pings within 5 nautical miles of the nearest reference port, mean latitude, and mean longitude. The ping-timing features (2) are the median and standard deviation of the time between consecutive AIS broadcasts.

Because the speed-change, course-change, and ping-timing features depend on each ping's immediately preceding ping, this preceding-ping relationship is recomputed separately for every sampling condition from only the pings that condition retains, after sorting and after sampling, rather than computed once against the full unsampled sequence and carried through into whichever rows a sampling strategy happens to retain. The latter approach, used in an earlier version of this pipeline, allowed a retained ping's feature values to reflect information from pings that sampling was supposed to discard; recomputing these features from each condition's own retained sequence, as described here, corrects this (see Evaluation Consistency Check). Adaptive sampling's event-selection rule (see Sampling Protocol) is permitted to reference the full unsampled training stream only to decide which pings qualify as high-change event pings; once selected, those pings' own feature values are still computed from the retained sequence alone. All model features are aggregated at the vessel level before model training. Vessel records with any missing value among the 18 behavioral features were dropped from the train/test set post-aggregation; the number of vessels retained

after this step varies by strategy and rate (see Results, Table 9).

Experimental Framework

We evaluated four machine learning tasks, measuring performance using F1 score. Vessel Classification (VC) classifies vessels into one of 7 vessel types using a Random Forest model.

Fishing Detection (FD) is a binary classification of vessels as fishing or non-fishing. Pooled Anomaly Detection (AD) detects vessels where vessel behavior is anomalous: a single Isolation Forest trained across all vessel types generates fixed anomaly labels, and Random Forest models trained on sampled AIS data are evaluated on their ability to reproduce these labels. Anomaly Detection by Type (ADT) detects vessels where vessel behavior is anomalous compared to other vessels of the same type: separate Isolation Forest models are trained for each vessel type to generate fixed anomaly labels, allowing vessels to be compared only with others of the same type, and Random Forest models trained on sampled AIS data are then evaluated on their ability to reproduce these labels. Because both anomaly tasks are trained to reproduce labels created by another model on the same features, our results measure agreement with those model-generated labels rather than detection of confirmed real-world anomalies (see also Limitations); Table 8 (Results) reports the number of vessels and positive labels within each vessel type for both tasks.

For Vessel classification and Fishing detection, the ground-truth labels are derived from the vessels' self-reported vessel types. For the primary 18-feature analysis, anomaly labels are generated once from the full, unsampled training dataset and used across all sampling strategies, rates, and seeds. For the 16-feature sensitivity analysis, AD and ADT anomaly labels are regenerated using the corresponding 16-feature specification. Test-set labels are generated using the same fitted Isolation Forest models. The objective is to measure how well models trained on sampled AIS data reproduce these labels rather than to evaluate the Isolation Forest algorithm itself.

Model Configuration

We used 100 trees with balanced class weights in the Random Forest models for Vessel Classification (VC), Fishing Detection (FD), and both anomaly detection tasks. Additionally, we kept the standard scikit-learn settings for the other Random Forest parameters and used random seeds from 0 to 4 for the repeated experiments. We used Isolation Forest models to create the anomaly labels used as ground truth for pooled anomaly detection (AD) and anomaly detection by type (ADT). Each model used 200 trees, a contamination rate of 0.05, and a fixed random seed of 42. For the primary anal-

ysis, each tree used up to 256 training samples and all 18 features; for the 16-feature sensitivity analysis, the two ping-timing features were omitted. For ADT, a separate Isolation Forest model was trained for each vessel type. All experiments were run in Python 3.11.2 using scikit-learn 1.9.0.

Train/Test Protocol

We split the dataset by vessel rather than by individual pings, using 80% of the 4,439-vessel cleaned dataset (3,551 vessels) for training and 20% (888 vessels) for testing using a fixed random state (seed = 42) to ensure reproducibility. This ensures that no vessel appears in both sets and prevents the models from learning vessel-specific patterns. Because lag-dependent features are now recomputed per sampling condition rather than once globally (see Feature Engineering), the split is built from the full cleaned dataset rather than a further lag-filtered subset; after per-condition feature aggregation, 3,166 of the 3,551 train vessels and 796 of the 888 test vessels have complete features at the full (100%) sampling rate. Sampling is applied only to the training data; the test set retains all pings, providing a consistent benchmark across sampling strategies. The processing order for each experiment is: clean the data, split vessels into training and test sets, sample the training pings, recompute lag-dependent features from the retained ping sequence, aggregate features to the vessel level, and train and evaluate the model.

Sampling Protocol

Each of the six sampling strategies is evaluated across four machine learning tasks at six target sampling rates: 1%, 5%, 10%, 25%, 50%, and 100%. Each experiment is repeated using five random seeds (0–4), yielding 720 trained Random Forest models (6 sampling strategies $\times$ 4 tasks $\times$ 6 sampling rates $\times$ 5 seeds).

Random: Selects pings uniformly at random from all training pings at the target sampling rate. **Stratified:** Samples pings independently within each of the seven vessel-type groups using the same target sampling rate for every group, preserving representation across vessel types; because vessel type is also the label used for Vessel Classification and, via the fishing/non-fishing split, for Fishing Detection, Stratified sampling has indirect access to this label-correlated information for those two tasks, unlike the other four strategies. **Spatial:** Divides pings into three zones based on distance to the nearest port: near-port (<5 nautical miles), coastal (5 to <50 nautical miles), and offshore ($\geq$ 50 nautical miles). The total sampling budget at each target rate is allocated 60%, 30%, and 10% across the near-port, coastal and offshore zones, respectively. If a zone cannot fill its allocation, the remaining budget is redistributed first to the near-port zone and then to the coastal

zone. Pings are sampled uniformly within each zone. **Importance:** Allocates the sampling budget across vessel types using predefined heuristic weights: fishing (3.0), tanker (2.0), cargo (1.5), passenger (1.0), towing/tug (0.5), pleasure craft (0.5), and other (0.3). Allocation is proportional to each vessel type's weight and available ping count, followed by uniform sampling within each type. Like Stratified Sampling, Importance sampling also uses the vessel-type label directly and therefore shares the same asymmetry relative to Vessel Classification and Fishing Detection. These weights were selected heuristically, meaning they were chosen based on predefined priorities rather than derived through model optimization. At target rates of 1%, 5%, 10%, and 25%, Importance sampling retains the requested proportion of training data. At the nominal 50% target, it retains 43.23% because the high-weight fishing and tanker groups exhaust their available pings and unused allocation is not redistributed to other vessel types. **Adaptive:** Prioritizes event pings where the absolute change in Speed Over Ground exceeds 1.0 knot or the absolute change in Course Over Ground exceeds 5°, identified using the full unsampled training stream. If event pings do not fill the target sampling budget, the remaining budget is filled by uniformly sampling stable pings. If event pings exceed the target budget, event pings are randomly sampled down to the target size. **Temporal:** Retains the first ping for each vessel within fixed time windows selected to approximate the target sampling rates. The resulting windows are 180, 45, 20, 7, and 3 minutes for nominal sampling rates of 1%, 5%, 10%, 25%, and 50%, respectively; the 100% condition retains all pings. Because sampling is constrained to fixed time windows, the corresponding actual retained proportions are approximately 1.14%, 4.44%, 9.83%, 26.49%, and 55.36%, respectively. Temporal sampling is deterministic for a given time window.

Evaluation Metrics

Performance is evaluated using F1 score. For the vessel-classification task, macro-averaged F1 is used so that each vessel type contributes equally to the metric. For fishing detection and both anomaly-detection tasks, binary F1 is used with fishing or anomaly as the positive class.

F1 scores are computed for each of the five random-seed repetitions. Performance retention relative to the full training dataset is calculated per seed, against that same seed's own full-data F1, as $\text{Retention (\%)} = 100 \times \text{F1}(\text{sampled, seed}) / \text{F1}(\text{full dataset, same seed})$, and the resulting five per-seed retention values are then averaged to give the mean and standard deviation reported in every table. Computing retention this way, rather than dividing a five-seed average by a single fixed baseline, ensures that the 100% condition – where every strategy trains on identical data – is exactly 100% retention with zero standard deviation for every seed by construction;

any ranking whose gap between strategies is smaller than the reported standard deviation should not be treated as a stable difference. Within each task, all sampling strategies use the same train/test split, feature definitions, model configuration, and fixed ground-truth labels.

Evaluation Consistency Check

After identifying a train/test split mismatch in the original Random sampling evaluation for the ADT task, we repeated the analysis using the unified vessel-level train/test split applied throughout the rest of the study. The purpose of this check was to determine how much the inconsistent split affected measured F1 retention and whether it could change the apparent sampling rate required to reach the 95% retention threshold. We compared the original and corrected pipelines at target sampling rates of 1%, 5%, 10%, 25%, and 50%. The effect of this correction on F1 retention is reported in Table 10. This mismatch was confined to this single strategy-task combination: the affected implementation contained logic only for the ADT task and only for random sampling, using a vessel-type-stratified split, while every other strategy-task combination was always computed through the shared pipeline described above, using the unstratified split.

Separately, we identified that lag-dependent features (speed change, course change, ping timing) had been computed once, before sampling, against each vessel's full unsampled sequence, rather than recomputed from each sampling condition's own retained sequence as described in Feature Engineering. We corrected this and reran the complete experiment grid; all primary results reported in Tables 2-9 and Figure 1 come from this corrected pipeline, and are not directly comparable in magnitude to the pre-correction values reported in the unified-split column of Table 10, which was computed before this correction was made. Correcting this issue reduced measured retention most at low sampling rates, since that is where the retained sequence differs most from the full sequence. As a further check, we repeated all four tasks with the two ping-timing features removed (16 features total instead of 18). Across the 20 reduced-data task-rate conditions for Temporal sampling, retention was maintained or higher without the timing features in 19 conditions and lower in one. Across all 24 task-rate comparisons, including the four 100% conditions, the 18-feature and 16-feature ± 1 standard-deviation intervals overlapped in 62% of cases; where they did not overlap, retention was higher without the timing features. The mean retention difference across all 24 comparisons was 7.7 percentage points in favor of the 16-feature specification. These results do not support the interpretation that Temporal sampling's relative performance is caused by the two ping-timing features alone. We additionally repeated the ADT task at both 18 and 16 features using a scikit-learn

Pipeline consisting of StandardScaler followed by Logistic Regression (class_weight='balanced', C=1.0, max_iter=5000, solver='lbfgs'). This second-model sensitivity analysis was performed for the ADT task only.

The sampling, feature-recomputation, and evaluation code used to produce every result in this paper is available in the GitHub repository: <https://github.com/kotharidhruv/effectiveness-of-data-sampling-strategies-maritime-ais>

Results

Table 2 reports the full-data F1 score for each task, alongside the corresponding majority-class baseline.

Table 2 Full-data F1 score and majority-class baseline by task.

Task	Full-data F1 (mean $\pm$ SD, 5 seeds)	Majority-class F1
Vessel Classification (VC)	0.593 $\pm$ 0.008	0.084 (towing/tug)
Fishing Detection (FD)	0.549 $\pm$ 0.018	0.000
Pooled Anomaly Detection (AD)	0.898 $\pm$ 0.010	0.000
Anomaly Detection by Type (ADT)	0.516 $\pm$ 0.011	0.000

Overall Sampling Efficiency

Table 3 summarizes the minimum target sampling rate at which each strategy reaches 95% F1 retention for each task, computed from the corrected pipeline described in Methods. For Vessel Classification (VC), only Temporal sampling reaches the threshold, at the 50% target; no other strategy reaches it below the full dataset. For Fishing Detection (FD), no strategy reaches the threshold below the full dataset. For pooled anomaly detection (AD), Random, Stratified, Adaptive, and Temporal reach the threshold at the 50% target; Spatial and Importance do not reach it below the full dataset. For Anomaly Detection by Type (ADT), no strategy reaches 95% retention below the full dataset within the rates tested here. Temporal has the highest mean retention at the 50% rate (81.1%), closely followed by Adaptive (79.4%); because their ± 1 SD intervals overlap, this ordering should not be interpreted as a stable strategy ranking.

Rates indicate nominal target sampling rates. Temporal and Importance sampling's actual retained proportions differ from these nominal targets at some rates (see Sampling Protocol). "Not reached $\leq 50\%$ " indicates the strategy did not reach 95% retention at any nominal rate at or below 50%; every strategy trivially reaches 100% retention at the 100% (full-data) condition, which is therefore omitted from Figure 1 below.

Table 3 Minimum target sampling rate required to reach 95% F1 retention.

Strategy	VC	FD	AD	ADT
Random	not reached ≤50%	not reached ≤50%	50%	not reached ≤50%
Stratified	not reached ≤50%	not reached ≤50%	50%	not reached ≤50%
Spatial	not reached ≤50%	not reached ≤50%	not reached ≤50%	not reached ≤50%
Importance	not reached ≤50%	not reached ≤50%	not reached ≤50%	not reached ≤50%
Adaptive	not reached ≤50%	not reached ≤50%	50%	not reached ≤50%
Temporal	50%	not reached ≤50%	50%	not reached ≤50%

Figure 1 summarizes F1 retention across the five tested sampling rates below 100% for the four evaluated tasks.

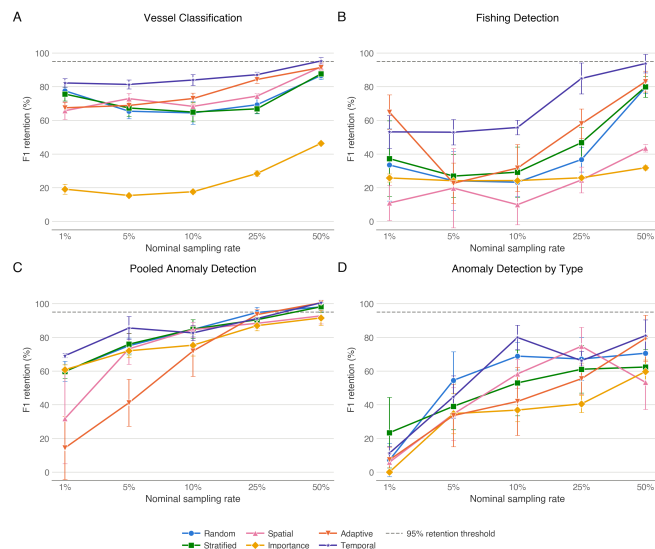


Figure. 1 F1 retention across target sampling rates for VC, FD, AD, and ADT. A) Vessel Classification (VC), B) Fishing Detection (FD), C) Pooled Anomaly Detection (AD), and D) Anomaly Detection by Type (ADT).

Points represent mean retention across five repetitions and error bars show ± 1 standard deviation. The dashed line marks the 95% retention threshold. Sampling rates on the x-axis are nominal targets; actual retained proportions for Temporal and Importance differ from some nominal targets as described in Sampling Protocol. The 100% condition is omitted because retention there is exactly 100% with zero variance by construction (see Evaluation Metrics). For Temporal, the same pings are selected in every seed run, so differences across the five runs come only from Random Forest initialization. For the other strategies, differences across runs come from both the sampled pings and the Random Forest initialization.

Table 4 reports full retention values for Vessel Classification

Table 4 Vessel Classification (VC) F1 retention (%) by strategy and sampling rate.

Strategy	1%	5%	10%	25%	50%
Random	77.5 $\pm$ 2.3	65.4 $\pm$ 4.4	64.4 $\pm$ 6.8	69.3 $\pm$ 5.3	86.6 $\pm$ 2.4
Stratified	75.6 $\pm$ 4.1	67.3 $\pm$ 4.9	64.9 $\pm$ 5.8	66.8 $\pm$ 2.7	87.6 $\pm$ 2.5
Spatial	65.7 $\pm$ 5.2	72.9 $\pm$ 3.1	68.3 $\pm$ 5.6	74.4 $\pm$ 1.5	91.8 $\pm$ 1.9
Importance	19.1 $\pm$ 2.9	15.4 $\pm$ 0.9	17.6 $\pm$ 1.2	28.4 $\pm$ 1.7	46.3 $\pm$ 1.1
Adaptive	67.5 $\pm$ 3.1	68.8 $\pm$ 2.3	73.0 $\pm$ 3.3	84.3 $\pm$ 2.3	91.3 $\pm$ 3.1
Temporal	82.2 $\pm$ 2.6	81.3 $\pm$ 2.7	83.9 $\pm$ 3.3	87.1 $\pm$ 1.5	95.3 $\pm$ 2.1

tion, and Table 5 reports full retention values for Fishing Detection, by strategy and sampling rate.

Table 5 Fishing Detection (FD) F1 retention (%) by strategy and sampling rate.

Strategy	1%	5%	10%	25%	50%
Random	33.5 $\pm$ 21.4	24.1 $\pm$ 17.5	23.2 $\pm$ 8.4	36.7 $\pm$ 7.4	79.6 $\pm$ 3.6
Stratified	37.2 $\pm$ 22.5	26.9 $\pm$ 12.8	29.2 $\pm$ 15.0	46.7 $\pm$ 9.1	79.9 $\pm$ 6.3
Spatial	10.9 $\pm$ 10.4	19.7 $\pm$ 23.6	9.9 $\pm$ 11.9	24.6 $\pm$ 7.7	43.5 $\pm$ 2.3
Importance	25.8 $\pm$ 2.4	24.1 $\pm$ 1.5	24.2 $\pm$ 1.0	25.9 $\pm$ 0.5	31.9 $\pm$ 1.5
Adaptive	64.8 $\pm$ 10.4	22.6 $\pm$ 12.0	31.6 $\pm$ 14.0	58.0 $\pm$ 8.8	83.0 $\pm$ 6.2
Temporal	53.2 $\pm$ 9.8	52.9 $\pm$ 7.5	55.7 $\pm$ 4.3	84.9 $\pm$ 9.2	93.7 $\pm$ 5.5

Pooled Anomaly Detection

Table 6 shows the F1 retention for different sampling strategies at various target sampling rates for the Pooled Anomaly Detection task. Adaptive sampling has the lowest retention at the 1% target (14.4%), compared with 31.8%–69.1% for the other strategies at that rate; this value carries a standard deviation of 18.7, indicating substantial seed-to-seed instability at this rate. Random, Stratified, Adaptive, and Temporal reach the 95% threshold at 50% target, where Adaptive (100.5%) and Temporal (100.6%) are the two highest-retention strategies, within one standard deviation of each other. Spatial and Importance remain below the threshold at 50% (92.8% and 91.6%, respectively).

Table 6 Pooled Anomaly Detection (AD) F1 retention (%) by strategy and sampling rate.

Strategy	1%	5%	10%	25%	50%
Random	59.8 $\pm$ 5.9	75.0 $\pm$ 4.6	84.7 $\pm$ 4.4	94.8 $\pm$ 3.1	98.1 $\pm$ 2.4
Stratified	59.7 $\pm$ 4.0	76.0 $\pm$ 6.3	84.9 $\pm$ 5.6	90.5 $\pm$ 3.1	98.2 $\pm$ 3.4
Spatial	31.8 $\pm$ 26.7	73.2 $\pm$ 9.2	84.6 $\pm$ 4.4	88.4 $\pm$ 2.9	92.8 $\pm$ 4.7
Importance	60.8 $\pm$ 2.9	72.0 $\pm$ 4.1	75.4 $\pm$ 6.4	87.0 $\pm$ 3.1	91.6 $\pm$ 4.4
Adaptive	14.4 $\pm$ 18.7	41.2 $\pm$ 14.0	71.9 $\pm$ 15.0	93.5 $\pm$ 3.3	100.5 $\pm$ 1.1
Temporal	69.1 $\pm$ 1.2	85.6 $\pm$ 6.8	82.7 $\pm$ 4.4	91.2 $\pm$ 3.1	100.6 $\pm$ 1.6

Retention values above 100% indicate that the sampled model achieved a slightly higher F1 score than the corresponding full-data baseline.

Anomaly Detection by Type

Table 7 shows the F1 retention for different sampling strategies at various target sampling rates for the Anomaly Detection by Type task. It shows substantially lower retention for ADT than for pooled AD at every rate tested. At the nominal 50% target, Temporal has the highest mean retention (81.1%), closely followed by Adaptive (79.4%). Their ± 1 SD intervals overlap, so this ordering should not be interpreted as a stable difference between the two strategies. Random, Stratified, Importance, and Spatial have mean retention values of 70.6%, 62.4%, 59.7%, and 53.3%, respectively. No strategy reaches the 95% retention threshold below the full dataset for this task within the rates tested here. Temporal is not uniformly the strongest strategy at every rate: Random has higher mean retention than Temporal at the 5% rate (54.5% vs 44.7%), and Spatial has higher mean retention than Temporal at the 25% rate (74.7% vs. 66.2%), though Spatial's retention then falls at the 50% rate (53.3%) with a standard deviation (16.0) large enough that this pattern should be treated cautiously.

Table 7 Anomaly Detection by Type (ADT) F1 retention (%) by strategy and sampling rate

Strategy	1%	5%	10%	25%	50%
Random	7.2 $\pm$ 9.9	54.5 $\pm$ 17.0	68.8 $\pm$ 8.1	67.1 $\pm$ 12.7	70.6 $\pm$ 8.6
Stratified	23.4 $\pm$ 20.9	39.1 $\pm$ 13.9	53.0 $\pm$ 19.5	61.1 $\pm$ 14.1	62.4 $\pm$ 10.3
Spatial	5.9 $\pm$ 5.4	34.5 $\pm$ 15.7	58.3 $\pm$ 8.7	74.7 $\pm$ 11.2	53.3 $\pm$ 16.0
Importance	0.0 $\pm$ 0.0	34.7 $\pm$ 11.9	36.8 $\pm$ 6.9	40.5 $\pm$ 5.1	59.7 $\pm$ 6.8
Adaptive	7.3 $\pm$ 7.5	33.6 $\pm$ 18.5	42.0 $\pm$ 20.2	55.5 $\pm$ 9.3	79.4 $\pm$ 13.7
Temporal	11.2 $\pm$ 3.9	44.7 $\pm$ 12.5	79.9 $\pm$ 7.3	66.2 $\pm$ 5.5	81.1 $\pm$ 9.3

Table 8 reports the number of training vessels with complete 18-feature vectors and the number and percentage of positive anomaly labels within each vessel type for both anomaly-detection label definitions. Several vessel types contain relatively few positive ADT labels, including passenger (10), fishing (13), tanker (13), and cargo (18), which may contribute to the wide seed-to-seed variability observed for ADT in Table 7 and Figure 1D.

Table 8 Vessel counts and positive anomaly labels by vessel type (complete-feature training population)

Vessel type	Total vessels	AD positive	AD positive %	ADT positive	ADT positive %
Cargo	352	31	8.8%	18	5.1%
Fishing	244	3	1.2%	13	5.3%
Other	427	27	6.3%	22	5.2%
Passenger	199	33	16.6%	10	5.0%
Pleasure	401	43	10.7%	20	5.0%
Tanker	260	17	6.5%	13	5.0%
Towing/tug	1,283	5	0.4%	65	5.1%

Table 9 reports the mean number of training vessels retained after per-condition feature aggregation, averaged across tasks and seeds, for each strategy and rate. Spatial and Adaptive sampling retain noticeably fewer vessels than the other strate-

gies at low rates because both concentrate their sampling budget on a ping subset not guaranteed to include at least one ping per vessel; Temporal sampling, which guarantees at least one ping per vessel per time window by construction, retains the most vessels at every rate.

Table 9 Mean training vessels retained after feature aggregation, by strategy and target rate (out of 3,551 raw train-split vessels).

Strategy	1%	5%	10%	25%	50%	100%
Random	2,342.8	2,984.8	3,091.0	3,140.2	3,154.6	3,166.0
Stratified	2,337.6	2,982.6	3,085.8	3,141.0	3,155.0	3,166.0
Spatial	1,715.8	2,772.0	2,956.6	3,097.0	3,145.4	3,166.0
Importance	2,182.6	2,931.6	3,038.4	3,122.8	3,149.0	3,166.0
Adaptive	1,830.8	2,452.6	2,612.8	2,769.6	3,159.4	3,166.0
Temporal	2,975.0	3,123.0	3,150.0	3,163.0	3,165.0	3,166.0

Evaluation Protocol Sensitivity

Table 10 quantifies the effect of the train/test split mismatch identified during evaluation of Random sampling on the ADT task, computed prior to the lag-recomputation correction described in Methods (see Evaluation Consistency Check). This unified-split column is not directly comparable in magnitude to the fully corrected Random/ADT values reported in Table 7, which additionally incorporate that correction; Table 10 isolates the split-mismatch effect alone.

Under the mismatched pipeline, Random sampling appeared to reach 95.8% F1 retention at the 10% target sampling rate, while the unified pipeline produced 72.3% retention. Therefore, the original mismatched split inflated measured performance by 23.5 percentage points at this one rate. Lower retention under the unified split is also observed at the 1%, 5%, 25%, and 50% sampling rates. This comparison reflects a single, isolated case rather than a general estimate of split-mismatch effects, and was architecturally confined to this one strategy-task combination for the reasons described in Methods.

Table 10 Effect of train/test split mismatch on F1 retention (%) for Random sampling on the ADT task (prior to the lag-recomputation correction).

Rate	Mismatched split	Unified split
1%	48.4	35.1
5%	90.8	63.6
10%	95.8	72.3
25%	98.5	87.3
50%	98.6	88.0

Table 11 reports Temporal sampling F1 retention using the original 18-feature specification and the 16-feature specification with the two ping-timing features removed, across all four tasks and reduced sampling rates. Temporal retention was maintained or higher with 16 features in 19 of the 20 reduced-data task-rate conditions and lower in one condition, indicat-

ing that its relative performance was not reduced overall when the two ping-timing features were removed. For AD and ADT, anomaly labels were regenerated using the corresponding 16-feature specification, so differences for these tasks reflect sensitivity to the full 16-feature specification rather than removal of the timing predictors alone.

Table 11 Temporal F1 retention (%) with 18 and 16 features by task and sampling rate

Task / Features	1%	5%	10%	25%	50%
VC – 18 features	82.2 ± 2.6	81.3 ± 2.7	83.9 ± 3.3	87.1 ± 1.5	95.3 ± 2.1
VC – 16 features	87.1 ± 3.8	83.7 ± 1.1	85.6 ± 1.4	96.0 ± 1.0	101.4 ± 2.2
FD – 18 features	53.2 ± 9.8	52.9 ± 7.5	55.7 ± 4.3	84.9 ± 9.2	93.7 ± 5.5
FD – 16 features	53.2 ± 8.6	58.0 ± 6.0	59.9 ± 1.0	85.3 ± 3.2	96.0 ± 3.2
AD – 18 features	69.1 ± 1.2	85.6 ± 6.8	82.7 ± 4.4	91.2 ± 3.1	100.6 ± 1.6
AD – 16 features	89.6 ± 1.7	94.7 ± 4.9	96.8 ± 2.2	97.9 ± 2.0	97.9 ± 3.0
ADT – 18 features	11.2 ± 3.9	44.7 ± 12.5	79.9 ± 7.3	66.2 ± 5.5	81.1 ± 9.3
ADT – 16 features	40.4 ± 7.7	70.2 ± 3.7	88.2 ± 5.8	85.1 ± 3.9	99.3 ± 3.5

For AD and ADT, the 16-feature analysis regenerated the model-generated anomaly labels using the 16-feature specification; therefore, changes in these two tasks reflect both removal of the timing predictors and the corresponding change in model-generated anomaly labels.

Second-Model Sensitivity

For ADT, strategy rankings were not consistently reproduced between Random Forest and Logistic Regression across the reduced sampling rates. Descriptive Spearman rank agreement ranged from -0.94 to 0.94 across the evaluated rates, indicating that specific per-strategy rankings were sensitive to classifier choice. This sensitivity analysis was performed only for ADT.

Discussion

We compared six AIS data sampling strategies across four vessel behavior detection tasks using the same evaluation pipeline. Our results show that the best sampling strategy depends on the task, and that for two of the four tasks, fishing detection and anomaly detection by vessel type, no strategy reached 95% retention below the full dataset within the rates we tested. Adaptive sampling had the lowest mean retention for pooled anomaly detection at the 1% rate, but the large seed-to-seed variability at this rate prevents treating that ordering as stable. By the 50% rate, Adaptive ($100.5 \pm 1.1\%$) and Temporal ($100.6 \pm 1.6\%$) had nearly identical mean retention with overlapping uncertainty intervals. For ADT, Adaptive had one of the highest mean retention values at the 50% rate, but its interval overlapped Temporal's, so no stable ordering is claimed. This is related to how Adaptive sampling selects data. Adaptive sampling prioritizes pings with large changes

in speed or course, reducing the representation of stable vessel behavior at lower sampling rates. Because pooled anomaly detection depends on learning both normal and unusual behavior, this imbalance leads to weaker performance when very little data is retained. As the sampling rate increases, Adaptive retains enough stable behavior while continuing to preserve high-change pings, which explains its improved performance by the 50% target. When anomaly detection is performed separately by vessel type, less data is available within each vessel group, and Adaptive remains a lower-mean strategy at low sampling rates; by the 50% rate, Adaptive and Temporal have the two highest mean retention values, but ± 1 SD intervals overlap, so this ordering should not be interpreted as stable. These results show that the effectiveness of a sampling strategy depends on both the downstream task and how the ground truth is defined.

Temporal sampling performed differently from the other strategies. It reached the 95% F1 retention threshold at the 50% target for vessel classification and pooled anomaly detection. For anomaly detection by vessel type, Temporal has the highest mean retention at the 50% rate (81.1%), closely followed by Adaptive (79.4%); however, their ± 1 SD intervals overlap, so this ordering should not be interpreted as a stable ranking. Like every other strategy tested, neither reaches the 95% threshold below the full dataset within the rates evaluated here. Temporal is also not uniformly the highest-mean strategy at every rate for this task, since Random and Spatial each have higher mean retention than Temporal at intermediate rates (see Results). Because Temporal sampling retains observations at fixed time windows rather than prioritizing specific vessel behaviors, it preserves observations across a vessel's trajectory. The results suggest that this type of sampling may be useful when reducing AIS data, although Temporal sampling only approximates the target sampling rates because the retained data volume depends on the selected time window.

We also showed that an inconsistent evaluation process can greatly inflate a sampling strategy's performance. For Random sampling on the ADT task, using different train/test splits increased measured F1 retention at the 10% sampling rate from 72.3% to 95.8%. Separately, we found that recomputing lag-dependent features from only the pings retained by each sampling condition, rather than carrying them through from a globally precomputed sequence, substantially reduces measured retention at low sampling rates, particularly for anomaly detection by vessel type. These findings highlight the importance of using the same train/test split and the same feature-computation order when comparing sampling strategies.

Limitations

This study uses one day of AIS data from the Gulf of Mexico, so the results may differ across other geographic regions, ves-

sel populations, seasons, or longer time periods. The anomaly labels were generated by Isolation Forest models rather than real-world labels, so our results measure how well the different sampling strategies reproduce the models' anomaly labels rather than their ability to identify confirmed maritime anomalies. Stratified and Importance sampling both allocate their sampling budget using vessel type, which is also the target label for Vessel Classification and, via the fishing/non-fishing split for Fishing Detection; this gives those two strategies indirect access to label-correlated information that the remaining four strategies do not have, and their relative standing on these two tasks should be interpreted with this asymmetry in mind. Importance sampling used heuristic vessel-type weights rather than weights optimized from the data, which affected how much of the training data it actually retained at some target rates (see Sampling Protocol); different weighting or redistribution approaches could produce different results. The second-model sensitivity check (Logistic Regression versus Random Forest) was performed only for the Anomaly-Detection-by-Type task; whether our findings for the other three tasks are specific to Random Forest was not tested in this study.

Future Work

Future work should evaluate these findings on larger AIS datasets from multiple regions and over longer time periods. Future studies could also evaluate sampling strategies using confirmed real-world anomaly labels, test different Importance sampling weights and redistribution approaches, and extend the second-model sensitivity check performed here for anomaly detection by type to the remaining three tasks, to examine whether the findings remain consistent across different machine learning models.

Conclusion

Overall, our results show that reducing AIS training data can preserve model performance, but no single sampling strategy works best across every task. The sampling strategy and evaluation approach should therefore be selected based on the specific machine learning task and how its ground truth is defined.

Acknowledgments

We thank Mr. Arijit Das, a research advisor based in Monterey, California, with experience in maritime AIS analytics, AI/ML, and large-scale data analysis, for his guidance on the research methodology, AIS data analysis, experimental design, and interpretation of the results. AI-assisted tools, including ChatGPT and Claude, were used to assist with coding, debugging, review of analysis outputs, literature searches,

and spelling and grammar checks. The authors independently reviewed and verified the technical content, data analysis, results, and references for the final manuscript.

References

- 1 International Maritime Organization. International convention for the safety of life at sea (SOLAS), Chapter V, Regulation 19. IMO, London, 2002.
- 2 NOAA Office for Coastal Management, "MarineCadastr.gov AIS Data," National Oceanic and Atmospheric Administration, 2024, <https://marinecadastre.gov/ais/>.
- 3 L. Zhao, G. Shi. A method for simplifying ship trajectory based on improved Douglas–Peucker algorithm. *Ocean Engineering*. Vol. 166, pg. 37–46, 2018, <https://doi.org/10.1016/j.oceaneng.2018.08.005>.
- 4 D. A. Kroodsma, J. Mayorga, T. Hochberg, N. A. Miller, K. Boerder, F. Ferretti, A. Wilson, B. Bergman, T. D. White, B. A. Block, P. Woods, B. Sullivan, C. Costello, B. Worm. Tracking the global footprint of fisheries. *Science*. Vol. 359, no. 6378, pg. 904–908, 2018, <https://doi.org/10.1126/science.aao5646>.
- 5 H. Zhong, X. Song, L. Yang. Vessel classification from space-based AIS data using random forest. *Proceedings of the 2019 5th International Conference on Big Data and Information Analytics (BigDIA)*, Kunming, China, pg. 9–12, 2019, <https://doi.org/10.1109/BigDIA.2019.8802792>.
- 6 Y. Shi, C. Long, X. Yang, M. Deng. Abnormal ship behavior detection based on AIS data. *Applied Sciences*. Vol. 12, pg. 4635, 2022, <https://doi.org/10.3390/app12094635>.
- 7 Z. Wei, X. Xie, X. Zhang. AIS trajectory simplification algorithm considering ship behaviours. *Ocean Engineering*. Vol. 216, art. 108086, 2020, <https://doi.org/10.1016/j.oceaneng.2020.108086>.
- 8 W. Lee, S.-W. Cho. AIS trajectories simplification algorithm considering topographic information. *Sensors*. Vol. 22, no. 18, art. 7036, 2022, <https://doi.org/10.3390/s22187036>.
- 9 C. Tang, H. Wang, J. Zhao, Y. Tang, H. Yan, Y. Xiao. A method for compressing AIS trajectory data based on the adaptive-threshold Douglas–Peucker algorithm. *Ocean Engineering*. Vol. 232, art. 109041, 2021, <https://doi.org/10.1016/j.oceaneng.2021.109041>.
- 10 X. Zhang, S. Zhou. Vessel trajectory data compression algorithm considering critical region identification. *Journal of Advanced Transportation*. Vol. 2023, art. 8831371, 2023, <https://doi.org/10.1155/2023/8831371>.
- 11 T. Zhang, Z. Wang, P. Wang. A method for compressing AIS trajectory based on the adaptive core threshold difference Douglas–Peucker algorithm. *Scientific Reports*. Vol. 14, art. 21408, 2024, <https://doi.org/10.1038/s41598-024-71779-4>.
- 12 Z. Zhou, Y. Zhang, X. Yuan, H. Wang. Compressing AIS trajectory data based on the multi-objective peak Douglas–Peucker algorithm. *IEEE Access*. Vol. 11, pg. 6802–6821, 2023, <https://doi.org/10.1109/ACCESS.2023.3234121>.
- 13 F. Zhu, Z. Ma. Ship trajectory online compression algorithm considering handling patterns. *IEEE Access*. Vol. 9, pg. 70182–70191, 2021, <https://doi.org/10.1109/ACCESS.2021.3078642>.
- 14 F. Han, Y. Liu, H. Tian, J. Li, Y. Tian. A comprehensive framework incorporating deep learning for analyzing fishing vessel activity using automatic identification system data. *ICES Journal of Marine Science*. Vol. 82, no. 2, art. fsae166, 2025, <https://doi.org/10.1093/icesjms/fsae166>.
- 15 Y. Wang, L. Yang, X. Song, X. Li. Ship classification based on random forest using static information from AIS data. *Journal of Physics: Conference Series*. Vol. 2113, art. 012072, 2021, <https://doi.org/10.1088/1742-6596/2113/1/012072>.

- 16 H. Welch, T. Clavelle, T. D. White, M. A. Cimino, J. Van Osdel, T. Hochberg, D. Kroodsma, E. L. Hazen. Hot spots of unseen fishing vessels. *Science Advances*. Vol. 8, no. 44, art. eabq2109, 2022, <https://doi.org/10.1126/sciadv.abq2109>.
- 17 M. Liang, L. Weng, R. Gao, Y. Li, L. Du. Unsupervised maritime anomaly detection for intelligent situational awareness using AIS data. *Knowledge-Based Systems*. Vol. 284, art. 111313, 2024, <https://doi.org/10.1016/j.knosys.2023.111313>.
- 18 L. Xie, T. Guo, J. Chang, C. Wan, X. Hu, Y. Yang, C. Ou. A novel model for ship trajectory anomaly detection based on Gaussian mixture variational autoencoder. *IEEE Transactions on Vehicular Technology*. Vol. 72, no. 11, 2023, <https://doi.org/10.1109/TVT.2023.3284908>.
- 19 D. Nguyen, R. Vadaine, G. Hajduch, R. Garelo, R. Fablet. GeoTrack-Net—A maritime anomaly detector using probabilistic neural network representation of AIS tracks and a contrario detection. *IEEE Transactions on Intelligent Transportation Systems*. Vol. 23, no. 6, pg. 5655–5667, 2022, <https://doi.org/10.1109/TITS.2021.3055614>.
- 20 H. Rong, A. P. Teixeira, C. Guedes Soares. Data mining approach to ship-ping route characterization and anomaly detection based on AIS data. *Ocean Engineering*. Vol. 198, art. 106936, 2020, <https://doi.org/10.1016/j.oceaneng.2020.106936>.
- 21 L. Zhang, Y. Zhu, J. Ren, W. Lu, Y. Yao. A method for detecting abnormal behavior of ships based on multi-dimensional density distance and an abnormal isolation mechanism. *Mathematical Biosciences and Engineering*. Vol. 20, no. 8, pg. 13921–13946, 2023, <https://doi.org/10.3934/mbe.2023620>.
- 22 D.-W. Gao, Y.-S. Zhu, J.-F. Zhang, Y.-K. He, K. Yan, B.-R. Yan. A novel MP-LSTM method for ship trajectory prediction based on AIS data. *Ocean Engineering*. Vol. 228, art. 108956, 2021, <https://doi.org/10.1016/j.oceaneng.2021.108956>.
- 23 B. Murray, L. P. Perera. Ship behavior prediction via trajectory extraction-based clustering for maritime situation awareness. *Journal of Ocean Engineering and Science*. Vol. 7, no. 1, pg. 1–13, 2022, <https://doi.org/10.1016/j.joes.2021.03.001>.
- 24 D. Jiang, G. Shi, N. Li, L. Ma, W. Li, J. Shi. TRFM-LS: Transformer-based deep learning method for vessel trajectory prediction. *Journal of Marine Science and Engineering*. Vol. 11, no. 4, art. 880, 2023, <https://doi.org/10.3390/jmse11040880>.
- 25 X. Chen, C. Wei, G. Zhou, H. Wu, Z. Wang, S. A. Biancardo. Automatic Identification System (AIS) data supported ship trajectory prediction and analysis via a deep learning model. *Journal of Marine Science and Engineering*. Vol. 10, no. 9, art. 1314, 2022, <https://doi.org/10.3390/jmse10091314>.
- 26 H. Arbabkhah, A. Sedaghat, M. Jafari Kang, M. Hamidi. Automatic Identification System-based prediction of tanker and cargo estimated time of arrival in narrow waterways. *Journal of Marine Science and Engineering*. Vol. 12, no. 2, art. 215, 2024, <https://doi.org/10.3390/jmse12020215>.
- 27 J. He, J. Ruan, Y. Tong. Improved AIS data simplification algorithm for extracting typical routes considering motion continuity. *PLOS ONE*. Vol. 21, no. 3, art. e0340021, 2026, <https://doi.org/10.1371/journal.pone.0340021>.
- 28 M. D. Ferreira, J. Campbell, E. Purney, A. Soares, S. Matwin. Assessing compression algorithms to improve the efficiency of clustering analysis on AIS vessel trajectories. *International Journal of Geographical Information Science*. Vol. 37, no. 3, pg. 660–683, 2023, <https://doi.org/10.1080/13658816.2022.2163494>.
- 29 O. Sener, S. Savarese. Active learning for convolutional neural networks: A core-set approach. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.
- 30 B. B. Moser, A. S. Shanbhag, S. Frolov, F. Raue, J. Folz, A. Dengel. A coresets selection of coresets selection literature: Introduction and recent advances. *arXiv:2505.17799*, 2025, <https://doi.org/10.48550/arXiv.2505.17799>.
- 31 B. Mirzasoleiman, J. Bilmes, J. Leskovec. Coresets for data-efficient training of machine learning models. *Proceedings of the 37th International Conference on Machine Learning (ICML)*, PMLR vol. 119, pg. 6950–6960, 2020.
- 32 K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, A. De, R. Iyer. GRAD-MATCH: Gradient matching based data subset selection for efficient deep model training. *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR vol. 139, pg. 5464–5474, 2021.
- 33 C. Coleman, C. Yeh, S. Musmann, B. Mirzasoleiman, P. Bailis, P. Liang, J. Leskovec, M. Zaharia. Selection via proxy: Efficient data selection for deep learning. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.
- 34 K. Wei, R. Iyer, J. Bilmes. Submodularity in data subset selection and active learning. *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, PMLR vol. 37, pg. 1954–1963, 2015.
- 35 M. Toneva, A. Sordoni, R. T. des Combes, A. Trischler, Y. Bengio, G. J. Gordon. An empirical study of example forgetting during deep neural network learning. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.
- 36 M. Paul, S. Ganguli, G. K. Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, 2021.