Detecting Feature Drift by Monitoring Feature-Rank Stability in Data Streams

Benjamin Mensah Dadzie[1][0000−0002−1323−5891] and Piotr Porwik[1][0000−0001−8989−9478]

Institute of Computer Science, University of Silesia in Katowice, Bedzinska 39, Sosnowiec, 41-200, Poland
{benjamin.dadzie, piotr.porwik}@us.edu.pl

Abstract. Data stream mining requires models robust to nonstationarity. We address feature drift, defined as changes in feature-relevance rankings over time. We propose FRDD, a chunk-wise, event-driven detector that monitors feature-rank stability and raises alarms when a sufficient fraction of features violates reference intervals. The method supports supervised (LASSO) and unsupervised (Laplacian Score) ranking. Experiments on synthetic and real streams show that the supervised variant is more conservative, while the unsupervised one is more reactive.

Keywords: feature drift, drift detection, feature ranking, classification

1 Introduction

Many machine-learning systems operate on evolving data streams, where both the input distribution $p(x)$ and the predictive relationship $p(y \mid x)$ may change over time. Classical concept-drift detectors typically monitor prediction error or related statistics. Although effective, such approaches often depend on timely labels and may generate frequent alarms. An alternative is to monitor changes in feature relevance over time. In this work, feature drift is understood as instability in feature-importance rankings computed on consecutive chunks. We do not assume that feature drift is equivalent to concept drift. Instead, we investigate whether feature-rank monitoring can provide a practical alarm signal for downstream model adaptation.

Most concept drift detectors monitor prediction error or related statistics, e.g., DDM, EDDM, and ADWIN [10,2,4]. Other approaches, such as CUSUM and Page–Hinkley, rely on sequential change detection [14]. While effective, these methods typically require labeled data and may be unstable when labels are delayed. Unsupervised methods aim to detect distribution changes without labels, e.g., HDDM and RDDM [3]. However, they focus on aggregate statistics and do not explicitly model feature relevance. Recent work highlights the role of feature drift [12,13], suggesting that changes in feature importance can signal evolving concepts. At the same time, feature ranking methods such as LASSO and Laplacian Score [8,11] are typically used in static settings. In contrast, FRDD monitors the stability of feature rankings over time, providing a label-independent drift signal complementary to classical approaches.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

1.1 Proposed contribution

Concept drift refers to changes in the conditional distribution $p(y \mid x)$, which may degrade predictive performance and require model adaptation [1]. We further distinguish: *Data drift*: changes in the input distribution, e.g., $p_t(x) \neq p_{t+1}(x)$ and *Feature drift*: changes in the relative predictive relevance of features over time. If π_t and π_{t+1} denote feature rankings at times t and $t+1$, feature drift is observed as instability of these rankings.

We introduce **FRDD** (Feature Ranking-Based Drift Detector), a chunk-wise, event-driven method that detects feature drift by monitoring violations of feature-rank stability. A reference chunk is split into N sub-chunks, feature rankings are computed, and feature-wise acceptance intervals are estimated from within-reference rank variability. An alarm is raised when the fraction of monitored features whose ranks fall outside these intervals exceeds a threshold. The reference is updated only after detected drift.

The main contributions are:

- A modular, interval-based detector supporting supervised and unsupervised ranking.
- An event-driven reference update policy to minimize redundant computations.
- A unified evaluation protocol with consistent adaptation.
- Empirical validation on synthetic and real streams, demonstrating the complementary nature of FRDD variants.

2 Background

Feature drift refers to changes over time in the relative predictive relevance of input features. In practice, there appears to be instability in the feature rankings computed across successive chunks. In this paper, rank instability is used as a drift signal complementary to classical error-based detectors, especially when labels are delayed, or unavailable [12]

2.1 LASSO for supervised feature drift detection

LASSO is an ℓ_1-regularized linear model widely used for sparse feature selection [8]. In FRDD, it is used only to produce a supervised feature ranking within each chunk:

$$\min_{\beta,\beta_0}\Big\{\|\mathbf{y} - \beta_0\mathbf{1} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1\Big\}, \tag{1}$$

where $\mathbf{X}$ is the feature matrix, $\mathbf{y}$ is the target vector, and λ controls sparsity. Features are standardized within each chunk, and importance is quantified by $|\beta_i|$. In the experiments, we use a fixed $\lambda = 0.001$.

2.2 Laplacian Score for unsupervised feature drift detection

The Laplacian Score is an unsupervised filter method that ranks features according to how well they preserve local neighborhood structure [11]. In FRDD,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

it provides a label-free ranking computed independently for each chunk. For feature f_r,

$$LS(f_r) = \frac{\sum_{i,j}(f_{r,i} - f_{r,j})^2 K_{i,j}}{\mathrm{Var}(\mathbf{f}_r)}, \tag{2}$$

where $K_{i,j}$ encodes instance similarity:

$$K_{i,j} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), & \text{if } \mathbf{x}_j \text{ is among the } k \text{ nearest neighbors of } \mathbf{x}_i, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

We set $k = 5$ and assign ranks in ascending order of LS.

2.3 Mathematical foundations of FRDD

FRDD processes the stream in chunks of size U. Its main parameters are the number of reference sub-chunks N, the ranking procedure $R(\cdot)$, the number of monitored features m, the interval width k, and the decision threshold τ.

Let $\{(x_t, y_t)\}_{t\geq 1}$ denote a data stream, where $x_t \in \mathbb{R}^d$ and y_t is optional. The stream is processed in chunks. The l-th chunk is defined as:

$$C_l = \{(x_{l,1}, y_{l,1}), \ldots, (x_{l,U}, y_{l,U})\}. \tag{4}$$

Here, l indexes chunks, $l \in \{1, \ldots, U\}$ indexes instances within a chunk, and each $x_{l,j} \in \mathbb{R}^d$ is a d-dimensional feature vector whose components are indexed by $i \in \{1, \ldots, d\}$.

Ranking procedure. Given a chunk C, the ranking procedure $R(\cdot)$ returns a score vector $s = R(C) \in \mathbb{R}^d$. We convert it into a rank vector $\pi = \mathrm{Rank}(s)$, where $\pi(i) \in \{1, \ldots, d\}$ and smaller values indicate higher importance. Ties are broken by feature index. We write π_t for the whole test chunk C_t and $\pi_{r,n}$ for the n-th reference sub-chunk $C_{r,n}$.

Reference chunk splitting. FRDD maintains a reference chunk indexed by r. Only the current reference chunk C_r is split into N non-overlapping sub-chunks of size $W = U/N$:

$$C_{r,n} = \{(x_{r,(n-1)W+1}, y_{r,(n-1)W+1}), \ldots, (x_{r,nW}, y_{r,nW})\}, \quad n \in \{1, \ldots, N\}. \tag{5}$$

For each reference sub-chunk we have, $s_{r,n} = R(C_{r,n})$, $\pi_{r,n} = \mathrm{Rank}(s_{r,n})$.

For each feature i we summarize the within-reference rank variability as

$$\mu_r(i) = \frac{1}{N}\sum_{n=1}^{N} \pi_{r,n}(i), \qquad \sigma_r(i) = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}\big(\pi_{r,n}(i) - \mu_r(i)\big)^2}. \tag{6}$$

Using the interval width parameter $\kappa > 0$, FRDD defines the acceptance interval

$$I_r(i) = \big[\mu_r(i) - \kappa\sigma_r(i),\ \mu_r(i) + \kappa\sigma_r(i)\big]. \tag{7}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

Top-m monitored features. FRDD monitors only the top-m features selected from the current reference. Let

$$S_r = \{\, i : \ \mu_r(i) \text{ is among the } m \text{ smallest over } i = 1, \ldots, d \,\}. \tag{8}$$

In the experiments, we use $m = d$ to avoid additional tuning.

Test chunk evaluation. For each subsequent test chunk C_t ($t > r$), FRDD computes

$$s_t = R(C_t), \qquad \pi_t = \mathrm{Rank}(s_t), \tag{9}$$

and defines the violation indicator for each monitored feature $i \in S_r$ as

$$v_t(i) = \mathbf{1}\big[\pi_t(i) \notin I_r(i)\big]. \tag{10}$$

Decision rule. The normalized drift score is $D_t = \frac{1}{m} \sum_{i \in S_r} v_t(i)$. Feature drift is declared when $D_t \geq \tau$.

Hyperparameters. FRDD is controlled by U, N, m, κ, and τ. In the experiments, we use $N = 10$, $\kappa = 1$, $\tau = 0.5$, and choose U so that $W = U/N \geq 50$.

2.4 Evaluation protocol

FRDD itself does not require a classifier; drift is detected solely from feature-ranking statistics. A classifier is used only to measure the downstream effect of alarms under a fixed adaptation policy. We use Random Forest (RF) [6] as a common baseline learner for all detectors.

For synthetic streams with known drift locations, alarms are evaluated using a tolerance window of size ϵ around each ground-truth drift point g. For abrupt drifts, we set $g = t_e$, where t_e denotes the drift point. For gradual and incremental drifts, we use the midpoint $g = (t_s + t_e)/2$, where t_s and t_e denote the start and end of the transition interval. For chunk-based methods such as FRDD, an alarm at chunk index t is mapped to the midpoint instance of chunk C_t.

We apply one-to-one matching between drifts and alarms: each drift can be matched to at most one alarm and each alarm to at most one drift. An alarm is counted as a true positive (TP) if it is the first unmatched alarm within the tolerance window of an unmatched drift. Remaining alarms are counted as false positives (FP), and unmatched drifts as false negatives (FN).

3 Input data preparation

We evaluate the methods on synthetic and real data streams. Synthetic streams are generated in MOA [5] using the Hyperplane Generator under four drift regimes: abrupt, gradual, incremental, and recurring. Unless stated otherwise, each stream contains 100,000 instances, $d = 20$ features, and two balanced classes. For each drift type, we generate 20 independent realizations and report average results. *Abrupt*: Drift at the 50k-th instance; drift duration: 1 instance. *Gradual*: Drift at the 50k-th instance; drift duration: 20k instances. *Recurring*:

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

Algorithm 1 FRDD with event-driven reference update

Input: Stream in chunks of size U; number of reference sub-chunks N (thus $W = U/N$); ranking procedure $R(\cdot)$; top-m monitored features; interval width κ; decision threshold τ.

Output: Drift alarms at chunk indices.

Initialization
$r \leftarrow 1$; read C_r (size U) and split into $\{C_{r,n}\}_{n=1}^{N}$
for $n = 1$ **to** N **do**
  $s_{r,n} \leftarrow R(C_{r,n})$; $\pi_{r,n} \leftarrow \mathrm{Rank}(s_{r,n})$
end
Compute $\mu_r(i)$ and $\sigma_r(i)$ for all $i = 1, \ldots, d$ from $\{\pi_{r,n}\}_{n=1}^{N}$
Compute $I_r(i) = [\mu_r(i) - \kappa\sigma_r(i),\ \mu_r(i) + \kappa\sigma_r(i)]$ for all $i = 1, \ldots, d$
Compute S_r (top-m) as the set of m smallest values of $\mu_r(i)$

for $t \leftarrow 2$ **to** ∞ **do**
  if *chunk* C_t *is not available* **then**
    break
  end
  Evaluate test chunk $s_t \leftarrow R(C_t)$; $\pi_t \leftarrow \mathrm{Rank}(s_t)$
  $V \leftarrow 0$
  foreach $i \in S_r$ **do**
    if $\pi_t(i) \notin I_r(i)$ **then**
      $V \leftarrow V + 1$
    end
  end
  $D_t \leftarrow V/m$
  if $D_t \geq \tau$ **then**
    Raise drift alarm at chunk t *Update reference* $r \leftarrow t$ Split C_r into $\{C_{r,n}\}_{n=1}^{N}$
    for $n = 1$ **to** N **do**
      $s_{r,n} \leftarrow R(C_{r,n})$; $\pi_{r,n} \leftarrow \mathrm{Rank}(s_{r,n})$
    end
    Recompute $\mu_r(i)$, $\sigma_r(i)$, $I_r(i)$ and S_r
  end
end

Alternation between concepts every 25k instances; drift duration: 1 instance. *Incremental*: The modification weight changes by 0.001 with each instance, with a 10% probability of reversing the change direction.

Real-world streams are taken from public repositories and common stream-mining benchmarks [9,15]. Their characteristics are summarized Table 1. All experiments were conducted in Python 3.8. Code and scripts are publicly available.[1]

Table 1: Characteristics of real datasets (#Inst/#Attrs/#Cls).

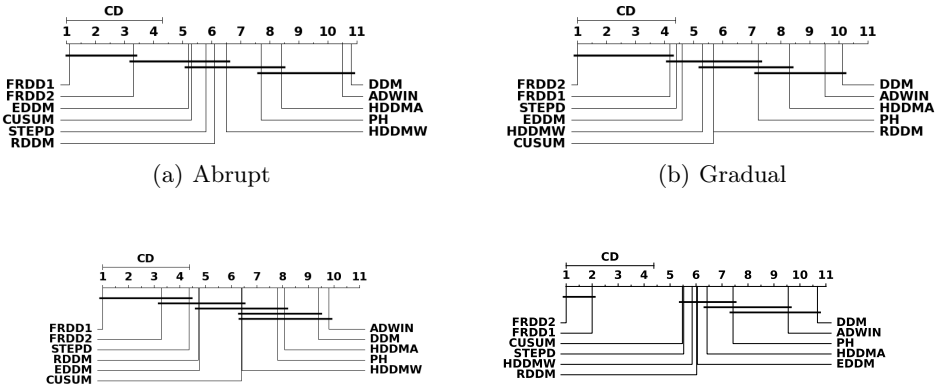
Dataset	**Airlines**	**Ozone**	**Electricity**	**CovType**
#Inst/#Attrs/#Cls	539,383/7/2	2,534/72/2	45,312/8/2	218,513/54/2

[1] `https://github.com/ZSKPP/drift`

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

4 Experimental evaluation: synthetic datasets

FRDD operates on chunks of size U. For synthetic streams, we set $U = 5{,}000$ and average results over 20 runs. We report ACC, complemented by TP/FP statistics (Table 2). Statistical significance is evaluated following the Demšar protocol [7], using the Friedman test and the Nemenyi post-hoc test. The corresponding critical difference diagrams for ACC are presented in Fig. 1.



(a) Abrupt (b) Gradual

(c) Incremental (d) Recurring

Fig. 1: Critical difference diagrams for ACC obtained with the Nemenyi post-hoc test on synthetic data with known drift points.

Table 2: Detection results (TP/FP) for different drift types.

Detector	Abrupt		Gradual		Incremental		Recurring	
	TP	**FP**	**TP**	**FP**	**TP**	**FP**	**TP**	**FP**
Oracle	1.0	0.0	1.0	0.0	1.0	0.0	3.0	0.0
FRDD1	0.80	**0.60**	0.30	**1.45**	0.20	**1.70**	0.60	**0.90**
FRDD2	0.85	**0.60**	0.20	3.00	0.20	**4.00**	0.60	5.20
CUSUM	1.00	2.80	0.55	3.95	0.30	7.70	2.80	10.00
PH	1.00	1.80	0.10	3.20	0.30	6.25	3.00	1.05
HDDM-A	1.00	1.95	0.15	3.10	0.20	6.00	2.85	2.15
HDDM-W	1.00	3.75	0.55	5.20	0.30	9.05	2.50	5.50
RDDM	1.00	10.35	0.60	10.55	0.45	12.30	2.40	7.60
DDM	0.07	8.45	0.65	6.20	0.35	4.60	0.95	2.60
EDDM	1.00	17.10	0.55	16.05	0.75	18.10	2.05	13.85
ADWIN	0.90	9.10	0.90	2.55	0.35	4.65	1.90	1.10
STEPD	1.00	13.05	0.75	13.65	0.70	13.95	2.60	11.65

Table 2 shows mean TP/FP over 20 runs. FRDD1 is more conservative (lower FP), while FRDD2 is more sensitive (more alarms), especially for gradual and incremental drifts; thus, FRDD1 favors precision and FRDD2 sensitivity.

5 Experimental evaluation: real datasets

We evaluate FRDD on the real-world streams listed in Table 1. As true drift locations are unknown, methods are compared within the same RF-based adaptation pipeline. Figure e 2 reports ACC and the number of detections.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

Table 3: Wilcoxon signed-rank test based on the two-tailed W^+ statistic for average ACC values on synthetic data. Table entries are p-values.

FRDD1 vs.	CUSUM	PH	DDM	EDDM	HDDM$_A$	HDDM$_W$	RDDM	ADWIN	STEPD
Abrupt	$= 0.001$	< 0.001	< 0.001	$= 0.004$	< 0.001	$= 0.002$	< 0.001	< 0.001	$= 0.001$
Gradual	$= 0.016$	$= 0.004$	< 0.001	$=\mathbf{0.059}$	$= 0.002$	$= 0.001$	$= 0.018$	$= 0.001$	$=\mathbf{0.052}$
Incremental	$=\mathbf{0.330}$	$=\mathbf{0.227}$	$= 0.048$	$=\mathbf{0.784}$	$=\mathbf{0.185}$	$=\mathbf{0.374}$	$=\mathbf{0.538}$	$=\mathbf{0.131}$	$=\mathbf{0.627}$
Recurring	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001

FRDD2 vs.	CUSUM	PH	DDM	EDDM	HDDM$_A$	HDDM$_W$	RDDM	ADWIN	STEPD
Abrupt	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001
Gradual	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	$= 0.001$	< 0.001	$= 0.001$	$= 0.001$
Incremental	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.002	$= 0.004$
Recurring	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001

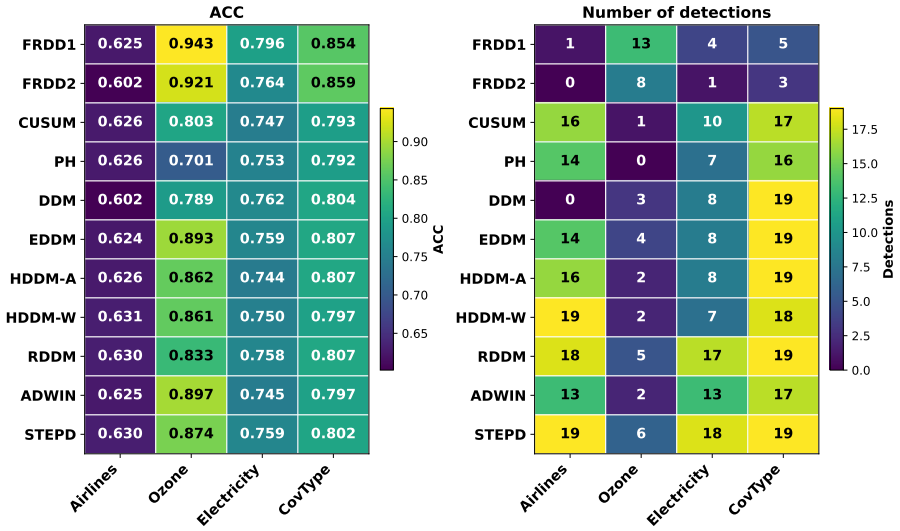


Fig. 2: Heatmaps of real-world performance under detector-triggered adaptation. Left: classification accuracy (ACC). Right: number of detected drifts. Rows correspond to detectors and columns to datasets.

6 Solution complexity

Assume a stream of L_c chunks, each of size U. FRDD computes one ranking per chunk at cost $A(U, d)$. After each detected drift, it rebuilds the reference from N sub-chunks of size $W = U/N$, giving an additional cost $N\,A(W, d)$. Therefore,

$$\mathcal{O}(L_c\,A(U, d) + L_d\,N\,A(W, d))\,. \tag{11}$$

Since N is constant and $L_d \leq L_c$, the total complexity is $\mathcal{O}(L_c\,A(U, d))$.

Ranking cost. LASSO (coordinate descent): $A(U, d) = \mathcal{O}(TUd)$, $A(W, d) = \mathcal{O}(TWd)$. Laplacian Score (k-NN graph): $A(U, d) = \mathcal{O}(U^2 d)$, $A(W, d) = \mathcal{O}(W^2 d)$.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27

7 Conclusion

We introduced FRDD, a chunk-based event-driven detector that identifies feature drift by monitoring feature-rank stability. The method supports both supervised (FRDD1, LASSO) and unsupervised (FRDD2, Laplacian Score) rankings. Experiments on synthetic and real-world streams show that rank-based monitoring provides a useful alarm signal under a unified adaptation policy. The study is primarily empirical. Hyperparameter sensitivity analysis, and real-world evaluation with drift annotations remain future work. *Reproducibility.* All data and source code used in this study, including the scripts for running the experiments and generating the reported tables and figures, are publicly available at `https://github.com/ZSKPP/drift`.

References

1. Agrahari, S., Singh, A.K.: Concept drift detection in data stream mining : A literature review. Journal of King Saud University - Computer and Information Sciences **34**(10, Part B), 9523–9540 (2022)
2. Baena-Garcia, M., Campo-Avila, J., Bifet, A., Gavald, R., Morales-Bueno, R.: Early drift detection. Advances in Aritificial Intelligence, Lecture Notes Artificial Intelligence **3171**, 286–295 (2006)
3. Barros, R.S., Cabral, D.R., Gonçalves, P.M., Santos, S.G.: Rddm: Reactive drift detection method. Expert Systems with Applications **90**, 344–355 (2017)
4. Bifet, A., Gavaldà, R.: Learning from time-changing data with adaptive windowing. Proceedings of the 7th SIAM International Conference on Data Mining pp. 443–448 (2007)
5. Bifet, A., Holmes, G., Kirkby, R., Pfahringer, B.: Moa: Massive online analysis. J. Mach. Learn. Res. **11**, 1601–1604 (2010)
6. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
7. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7**(1), 1–30 (2006)
8. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. The Annals of Statistics **32**(2) (2004)
9. Frank, A., Asuncion, A.: http://archive.ics.uci.edu/ml. UCI machine learning repository: (2010)
10. Gama, J., Medas, P., Castillo, G., Rodrigues, P.: Learning with drift detection. Advances in Artificial Intelligence - SBIA 2004 pp. 286–295 (2004)
11. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. Proceedings of the 18th International Conference on Neural Information Processing Systems p. 507–514 (2005)
12. Hinder, F., Vaquet, V., Hammer, B.: Feature-based analyses of concept drift. Neurocomputing **600**, 127968 (2024)
13. Nguyen, H.L., Woon, Y.K., Ng, W.K., Wan, L.: Heterogeneous ensemble for feature drifts in data streams. Advances in Knowledge Discovery and Data Mining pp. 1–12 (2012)
14. Page, E.S.: Continuous inspection schemes. Biometrika **41**(1/2), 100–115 (1954)
15. Souza, V.M.A., dos Reis, D.M., Maletzke, A.G., Batista, G.E.A.P.A.: Challenges in benchmarking stream learning algorithms with real-world data. Data Min. Knowl. Discov. **34**(6), 1805–1858 (2020)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_27