DETECTING CONCEPT DRIFT IN DATA STREAMS USING INSTRUMENTAL VARIABLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Concept drift is a common phenomenon in real-world data streams that requires models to adapt to changing data distributions. Yet, most concept drift detectors raise flags according to changes in observable distributions or predictive error rates, often monitoring proxy signals, and requiring continuous access to ground-truth labels. In addition, they do not provide insight into why or how the data-generating process changed. Although recent work frames concept drift as changes in underlying causal mechanisms, identifying these mechanisms shifts from observational data is challenging due to unobserved confounding. In this work, we track mechanism changes in continuous data streams using Instrumental Variables (IVs). Specifically, we introduce an online IV monitoring framework that assumes an initial causal graph to identify valid instruments and continuously estimates their associated causal effects using the Wald estimator. We demonstrate two complementary applications of this framework. First, the Instrumental Variable-monitored Drift Detector (IVDD) compares successive IV estimates to detect and localize changes in monitored causal relationships without ground-truth labels. Second, the Instrumental Variable-monitored Adaptive Tree (IVAT) incorporates local IV-based causal-effect estimates into online tree learning, using causal-effect heterogeneity to identify relevant subpopulations and local monitoring to guide targeted adaptation. Empirical evaluations on synthetic structural causal models and real-world streams demonstrate that our IV-monitoring approaches pinpoint changes in causal mechanisms, achieving competitive performance against state-of-the-art drift detectors in delayed-feedback scenarios. Source code is available on GitHub[1].

1 INTRODUCTION

The ability of Machine Learning (ML) models to adapt to changing distributions is an important characteristic under non-stationary data streams. Among many adaptation strategies, concept drift detection has long played a central role by signaling changes that may require a learner to update (Lu et al., 2019; Alencar et al., 2026). Most drift detectors, however, operationalize concept drift through changes in observational distribution or predictive performance, typically estimated from finite data windows (Tosi & Theobald, 2024; Alencar et al., 2026). Recent work has questioned whether such operational definitions properly capture the underlying notion of concept drift, highlighting how finite-window comparisons can make drift detection dependent on the choice of window and statistic being monitored (Gower-Winter et al., 2026).

We argue, however, that the issue is not the use of finite windows, but rather, *which property of the data-generating process is being compared across windows*. A distribution-based detector flags whether the observed distribution has changed (Agrahari & Singh, 2024), while error-based detectors flag whether performance has changed (Barros et al., 2017; Tosi & Theobald, 2024). These quantities provide useful evidence of change, but neither necessarily identifies the mechanism responsible for it. A change in an observed distribution, for example, may result from changes in exogenous sources while the relationships among variables remain invariant. For instance, in a stream predicting hospital admissions from respiratory symptoms, a seasonal flu outbreak may increase the number of symptomatic patients without changing the underlying disease mechanism. Conversely, a viral mutation that changes the relationship between symptoms and hospitalization would constitute a

[1] Source code available in supplementary material during review.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

change in the corresponding causal mechanism. These distinctions in data streams are mentioned in (Barboza et al., 2026b), which provides a causal perspective for defining concept drift. This perspective motivates a different question about concept drift: *rather than monitoring whether an observable distribution changes, can we directly monitor properties of the causal relationships that generate the observations?* One natural property to monitor is the stability of causal effects. Consider a treatment variable W and an outcome T. We hypothesize that when the causal mechanism relating W to T admits a stable and identifiable causal effect, a stable cause–effect relationship is evidence that the concept is stable. In contrast, a change in this causal effect, under suitable assumptions, is evidence that the causal relationship has changed. The challenge is that causal effects are usually unidentifiable from observational associations alone, especially in the presence of unobserved confounding.

This motivates the use of Instrumental Variables (IVs), which can identify causal effects even in the presence of unobserved confounding under appropriate assumptions (Angrist et al., 1996; Lousdal, 2018). An instrument Z affects the treatment W, is independent of unobserved confounders of W and T, and affects T only through W. Under these assumptions, the causal effect can be identified by the Wald estimator Therefore, we hypothesize that IVs are a natural basis for monitoring whether a causal relationship remains invariant as a data stream evolves.

In this work, we introduce an online Instrumental Variable (IV) monitoring framework designed to track the stability of identifiable causal effects over time. Given a known or estimated causal graph, our framework identifies valid IV structures and continuously estimates their associated causal effects using the Wald estimator across sliding windows. While IVs provide an established means of identifying causal effects, their use as a continuously monitored signal for detecting concept drift has received less attention. Our framework operationalizes this idea by treating the estimated causal effects within an IV structure as a time-varying quantity. Because multiple IV structures can be associated with different parts of a causal graph, the framework can also provide information about *which causal relationship* underlies the detected change. We evaluate this framework on synthetic and real-world data streams and further investigate its use for adaptive online learning.

Our main contributions are as follows: *i) IV Monitoring Framework:* we formulate a causal approach to concept drift that monitors the stability of cause–effect relationships rather than observable distributions or predictive performance; *ii) Instrumental Variable-monitored Drift Detector (IVDD):* we present label-free detector that applies statistical hypothesis testing to successive IV estimates. Intrumental Variable-monitored Drift Detector (IVDD) does not require ground-truth labels and associates detected changes with the monitored causal relationships in the corresponding IV structure; *iii) Online Causal Adaptive Tree:* we extend the monitoring framework into an adaptive tree-based model, utilizing sequential Wald estimates as a dynamic node-splitting criterion to isolate heterogeneous causal effects, which we call IV-monitored Adaptive Tree (IVAT).

2 BACKGROUND

Notation We denote by $\mathbf{x} = (X_1, \ldots, X_d) \in \mathcal{X} \subseteq \mathbb{R}^d$ a d-dimensional feature vector, and by $y \in \mathcal{Y}$ its corresponding class label. A data stream is a sequence of labeled samples arriving over time, $S = (\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots$, such that $(\mathbf{x}_t, y_t)$ is the data sample arriving at time t. We assume that the data are generated by an underlying causal mechanism $\mathcal{M}$ that represents the cause–effect relationships among the variables $\{X_1, \ldots, X_d, y\}$. At each time step t, this mechanism induces a joint probability distribution $P^{(t)}(\mathbf{x}, y)$, from which the sample $(\mathbf{x}_t, y_t)$ is drawn.

Structural Causal Models A Structural Causal Model (SCM) $\mathcal{M}$ is composed of a set of d nodes $V = \{X_1, \ldots, X_d\}$, such that each node X_i is mapped by a structural assignment $X_i = f_i(pa_i, U_i)$, such that $pa_i \subseteq V \setminus \{X_i\}$ denotes the set of parents, or causes, of X_i, and U_i is exogenous noise.

We view concept drift as any shift in the structural mechanisms that affect a variable, as in (Barboza et al., 2026b). Thus, we aim to detect shifts in such cause–effect relationships among variables, rather than solely on the joint probability $P(\mathbf{x}, y)$, as most definitions of *concept drift* consider. In other words, we denote a change in how a variable affects another as a *concept drift*.

To measure cause–effect relationships, the go-to approach is to perform randomized controlled trials or rely on interventions via do-calculus. However, in observational data streams, the presence of

unobserved confounders (i.e., hidden variables affecting both the features and the target) biases standard statistical estimators. This confounding makes true causal mechanisms indistinguishable from spurious correlations, necessitating alternative approaches for causal discovery and drift detection.

Concept drift A concept drift occurs whenever the data-generating distribution changes over time, that is, whenever

$$P^{(t_i)}(\mathbf{x}, y) \neq P^{(t_j)}(\mathbf{x}, y)$$

for some $t_i \neq t_j$. In this work, we approach concept drift as changes in causal mechanisms in the underlying generating process of datasets, modeled by an SCM $\mathcal{M}$, the same perspective adopted in (Barboza et al., 2026b).

Instrumental Variables When unobserved confounding is present, IVs provide a powerful framework for estimating causal effects from observational data Angrist et al. (1996); Lousdal (2018). An IV Z is a variable that affects an endogenous feature W but has no independent effect on the outcome T other than through W. Formally, a valid instrument Z must satisfy three conditions:

1. **Relevance:** The instrument must have a direct causal effect on the endogenous feature, meaning $Z \not\perp\!\!\!\perp W$.
2. **Exclusion Restriction:** The instrument affects the outcome only through its influence on W. Therefore, $Z \perp\!\!\!\perp T \mid (W, U)$.
3. **Unconfoundedness:** The instrument is independent of any unobserved confounders U that affect both W and T, such that $Z \perp\!\!\!\perp U$.

By isolating the variance of W attributable to Z, we can estimate the unconfounded causal effect of W on T. We illustrate a Directed Acyclic Graph (DAG) structure containing an IV and its assumptions in Figure 1. In data streams, monitoring the statistics of this IV-based causal estimate over time allows us to directly track the structural mechanism $W \rightarrow T$, which underpins our proposed IV framework and the applications we present in Section 4.

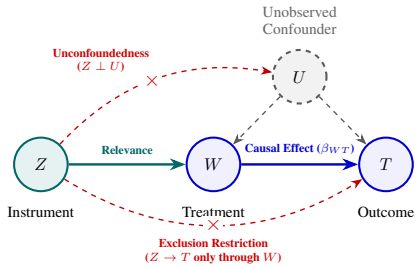


Figure 1: A DAG illustrating an IV structure. Z is the instrument, W the treatment variable, T the outcome, or local target, and U an unobserved confounder.

3 Related Work

Incremental Learners and Causal Tree Online decision trees, such as the Hoeffding Tree (HT) (Domingos & Hulten, 2000), incrementally expand their structure by continuously evaluating statistical bounds, e.g., Hoeffding's bound. While adaptive trees update or prune branches when predictive accuracy degrades, they rely strictly on observable statistical associations and demand continuous ground-truth supervision (Bifet & Gavalda, 2009; Manapragada et al., 2018; Heyden et al., 2024; Nowak Assis et al., 2025). In contrast, causal trees and causal forests (Athey & Imbens, 2016; Athey et al., 2018) extend recursive partitioning to the causal domain by optimizing splits for treatment effect heterogeneity rather than predictive purity. However, these causal tree architectures are formulated strictly for static, offline settings. To operate in dynamic streams, models must account for the Sparse Mechanism Shift (SMS) hypothesis (Schölkopf et al., 2021), which posits that distribution shifts typically alter only a small subset of causal mechanisms while leaving the remainder invariant. This motivates localized adaptation: rather than resetting the global model upon drift, an adaptive causal learner should localize the structural shift and update the subtrees governing the altered mechanisms.

Supervised Drift Detection Supervised drift detectors rely on ground-truth labels to monitor variations in a model's error rate. Early works like DDM (Gama et al., 2004) and its extensions, EDDM (Baena-García et al., 2006) and RDDM (Barros et al., 2017), track statistical deviations in sequential error rates or intervals between mispredictions. Similarly, STEPD (Nishida & Yamauchi, 2007), ADWIN (Bifet & Gavaldà, 2007), SeqDrift (Sakthithasan et al., 2013; Pears et al., 2014), and OPTWIN (Tosi & Theobald, 2024) employ statistical hypothesis testing within sliding windows or dynamic batches to detect accuracy divergence. Alternatively, MDDM (Pesaranghader et al., 2018) applies time-weighted statistical bounds to track evolving streams. While effective, these methods are limited by their reliance on label availability, which is often scarce in real-world environments, and they cannot determine *which* causal relationships in the data-generating process have changed.

Unsupervised Drift Detection Unsupervised detectors bypass the need for true labels by monitoring alternative signals or feature distributions. Model-based approaches track predictive anomalies, such as the divergence between student and teacher models in STUDD (Cerqueira et al., 2023), fluctuations in neural network uncertainty, as in UDD (Baier et al., 2022), or temporal shifts in predicted label correlations in LD3 (Gulcan & Can, 2023). In contrast, data-centric methods such as DD-SCC and DD-KRC (Agrahari & Singh, 2024) use rank-correlation statistics to detect distributional changes, while MMDEW (Kalinke et al., 2025) applies the Maximum Mean Discrepancy to compare distributions over exponential windows for online change detection. The mentioned methods may be more effective in environments with limited label availability and can localize which feature has suffered a drift in the covariate $P(\mathbf{x})$, but still monitor surface-level proxies or marginal distributions rather than the underlying causal mechanisms.

Our work bridges these paradigms by introducing an online IV monitoring framework. Rather than relying on surface-level statistical proxies for monitoring, our framework leverages a known or discovered graph to isolate valid IV triplets and track the stability of their causal effects using the Wald estimator (Angrist et al., 1996). This yields two components: 1) IVDD, a drift detector that performs hypothesis testing on sequential Wald estimates to identify and localize mechanism drift without labels, and 2) IVAT, an online adaptive tree that partitions feature space by causal heterogeneity while employing subtree adaptation to the localized subpopulation undergoing drift.

4 Monitoring Mechanism Drift with Instrumental Variables

In this work, we introduce an IV monitoring framework designed to track the stability of identifiable causal effects over time. The framework assumes that the underlying data generation process can be represented by a DAG, which can be known or estimated, e.g., through causal discovery algorithms. From this DAG, the monitor identifies valid IV structures.

Under unobserved confounding, the causal effect of W on T cannot generally be identified from observational data alone (Pearl, 1995; Angrist et al., 1996). However, if a valid IV is available, it becomes possible to estimate the causal effect of a subpopulation that is influenced by the instrument (Angrist et al., 1996). The key hypothesis behind our framework is that, as long as the structural mechanism relating treatment and outcome remains unchanged, the causal effect identified by a valid IV should remain stable over time. Therefore, an IV monitor provides a continuous estimate of causal relationships that can be leveraged for various adaptive learning tasks.

4.1 Instrument Discovery

The first stage of the monitor is to identify all valid IV structures in a graph. Given a DAG $\mathcal{G} = (V, E)$, the framework searches for all triplets (Z, W, T), where Z is a valid instrument for estimating the causal effect of a treatment W on an outcome T. For each candidate treatment node W, we consider every direct child of W as a potential local outcome T. Candidate instruments are then evaluated according to the graphical conditions implied by the IV assumptions (Angrist et al., 1996).

The first requirement is *instrument relevance*, which states that the instrument must be associated with the treatment. Graphically, this requires that Z and W are not d-separated, i.e., $Z \not\perp\!\!\!\perp W$. To verify the exclusion restriction, IVDD constructs a mutilated graph by removing all outgoing edges from the treatment variable, $\mathcal{G}_W = (V, E) \backslash \{W \to T\}$, following Pearl's intervention semantics

(Pearl, 1995). In this mutilated graph, a valid instrument must be d-separated from the outcome, i.e., $Z \perp\!\!\!\perp T \mid \mathcal{G}_W$.

If both conditions are satisfied, the triplet (Z, W, T) is added to the set of monitored IV structures, $\mathcal{I} = \{(Z_1, W_1, T_1), \ldots, (Z_m, W_m, T_m)\}$. Each discovered triplet corresponds to an independent causal mechanism that will subsequently be monitored. We present the pseudocode for this process in Algorithm 1, in Appendix A.

4.2 Estimating Causal Effects

For any discovered IV structure (Z, W, T), the framework continuously estimates the local causal effect. Under the standard IV assumptions (see Section 2), the IV estimator is (Angrist et al., 1996; Lousdal, 2018):

$$\hat{\beta} = \frac{E[T \mid Z = 1] - E[T \mid Z = 0]}{E[W \mid Z = 1] - E[W \mid Z = 0]}, \tag{1}$$

and for continuous variables,

$$\hat{\beta} = \frac{Cov(Z, T)}{Cov(Z, W)}. \tag{2}$$

Given a sliding window w_t containing the most recent observations, our framework computes the corresponding estimate β_t. Under a stationary causal mechanism, successive estimates of the causal effect are expected to remain statistically consistent, apart from sampling variability. Conversely, changes in the underlying causal mechanism should manifest as statistically significant changes in the estimated causal effect. To ensure mechanism localization, we restrict candidate outcomes to the direct children of the treatment variable. This allows each monitor to correspond to a small set of nodes in the causal graph, simplifying the interpretation of detected changes.

4.3 Instrumental Variable-monitored Drift Detector

We instantiate the monitoring framework into an unsupervised drift detector, IVDD, to track mechanism evolution in data streams. After discovering the set of IV structures $\mathcal{I}$, IVDD initiates an independent monitor per structure. Consequently, each monitor tracks the stability of a single identifiable causal mechanism throughout the data stream, which allows us to pinpoint *where* in the causal graph a mechanism change occurred.

For each monitor, IVDD maintains two windows of observations: 1) a reference window R_i, representing the current baseline concept, and 2) a sliding window w_{ti}, containing the most recent observations. Each incoming instance contributes only to the variables associated with the corresponding IV structure. Formally, each monitor $\mathcal{Q}_i$ maintains the tuple:

$$\mathcal{Q}_i = (R_i, w_{ti}, \hat{\beta}_i, \hat{v}_i),$$

where R_i denotes the reference window, w_{ti} the current sliding window of monitor i, $\hat{\beta}_i$ the baseline causal effect, and $\hat{v}_i$ its estimated associated variance. The current window is continuously updated as new observations arrive, whereas the reference window, defined at the beginning of the stream or concept, remains fixed until a drift is detected.

IVDD frames concept drift detection as a statistical hypothesis test, comparing the causal effect estimated from w_{ti} against the baseline effect from R_i. Let $\hat{\beta}_i^r$ and $\hat{\beta}_i^c$ denote the causal effect estimates obtained from the reference and current windows, respectively, with $\hat{v}_i^r$ and $\hat{v}_i^c$ denoting their non-parametric bootstrap variance estimates (resampled B times with replacement). We test $H_0 : \hat{\beta}_i^r = \hat{\beta}_i^c$ against $H_1 : \hat{\beta}_i^r \neq \hat{\beta}_i^c$ using the standardized statistic:

$$z_i = \frac{\hat{\beta}_i^c - \hat{\beta}_i^r}{\sqrt{\hat{v}_i^r + \hat{v}_i^c}}. \tag{3}$$

A monitor $\mathcal{Q}_i$ flags a drift whenever $|z_i|$ exceeds a critical value c. Since IVDD instantiates one monitor per IV structure, the detector signals a concept drift if *any* monitor rejects the null hypothesis:

$$\text{drift}_t = \bigvee_{i=1}^{m} \mathbb{1}\left[|z_i| > c\right]. \tag{4}$$

Because each IV structure is monitored separately, a drift signal can be localized to the corresponding structural relationship. To control the family-wise error rate across m simultaneous tests at nominal significance level α, the critical threshold c is adjusted via Bonferroni correction:

$$c = \Phi^{-1}\Big(1 - \frac{\alpha}{2m}\Big), \tag{5}$$

where $\Phi^{-1}(\cdot)$ is the standard normal quantile function. This guarantees that under the global null hypothesis (no mechanism changes in any structure), the probability of triggering a false alarm remains strictly bounded by α. We ablate on the impact of the Bonferroni correction in Appendix C.

4.4 Instrumental Variable-monitored Adaptive Tree

While IVDD monitors causal relationships at the global stream level, causal effects may differ across subpopulations and evolve differently over time. For example, a treatment may remain effective for the overall population while becoming less effective for a specific subgroup. We therefore introduce IVAT, an online decision tree that combines predictive partitioning with localized IV monitoring. Each node represents a subpopulation and maintains a local estimate of an identifiable causal effect, allowing mechanism changes to be detected and localized to the affected region. When drift is detected, IVAT adapts only the corresponding subtree rather than rebuilding the complete model.

Local Causal Monitoring Each IVAT node maintains a local monitor associated with an IV structure (Z, W, T). When multiple valid structures are available, IVAT selects one using three criteria: *1) target relevance*, prioritizing structures whose outcome T corresponds to the prediction target; *2) instrument strength*, ranking candidates by $\gamma_{Z,W} = |\text{Corr}(Z, W)|$; and *3) ancestor fallback*, selecting the strongest available instrument when no discovered structure directly targets the prediction outcome. The selected structure is then monitored locally throughout the node's lifetime. After an initial warmup period, causal monitoring is activated only when $\gamma_{Z,W} > \tau_{\text{IV}}$. If instrument strength subsequently falls below this threshold, the node becomes *inconclusive* and suspends causal monitoring until sufficient strength is recovered. An active node enters a *flagged* state when a statistically significant change in its Wald estimate is detected. This explicit treatment of weak instruments avoids interpreting unstable IV estimates as mechanism changes (Andrews et al., 2019).

Hybrid causal-predictive splitting IVAT constructs partitions that are useful for both prediction and identifying heterogeneous causal effects. For a candidate split, we compute the Gini gain G_{Gini} and, when valid IV estimates are available for both children, the causal-effect heterogeneity gain

$$G_{\text{IV}} = \frac{n_L n_R}{n^2}\left(\hat{\beta}_L - hat\beta_R\right)^2, \tag{6}$$

where $\hat{\beta}_L$ and $\hat{\beta}_R$ are the Wald estimates in the left and right child populations, and $n = n_L + n_R$. This criterion favors splits that yield sufficiently large subpopulations with distinct local causal effects, consistent with the principles of causal tree methods (Athey & Imbens, 2016). The final split score is

$$s = \lambda G_{\text{IV}} + (1 - \lambda) G_{\text{Gini}}, \qquad \lambda \in [0, 1], \tag{7}$$

where both gains are normalized to $[0, 1]$ across candidate splits. If a valid IV estimate is unavailable, the split is evaluated based solely on predictive gain. Candidate splits are selected using Hoeffding's bound, following HT (Domingos & Hulten, 2000).

Mechanism Cascade and Adaptive Replacement The hierarchical structure of IVAT is used to distinguish changes in mechanisms from changes in the composition of a node's population. When a node flags a change in its Wald estimate, a Hoeffding-bound proportion tracker first tests whether the routing proportions to its children have changed significantly, i.e., if the change refers to a *covariate shift* rather than a mechanism shift. If so, the change is attributed to population redistribution rather than a change in the structural mechanism, and the node is re-baselined using the most recent observations. If routing remains stable, the cascade recursively examines child subtrees to determine whether the change can be localized to a descendant, proceeding until the deepest responsible node is identified. When a flagged change cannot be explained by descendant nodes, or occurs at a leaf, IVAT initializes an alternative background subtree at the affected node and re-baselines its local IV monitor. Instances routed to the affected region are subsequently processed by both the primary and alternate subtrees. The alternate tree replaces the primary subtree when its predictive error falls significantly below that of the primary subtree, enabling localized adaptation without rebuilding unaffected portions of the tree. Appendix B provides further details and pseudocode.

5 EXPERIMENTS

We evaluate the applications of our IV framework on controlled synthetic streams generated from SCMs and on real-world data streams. Our experiments aim to answer three research questions: (i) can IVDD reliably detect mechanism changes in IV structures and localize *which* mechanism in the data-generating process drifted?, (ii) how does IVDD compare with existing drift detectors in learner predictive performance under delayed label availability? (iii) can we use the drift location to adapt an incremental learner (IVAT)? Experiments are conducted in an online manner, with varying levels of label delay. Further details about the experimental settings are provided in Appendix D.

Detecting drift in instrumental variables (RQ1) We construct a controlled multi-structure SCM with 50,000 instances that contains four IV structures, each comprising an instrument Z_i, treatment W_i, and latent confounder U_i, with $W_i = aZ_i + bU_i + \epsilon_{W_i}$. All treatments and confounders affect a shared outcome, $Y = \sum_{i=1}^{4}(\beta_i W_i + cU_i) + \epsilon_Y$. We introduce pre-defined shifts in β_i: Structure 1 undergoes *abrupt* drift, Structure 2 *gradual* drift (stochastic transition over 500 steps), Structure 3 *incremental* drift (linear transition over 500 steps), and Structure 4 remains stationary as a control. An independent IVDD monitor is assigned to each (Z_i, W_i, Y) structure. Results over 30 runs are reported in Table 1.

IVDD achieved TPRs of 1.0, 0.989, and 1.0 for abrupt, gradual, and incremental drift, respectively. Detection delay was higher than in the isolated single-structure setting (182.7 vs. 76.2; Appendix E), as additional variables increase variance in the shared outcome Y. Delays were also higher for gradual and incremental drift, where changes unfold over longer transition periods and a wider true-positive window is used (Pesaranghader & Viktor, 2016). Despite this, IVDD reliably detected all three drift types while producing only 0.57 false alarms on average for the stationary Structure 4. Figure 2 shows the monitored Z-scores over time, illustrating the localization of drift to the affected IV structures.

Table 1: True Positive Rate (TPR), False Positives (FPs), the F1 score, and the mean detection delay obtained by IVDD on a dataset generated using four IV structures. Results are an average of 30 runs.

Structure	Drift Type	TPR	FPs per run	F1	Mean Delay
Str 1	Abrupt	1.000 ± 0.000	0.90 ± 0.88	0.883 ± 0.106	182.7 ± 57.2
Str 2	Gradual	0.989 ± 0.061	1.20 ± 1.24	0.850 ± 0.147	433.7 ± 92.5
Str 3	Incremental	1.000 ± 0.000	1.13 ± 0.73	0.850 ± 0.086	426.1 ± 86.1
Str 4	None (Stationary)	—	0.57 ± 0.90	—	—

We next evaluate whether IV-based causal monitoring remains useful when timely labels are unavailable (RQ2), comparing IVDD with state-of-the-art supervised and unsupervised drift detectors on a subset of the USP DS repository (Souza et al., 2020). Dataset information is provided in Appendix D.

Comparing drift detectors (RQ2) We pair each detector with a standard HT classifier and evaluate immediate feedback and three label latencies, $L \in 100, 500, 1000$. Table 2 reports accuracy, Cohen's

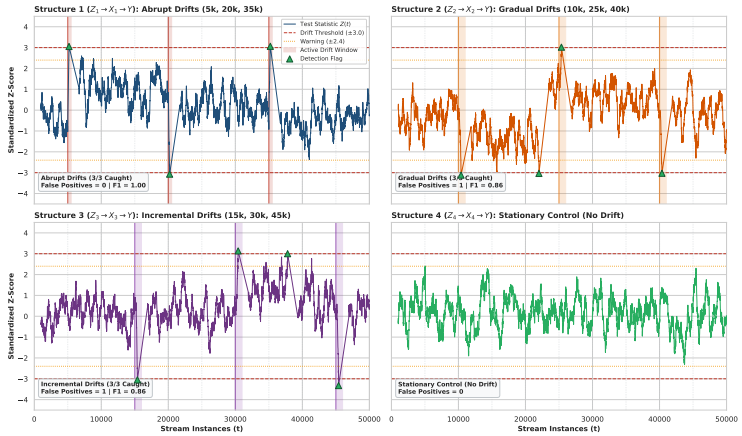


Figure 2: Z-Score over time for each IV structure in a dataset containing 4 IV structures that share the same outcome. Abrupt drift events occur in X_1 at $t = 5000, 20000$, and 35000. Gradual changes happens in X_2 at $t = 10000, 25000$, and 40000, and incremental changes in X_3 at $t = 15000, 30000$, and 45000. Gradual and incremental changes have drift length $\delta = 500$.

Table 2: Average Accuracy and Cohen's kappa (κ) of HT using different Drift Detectors in real-world datasets under immediate and delayed label availability.

Detector	Immediate			L=100			L=500			L=1000		
	Acc↑	κ ↑	Rank↓	Acc↑	κ ↑	Rank↓	Acc↑	κ ↑	Rank↓	Acc↑	κ ↑	Rank↓
Supervised												
OPTWIN	0.673	0.515	6.1	0.630	0.444	5.0	0.594	0.398	4.5	0.576	0.371	4.4
ADWIN	0.678	0.518	5.6	0.609	0.405	5.4	0.576	0.358	5.8	0.554	0.334	5.6
DDM	0.672	0.514	5.9	0.628	0.437	5.6	0.596	0.393	**3.6**	0.577	0.366	4.1
HDDMA	0.687	0.530	4.0	0.573	0.347	6.8	0.549	0.316	6.1	0.532	0.295	5.5
HDDMW	0.674	0.511	6.6	0.583	0.363	7.3	0.558	0.330	5.8	0.537	0.303	5.3
RDDM	**0.689**	**0.511**	**3.3**	0.606	0.354	5.0	0.573	0.371	5.0	0.550	0.325	4.9
STEPD	0.679	0.519	4.3	0.572	0.335	7.8	0.541	0.296	7.0	0.521	0.276	6.1
Unsupervised												
DD-SCC	0.671	0.455	7.4	0.602	0.387	5.3	0.540	0.298	7.9	0.516	0.269	8.0
DD-KRC	0.672	0.457	7.1	0.604	0.389	**4.6**	0.543	0.305	7.6	0.518	0.272	7.9
MMDEW	0.654	0.494	9.1	0.593	0.378	7.8	0.529	0.312	8.9	0.519	0.274	8.9
IVDD (Ours)	0.664	0.493	6.3	**0.638**	**0.453**	5.1	**0.612**	**0.424**	**3.6**	**0.604**	**0.407**	**2.9**

Kappa (κ), and average rank across streams. For IVDD, we discover the DAGs using LPCMCI (Gerhardus & Runge, 2020) on the initial warmup samples.

Under immediate feedback, supervised detectors (such as RDDM and HDDMA) achieve strong performance because error feedback is instantaneous. However, their performance generally degrades as label latency increases, whereas IVDD continuously monitors identified causal relationships without requiring labels. At $L = 100$, 500, and 1000, IVDD achieves accuracies of 63.8%, 61.2%, and 60.4%, respectively, with the best average rank at $L = 500$ and $L = 1000$. Among label-free baselines, IVDD similarly achieves higher accuracy at longer delays. At $L = 1000$, it exceeds DD-SCC, DD-KRC, and MMDEW by 8.8, 8.6, and 8.5 percentage points, respectively. These results support the hypothesis that causal-effect monitoring is beneficial under delayed feedback.

We further perform pairwise Wilcoxon signed-rank tests. As shown in Table 3, performance differences between immediate feedback and mild delay are not significant, but IVDD becomes statistically significant as latency increases. At $L = 500$, IVDD achieves statistically significant gains over DD-SCC ($p = 0.04$) and approaches significance against ADWIN ($p = 0.06$). Under higher latency ($L = 1000$), IVDD demonstrates statistically significant superiority over OPTWIN, DD-SCC, DD-KRC, RDDM, and ADWIN, confirming that causal invariance monitoring provides an advantage in the presence of label scarcity. Additional experiments with synthetic data is in Appendix E.2.

Table 3: P-values of the pairwise Wilcoxon signed-rank test comparing the accuracy of IVDD against other detectors. The "+" sign indicates significant differences from the column-wise method.

	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
Immediate feedback										
IVDD	0.31	0.74	0.84	0.46	0.10	0.31	0.15	0.25	0.64	0.25
$L = 100$										
IVDD	0.95	0.55	0.64	0.26	0.64	0.74	0.31	0.38	0.25	**0.04 (+)**
$L = 500$										
IVDD	0.38	**0.04 (+)**	0.08	0.40	0.15	0.06	0.11	0.11	0.11	**0.01 (+)**
$L = 1000$										
IVDD	**0.02 (+)**	**0.02 (+)**	**0.02 (+)**	0.11	**0.04 (+)**	**0.02 (+)**	0.07	0.06	0.06	**0.01 (+)**

Downstream Adaptation with IVAT (RQ3) Now we investigate whether incorporating IV monitoring into decision trees for splitting and targeted adaptation provides an advantage over existing streaming tree models. We report the overall accuracy and Cohen's kappa in Table 4. With immediate feedback, state-of-the-art adaptive trees such as PLASTIC (69.4%) and LAST (69.3%) achieve the highest accuracies, with IVAT at 68.1%. However, as label latency increases, IVAT demonstrates good resilience. At $L = 100$, IVAT achieves both the highest overall accuracy (65.2%) and Cohen's kappa ($\kappa = 0.473$), and also the best average rank (2.4). IVAT also achieved the best average Cohen's kappa and rank under $L = 500$. Under higher delay ($L = 1000$), IVAT maintains strong accuracy (61.0%) and achieves the highest Cohen's kappa (0.412, outperforming LAST's 0.376 and HAT's 0.399).

Table 4: Average accuracy and Cohen's kappa (κ) for different online tree classifiers on real-world datasets under immediate and delayed label availability.

Learner	Immediate			L=100			L=500			L=1000		
	Acc↑	$\kappa\uparrow$	Rank↓	Acc↑	$\kappa\uparrow$	Rank↓	Acc↑	$\kappa\uparrow$	Rank↓	Acc↑	$\kappa\uparrow$	Rank↓
HT	0.644	0.471	5.0	0.614	0.391	4.9	0.599	0.370	4.7	0.584	0.348	4.7
HAT	0.671	0.509	3.4	0.636	0.456	3.3	0.608	0.433	3.1	0.592	0.399	3.0
LAST	0.693	0.526	2.4	0.649	0.426	2.8	**0.629**	0.405	2.9	**0.612**	0.376	**2.7**
PLASTIC	**0.694**	**0.534**	3.0	0.640	0.407	3.4	0.605	0.375	4.1	0.595	0.334	3.8
EFDT	0.690	0.526	2.8	0.630	0.408	4.1	0.609	0.396	3.4	0.587	0.363	4.1
IVAT (Ours)	0.681	0.519	4.3	**0.652**	**0.473**	**2.4**	0.625	**0.438**	**2.8**	0.610	**0.412**	2.8

6 Conclusion

In this work, we introduced an online framework to measure causal effects using IVs on streaming data. From its foundation, we developed two applications: IVDD, a label-free drift detector that tracks shifts in the underlying data-generating process, and IVAT, an adaptive online decision tree that leverages causal heterogeneity for split decisions.

Some practical limitation are worth considering. First, IV-based monitoring depend on the IV assumptions (relevance, exclusion, and unconfoundedness). In domains lacking ground-truth DAGs, candidate triplets must be inferred using causal discovery algorithms, where latent confounding and non-stationarity risk introducing spurious or misoriented edges. Nonetheless, our empirical findings demonstrate that approximately discovered graphs still provide a reliable baseline for monitoring. Second, as the dimensionality gros, the number of candidate IVs also grows, risking the increase of false alarms. IVDD mitigates this through Bonferroni correction, but it can incur longer detection delay for subtle shifts.

Despite the challenges and limitations, IVDD and IVAT represent a significant conceptual shift for incremental learning on data streams, transitioning the monitoring objective from distribution shifts to underlying structural mechanisms. Future research includes integrating online causal discovery to dynamically update the underlying DAG, as well as developing dependence-aware multiple testing to mitigate conservative detection thresholds. Ultimately, IVDD and IVAT provide a foundational step toward mechanism-aware, robust learning across non-stationary environments.

REFERENCES

Supriya Agrahari and Anil Kumar Singh. Comparison based analysis of window approach for concept drift detection and adaptation. *Applied Intelligence*, 55(1):39, Nov 2024. ISSN 1573-7497.

Brenno Alencar, Guilherme Cassales, Heitor Murilo Gomes, Cássio Prazeres, Tatiane Rios, Albert Bifet, and Ricardo Rios. Concept drift detection in delayed and partially labeled data streams: An experimental survey. *Digital Signal Processing*, 182:106325, 2026. ISSN 1051-2004.

Isaiah Andrews, James H Stock, and Liyang Sun. Weak instruments in instrumental variables regression: Theory and practice. *Annual Review of Economics*, 11(1):727–753, 2019.

Joshua D Angrist, Guido W Imbens, and Donald B Rubin. Identification of causal effects using instrumental variables. *Journal of the American statistical Association*, 91(434):444–455, 1996.

Susan Athey and Guido Imbens. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016.

Susan Athey, Julie Tibshirani, and Stefan Wager. Generalized random forests, 2018.

Manuel Baena-García, José Campo-Ávila, Raúl Fidalgo-Merino, Albert Bifet, Ricard Gavald, and Rafael Morales-Bueno. Early drift detection method. *Proc. 4th Int. Workshop Knowledge Discovery from Data Streams*, pp. 77–86, 01 2006.

Lucas Baier, Tim Schlör, Jakob Schöffer, and Niklas Kühl. Detecting concept drift with neural network model uncertainty, 2022.

Eduardo V. L. Barboza, Jean Paul Barddal, Robert Sabourin, and Rafael M. O. Cruz. Cadrift: A time-dependent causal generator of drifting data streams, 2026a.

Eduardo V. L. Barboza, Jean Paul Barddal, Robert Sabourin, and Rafael M. O. Cruz. Concept drift from a causal perspective, 2026b.

Roberto S.M. Barros, Danilo R.L. Cabral, Paulo M. Gonçalves, and Silas G.T.C. Santos. Rddm: Reactive drift detection method. *Expert Systems with Applications*, 90:344–355, 2017. ISSN 0957-4174.

Albert Bifet and Ricard Gavaldà. Learning from time-changing data with adaptive windowing. In *SDM*, pp. 443–448, 2007.

Albert Bifet and Ricard Gavalda. Adaptive learning from evolving data streams. In *International symposium on intelligent data analysis*, pp. 249–260. Springer, 2009.

Vitor Cerqueira, Heitor Murilo Gomes, Albert Bifet, and Luis Torgo. Studd: a student–teacher method for unsupervised concept drift detection. *Machine Learning*, 112(11):4351–4378, Nov 2023. ISSN 1573-0565.

Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pp. 71–80, New York, NY, USA, 2000. Association for Computing Machinery. ISBN 1581132336.

P. Erdős and A. Rényi. Asymmetric graphs. *Acta Mathematica Academiae Scientiarum Hungarica*, 14(3):295–315, Sep 1963. ISSN 1588-2632.

Isvani Frías-Blanco, José del Campo-Ávila, Gonzalo Ramos-Jiménez, Rafael Morales-Bueno, Agustín Ortiz-Díaz, and Yailé Caballero-Mota. Online and non-parametric drift detection methods based on hoeffding's bounds. *IEEE Transactions on Knowledge and Data Engineering*, 27(3):810–823, 2015.

João Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. Learning with drift detection. In Ana L. C. Bazzan and Sofiane Labidi (eds.), *Advances in Artificial Intelligence – SBIA 2004*, pp. 286–295, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. ISBN 978-3-540-28645-5.

Andreas Gerhardus and Jakob Runge. High-recall causal discovery for autocorrelated time series with latent confounders. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 12615–12625. Curran Associates, Inc., 2020.

Heitor Murilo Gomes, Anton Lee, Nuwan Gunasekara, Yibin Sun, Guilherme Weigert Cassales, Justin Jia Liu, Marco Heyden, Vitor Cerqueira, Maroua Bahri, Yun Sing Koh, Bernhard Pfahringer, and Albert Bifet. CapyMOA: Efficient machine learning for data streams in python, 2025. URL https://arxiv.org/abs/2502.07432.

Brandon Gower-Winter, Misja Groen, and Georg Krempl. The window dilemma: Why concept drift detection is ill-posed, 2026.

Ege Berkay Gulcan and Fazli Can. Unsupervised concept drift detection for multi-label data streams. *Artificial Intelligence Review*, 56(3):2401–2434, Mar 2023. ISSN 1573-7462.

Marco Heyden, Heitor Murilo Gomes, Edouard Fouché, Bernhard Pfahringer, and Klemens Böhm. Leveraging plasticity in incremental decision trees. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 38–54. Springer, 2024.

Florian Kalinke, Marco Heyden, Georg Gntuni, Edouard Fouché, and Klemens Böhm. Maximum mean discrepancy on exponential windows for online change detection, 2025.

Mette Lise Lousdal. An introduction to instrumental variable assumptions, validation and estimation. *Emerging themes in epidemiology*, 15(1):1, 2018.

Jie Lu, Anjin Liu, Fan Dong, Feng Gu, João Gama, and Guangquan Zhang. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*, 31(12):2346–2363, 2019. doi: 10.1109/TKDE.2018.2876857.

Chaitanya Manapragada, Geoffrey I Webb, and Mahsa Salehi. Extremely fast decision tree. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1953–1962, 2018.

Kyosuke Nishida and Koichiro Yamauchi. Detecting concept drift using statistical testing. In Vincent Corruble, Masayuki Takeda, and Einoshin Suzuki (eds.), *Discovery Science*, pp. 264–269, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-75488-6.

Daniel Nowak Assis, Jean Paul Barddal, and Fabrício Enembreck. Behavioral insights of adaptive splitting decision trees in evolving data stream classification: D. nowak assis et al. *Knowledge and Information Systems*, 67(7):5751–5782, 2025.

Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995. ISSN 00063444, 14643510.

Russel Pears, Sripirakas Sakthithasan, and Yun Sing Koh. Detecting concept change in dynamic data streams. *Machine Learning*, 97(3):259–293, December 2014. ISSN 1573-0565.

Ali Pesaranghader and Herna L Viktor. Fast hoeffding drift detection method for evolving data streams. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 96–111. Springer, 2016.

Ali Pesaranghader, Herna Viktor, and Eric Paquet. Mcdiarmid drift detection methods for evolving data streams, 2018.

Sripirakas Sakthithasan, Russel Pears, and Yun Sing Koh. One pass concept change detection for data streams. In Jian Pei, Vincent S. Tseng, Longbing Cao, Hiroshi Motoda, and Guandong Xu (eds.), *Advances in Knowledge Discovery and Data Mining*, pp. 461–472, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-37456-2.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Vinicius M. A. Souza, Denis M. dos Reis, André G. Maletzke, and Gustavo E. A. P. A. Batista. Challenges in benchmarking stream learning algorithms with real-world data. *Data Mining and Knowledge Discovery*, 34(6):1805–1858, jul 2020.

Mauro D. L. Tosi and Martin Theobald. Optwin: Drift identification with optimal sub-windows. In *2024 IEEE 40th International Conference on Data Engineering Workshops (ICDEW)*, pp. 331–337. IEEE, May 2024.

AI USE STATEMENT

In this work, we used generative AI tools (Large Language Models) for editing and polishing the manuscript text to improve readability, checking grammar and formatting LaTeX tables, and assisting with debugging and plotting scripts. We have not used generative AI tools for generating synthetic datasets, developing the theoretical framework, formulating mathematical claims, proposing hypotheses, or interpreting experimental results. We have reviewed and verified all AI-assisted work: all experimental implementations and plotting scripts were manually inspected and tested for correctness by the authors, and all AI-edited text was verified for factual and technical accuracy. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

A INSTRUMENT DISCOVERY PSEUDOCODE

Algorithm 1 shows the pseudocode for instrument discovery. It tests the assumptions of relevance and exclusion in order to extract valid instrument from a DAG $\mathcal{G}$.

Algorithm 1 Instrument Discovery

Require: DAG $\mathcal{G} = (V, E)$
$\mathcal{I} \leftarrow \emptyset$
for $W \in V$ **do**
  if W has no parents **or** no children **then**
    continue
  for $T \in$ children(W) **do**
    $\mathcal{G}_W \leftarrow \mathcal{G}$ with all outgoing edges of W removed
    for $Z \in V \setminus \{W, T\}$ **do**
      relevant $\leftarrow \neg(Z \perp\!\!\!\perp W \mid \mathcal{G}_W)$
      exclusion $\leftarrow (Z \perp\!\!\!\perp T \mid \mathcal{G}_W)$
      if relevant **and** exclusion **then**
        $\mathcal{I} \leftarrow \mathcal{I} \cup \{(Z, W, T)\}$
return $\mathcal{I}$

B TOP-DOWN CASCADE PSEUDOCODE

We present the pseudocode for IVAT's top-down cascade procedure in Algorithm 2. The procedure is triggered when a node flags a change in its IV estimate. Such a change may arise from: 1) a covariate shift that changes routing proportions, 2) a causal change in a descendant subtree, or 3) a change at the flagged node itself. The cascade aims to identify the source of the observed shift.

Let v be the flagged node. If v is a leaf, the shift cannot be delegated to a descendant, so the leaf is rebuilt (Lines 1–3). At an internal node, we assume that the Wald estimate can be viewed approximately as a mixture of the effects of its children, $\beta_v \approx \sum_c p_c \beta_c$. Consequently, routing-proportion changes can alter β_v even when the causal relationships within all children remain unchanged. If the Hoeffding proportion tracker detects a significant change in the routing proportions, the flag is attributed to covariate shift. Node v then re-baselines its Wald monitor to the new mixture distribution without discarding its learned subtree (Lines 4–7).

If the routing proportions are stable, the cascade recursively examines children whose monitors have flagged (Lines 8–10). If any flagged child returns AMBIGUOUS, the parent is also marked

Algorithm 2 Top-down cascade for resolving a flagged IVAT node

Require: flagged node v
Ensure: outcome $\in$ REBUILT, COVSHIFT, AMBIGUOUS
1: **if** v is a leaf **then**
2: REBUILD(v)
3: **return** REBUILT
4: **if** v has a detected covariate shift **then** ▷ Measured by Hoeffding's bound on children proportion
5: REBASELINE(v)
6: Reset covariate-shift tracker of v
7: **return** COVSHIFT
8: $F \leftarrow c \in \text{children}(v) : c$ is FLAGGED
9: **for all** $c \in F$ **do**
10: $o_c \leftarrow$ CASCADE(c) ▷ Check descendants
11: **if** $\exists c \in F : o_c =$ AMBIGUOUS **then**
12: $v.\text{state} \leftarrow$ AMBIGUOUS
13: **return** AMBIGUOUS
14: **else if** $F \neq \emptyset$ **then**
15: REBASELINE(v)
16: **return** REBUILT
17: **if** $\exists c \in \text{children}(v) : c$ is INCONCLUSIVE **then**
18: $v.\text{state} \leftarrow$ AMBIGUOUS
19: **return** AMBIGUOUS
20: REBUILD(v)
21: **return** REBUILT

AMBIGUOUS, since the source of the change cannot be reliably localized. Otherwise, if at least one flagged child resolves the change, the parent re-baselines its own monitor to reflect the updated child subtree while leaving unaffected siblings intact (Lines 11–16).

If no child is flagged, but at least one child is INCONCLUSIVE due to weak instrument strength ($|\text{corr}(Z, W)| < \tau$) or insufficient sample support, the parent cannot reliably determine whether the change originated below it. Rather than assuming the child is stable and rebuilding v, the cascade propagates an AMBIGUOUS state upward (Lines 17–19). Finally, if no descendant is flagged or inconclusive, the change is attributed to v itself, and the node is rebuilt (Lines 21–22).

C Ablation Study: Scaling Graph Size and Multiple Hypothesis Correction

We conduct experiments to evalute the impact of Bonferroni correction on mitigating false alarms when multiple IV monitors work together. We generate synthetic data with $K \in \{1, 2, 4, 8, 12, 16\}$ IVs, and measure the number of false alarms in streams of $25{,}000$ instances across 10 runs. As shown in Figure 3, Bonferroni correction succeeded in decreasing the average number of false alarms.

D Additional experimental details

Assessing Drift Detection To score detection, we match each ground-truth drift point t_d to the earliest detector flag within a tolerance window $[t_d, t_d + \gamma]$ (we use $\gamma = 500$ for abrupt drift (since IVDD uses a sliding window of 500 samples), and $\gamma = 1,000$ for gradual and incremental drift, as in (Pesaranghader & Viktor, 2016)). A ground-truth point with no flag in its tolerance window is counted as a false negative (FN). Flags that cannot be matched to any ground-truth point are counted as false positives (FP). For matched detections, we call them true positives (TP). We also report the F1-score and detection delay given by $t_{\text{flag}} - t_d$.

Dataset Generation Protocol In our controlled synthetic experiments, we use two SCMs: one containing a single IV structure with a hidden confounder, shown in Figure 4a, and another containing four IV structures, shown in Figure 4b. In the single-IV structure SCM, the exogenous variables

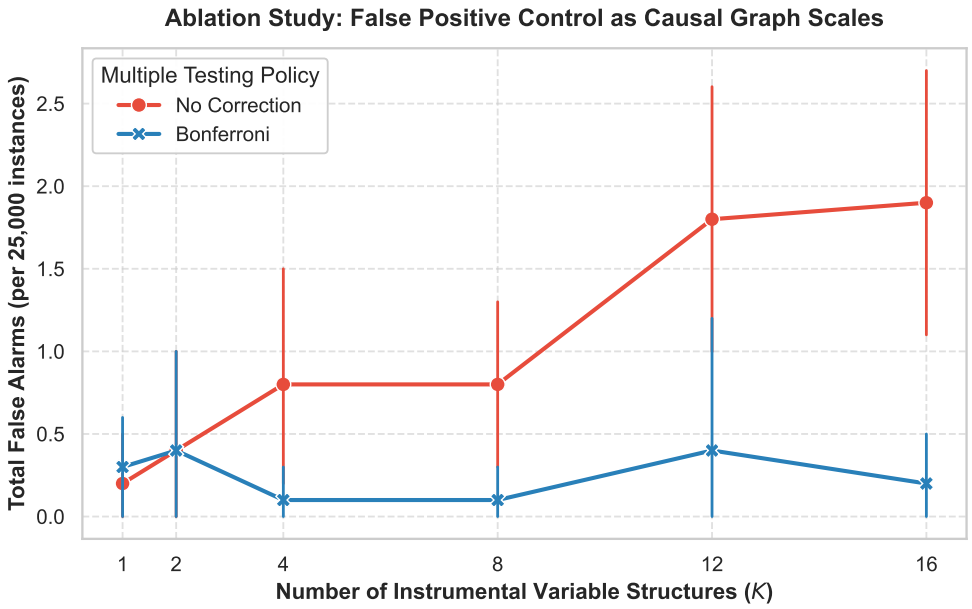
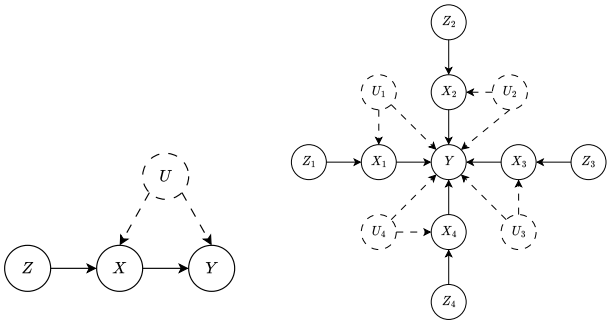


Figure 3: False alarm count vs. the number of monitored IV structures.

Z and U are independently sampled from $\mathcal{N}(0,1)$. The treatment variable X is generated as $X = aZ + bU + \epsilon_X$, while the outcome Y is $Y = \beta X + cU + \epsilon_Y$, such that $\epsilon_X, \epsilon_Y \sim \mathcal{N}(0, 0.1)$. We introduce concept drift by treating β as piecewise constant over the stream: a new value of β is drawn independently from $\mathcal{U}(-3, 3)$ every $5,000$ instances.

For the four IV structure experiment, we sample data using CaDrift Barboza et al. (2026a). We have an SCM with 4 instruments ($Z_1..Z_4$), 4 treatments ($X_1..X_4$), and 4 latent confounders ($U_1..U_4$), all sharing the same outcome Y. The interventions are applied every 5,000 samples. First, we apply abrupt drift to X_1, then gradual drift to X_2, incremental drift to X_3, and then 10,000 samples pass without a drift event (we expect that no monitor triggers a drift event), and repeat once. This generates datasets of 50,000 samples. We did not apply any intervention on X_4.



(a) DAG #1. (b) DAG #2.

Figure 4: DAGs used to generate synthetic data for assessing IVDD's drift tracking.

D.1 Real-world Evaluation Experimental Details

Real-world Datasets We evaluate on a subset of the USP DS Repository: 8 widely adopted real-world streaming datasets from the literature (available in MOA/CapyMOA), spanning diverse domains including weather forecasting, sensor streams, aviation, and biology:

1. **NOAA**: A meteorological stream consisting of $18{,}159$ daily observations, 8 continuous features (temperature, dew point, wind speed, pressure, precipitation), and 2 classes. The objective is to predict whether rain will occur on the following day. Concept drift is driven by recurring seasonal weather transitions and gradual sensor recalibration.
2. **Electricity**: A pricing time-series benchmark consisting of $45{,}312$ sequential instances, 8 numerical features, and 2 classes (UP vs. Down). The objective is to forecast the direction of price movements in the New South Wales electricity market. Concept drift arises from demand shifts, generation supply fluctuations, and regulatory market policies.
3. **Airlines**: A flight performance stream consisting of $539{,}382$ records, 7 attributes (flight departure timing, origin/destination distance, carrier), and 2 classes. The objective is to predict whether a scheduled flight will be delayed. Concept drift is induced by seasonal flight volumes, airport scheduling congestion, and logistics.
4. **Insects**: An optical sensor stream evaluating flying insect classification across 6 species, with 33 wing-beat acoustic and morphological features ($52{,}848$ instances for abrupt drift). The objective is to identify insect species and sex. Abrupt, gradual, and incremental concept drifts are induced by ambient environmental temperature changes that alter wing-beat frequencies.
5. **Covertype**: A large-scale geospatial benchmark consisting of $581{,}012$ instances, 54 cartographic and soil features, and 7 classes. The objective is to predict the predominant forest tree species across land parcels. Streaming the data introduces spatial concept drift as the sequence traverses distinct geographical wilderness zones and elevation regimes.
6. **Rialto**: An outdoor computer vision stream consisting of $82{,}250$ sequential frames, 27 normalized color histogram features, and 10 visual scene classes. The objective is to classify specific architectural building facades visible from a fixed Venice webcam. Concept drift is driven by illumination shifts from the day-night cycle, cloud cover, and weather conditions.

We report the meta-information in Table 5.

Table 5: Datasets used in the real-world evaluation

Dataset	# instances	# features	# classes
Electricity	45,312	8	2
Airlines	539,382	7	2
NOAA	18,159	8	2
Insects^A	52,848	33	6
Insects^G	24,150	33	6
Insects^I	57,018	33	6
Covertype	581,012	54	7
Rialto	82,250	27	10

Online Streaming Evaluation Setup We adopt a warmup of $2{,}500$ samples for the base learners on all of all experiments, which is also the number of samples used for causal discovery with LPCMCI. For datasets with $d > 15$, we decrease the warmup size to $1{,}000$ samples for quicker causal discovery. We adopt the following policies for evaluation:

- **Immediate feedback (Test-then-train)**: each incoming streaming instance $\mathbf{x}_t$ is first predicted by the current model, and the label y is available right after prediction for incremental learning.
- **Label latency** (L): in real-world streams, true ground-truth labels y_t are scarce. We evaluate under delayed label feedback with delay parameter $L \in \{100, 500, 1000\}$:

  $$\text{Supervised observation at time } t: \quad \mathbf{x}_{t-L}, y_{t-L}.$$

  Supervised drift detectors (e.g., DDM, ADWIN, HDDM, OPTWIN) only observe error signals at time step t for instance $t - L$, causing a lag of at least L steps in drift alerts. In contrast, IVAT's incrument tracker operate directly on incoming streams. The unsupervised detectors used for comparison can also update their statistics with the ground-truth labels.

- **Model Adaptation**: The base model (Hoeffding Tree) is updated using the delayed stream. Upon drift detection, adaptation policies are:
  1. *Standard policy*: Immediate reset of the base model followed by training on newly available samples.
  2. *Background tree*: The existing model is retained to give predictions throughout a period of duration L. When delayed labels from the post-drift distribution become available at time $t + L$, the model is replaced and trained on the post-drift supervision. IVDD and unsupervised detectors (DD-SCC, DD-KRC, and MMDEW) use this policy under delayed feedback.

Drift Detectors for Comparison Most of the detectors used in our evaluation are available in the CapyMOA library (Gomes et al., 2025). The exceptions are MMDEW, which we used the authors' implementation (Kalinke et al., 2025), while DD-SCC, and DD-KRC are our implementations based on the paper's description (Agrahari & Singh, 2024), as we did not find official implementations online.

- **DDM**: monitors the online error rate of a classifier and detects drift when the estimated error rate significantly increase relative to the historical minimum (Gama et al., 2004).
- **RDDM**: extends DDM with a reactive mechanism that periodically discard outdated observations from the error history, preventing long stable periods from making the detector insensitive to recent changes (Barros et al., 2017).
- **HDDMA**: uses Hoeffding's inequality to detect significant changes in the classifier's error rate. It estimates the error using a moving-average statistic and singals warning or drift when the observed difference exceeds the corresponding Hoeffding bound (Frías-Blanco et al., 2015).
- **HDDMW**: replaces the moving-average of HDDMA with a moving weighted average (Frías-Blanco et al., 2015).
- **ADWIN**: maintains a dynamically sized window and repeatedly tests whether the two subwindows have statistically different means (Bifet & Gavaldà, 2007).
- **OPTWIN**: maintains a window of classifier error and searches for a statistically optimal split between historical and recent errors (Tosi & Theobald, 2024).
- **STEP**: compares the recent classification accuracy with the historical accuracy using a statistical test of equal proportions (Nishida & Yamauchi, 2007).
- **MMDEW**: uses Maximum Mean Discrepancy (MMD) to compare observations from different portions of the stream. It organizes observations into exponentially sized windows for two-samples testing while detecting changes in the underlying data distribution (Kalinke et al., 2025).
- **DD-SCC and DD-KRC**: correlation-based drift detectors that monitor changes in feature distributions. DD-SCC uses Spearman's rank correlation coefficient, whereas DD-KRC uses Kendall's rank correlation coefficient to quantify changes in feature-level statistics. We implemented both detectors based on the description provided in (Agrahari & Singh, 2024).

Incremental trees for Comparison

- **Hoeffding Tree (HT)**: an incremental decision tree learner that constructs a tree from streaming data using the Hoeffding bound to make split decisions (Domingos & Hulten, 2000).
- **Hoeffding Adaptive Tree (HAT)**: extends HT with mechanisms for adapting to concept drift. Change detectors monitor the performance of individual tree nodes, and when deterioration is detected, alternative subtrees are grown and may replace existing subtrees once they become more accurate (Bifet & Gavalda, 2009).
- **Extremely Fast Decision Tree (EFDT)**: modifies the incremental splitting strategy of HT by allowing split decisions to be reconsidered as additional observations arrive (Manapragada et al., 2018).

- **Local Adaptive Streaming Tree (LAST)**: an adaptive decision tree that uses performance degradation to guide structural adaptation. LAST monitors the state and performance of individual tree regions and branches the tree when an alternative structure may be beneficial (Nowak Assis et al., 2025).
- **PLASTIC**: an incremental decision tree designed to improve the structural adaptation of EFDT (Heyden et al., 2024).

E Additional Experiments

To test whether IVDD can track drift in IV structures, we sample according to the SCM presented in Figure 4a (Appendix D). IVDD achieved a 0.981 true positive rate (TPR) and an average of 1.1 false positives per run, with an F1-score of 0.935. The mean detection delay was 76.2, indicating that, for a single independent IV structure, IVDD can reliably detect changes in cause–effect mechanisms. These results are summarized in Table 6. In Figure 5, we present the Z-Score over time and the points at which IVDD triggered concept drift in one run. For a single IV structure, IVDD provides fast and accurate detection of concept drift.

Table 6: True Positive Rate (TPR), False Positives (FPs) per run, and mean detection delay obtained by IVDD on a dataset generated using a single IV structure. Results are an average of 30 runs.

Metric	Mean $\pm$ std dev
TPR	0.978 ± 0.054
FPs per run	1.17 ± 1.12
F1	0.930 ± 0.073
Detection Delay	74.5 ± 5.3

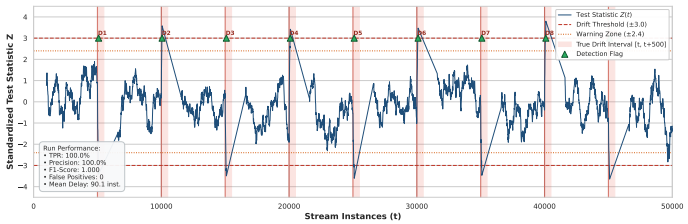


Figure 5: IVDD's Z-Score over time in a dataset sampled using an SCM with one IV structure.

E.1 Full results – real-world comparison

Here we present the full results on all real-world data streams. The accuracies for immediate feedback and all latency levels ($L = \{100, 500, 1000\}$) are in Tables 7, 8, 9, and 10, respectively. The Cohen's kappa (κ), under the same configurations are in Tables 11, 12, 13, and 14.

Table 7: Full overall accuracy results in real-world data streams with immediate feedback.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.733	0.743	0.687	0.686	0.717	0.726	0.744	0.721	0.719	0.721	0.686
Airlines	0.649	0.650	0.576	0.579	0.647	0.645	0.643	0.640	0.642	0.634	0.597
Electricity	0.829	0.844	0.83.3	0.829	0.857	0.854	0.845	0.854	0.857	0.849	0.837
Insects-AB	0.655	0.635	0.648	0.664	0.614	0.653	0.648	0.660	0.657	0.643	0.599
Insects-GB	0.651	0.677	0.699	0.699	0.698	0.708	0.682	0.705	0.683	0.677	0.642
Insects-IB	0.562	0.550	0.535	0.534	0.544	0.569	0.553	0.566	0.556	0.553	0.551
Covertype	0.845	0.850	0.834	0.834	0.869	0.864	0.836	0.868	0.872	0.850	0.834
Rialto	0.388	0.437	0.555	0.554	0.431	0.496	0.470	0.417	0.511	0.461	0.489
Average	0.664	0.673	0.671	0.672	0.672	0.689	0.678	0.679	0.687	0.674	0.654

Table 8: Full overall accuracy results in real-world data streams with $L = 100$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.720	0.732	0.664	0.677	0.712	0.675	0.713	0.667	0.678	0.672	0.689
Airlines	0.645	0.650	0.625	0.627	0.649	0.641	0.638	0.628	0.639	0.629	0.636
Electricity	0.758	0.729	0.680	0.681	0.709	0.673	0.714	0.590	0.614	0.656	0.703
Insects-AB	0.643	0.645	0.660	0.656	0.639	0.651	0.651	0.648	0.656	0.655	0.603
Insects-GB	0.643	0.667	0.698	0.697	0.645	0.682	0.664	0.690	0.647	0.675	0.639
Insects-IB	0.545	0.548	0.568	0.568	0.562	0.565	0.558	0.559	0.558	0.566	0.553
Covertype	0.815	0.738	0.558	0.555	0.780	0.614	0.605	0.466	0.456	0.504	0.589
Rialto	0.334	0.328	0.364	0.373	0.328	0.344	0.331	0.331	0.333	0.310	0.329
Average	**0.638**	0.630	0.602	0.604	0.628	0.606	0.609	0.572	0.573	0.583	0.593

Table 9: Full overall accuracy results in real-world data streams with $L = 500$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.717	0.725	0.665	0.692	0.725	0.675	0.687	0.659	0.653	0.653	0.686
Airlines	0.648	0.651	0.580	0.580	0.647	0.640	0.639	0.607	0.631	0.607	0.585
Electricity	0.730	0.674	0.516	0.504	0.632	0.592	0.599	0.493	0.509	0.544	0.434
Insects-AB	0.603	0.601	0.601	0.610	0.607	0.611	0.617	0.608	0.613	0.614	0.580
Insects-GB	0.630	0.637	0.645	0.644	0.627	0.626	0.629	0.651	0.636	0.639	0.610
Insects-IB	0.548	0.542	0.543	0.541	0.554	0.557	0.540	0.550	0.553	0.551	0.546
Covertype	0.803	0.734	0.615	0.621	0.762	0.677	0.720	0.573	0.596	0.660	0.625
Rialto	0.221	0.190	0.156	0.156	0.212	0.203	0.179	0.185	0.204	0.194	0.164
Average	**0.612**	0.594	0.540	0.543	0.596	0.573	0.576	0.541	0.549	0.558	0.529

E.2 Experiments with random SCM-generated datasets

We generate 10 random DAGs using Erdõs-Rényi (Erdős & Rényi, 1963) varying the dimensionality $d \in \{10, 20, 30\}$, and the edge probability $p \in [0.1, 0.4]$. We sample datasets using CaDrift (Barboza et al., 2026a;b), and generate data with *exogenous*, *endogenous*, *confounder*, and *target* drifts. We also generate synthetic data with different drift rates: *abrupt*, *gradual*, and *incremental*, for a total of 120 datasets per delay policy.

We expose the metrics' visualization for immediate feedback, $L = 100$, $L = 500$, and $L = 1000$ in Figures 6, 7, 8, and 9, respectively. For immediate feedback, IVDD and MMDEW almost tie in TPR. The methods that achieved the biggest TPR were DD-SCC and DD-KRC, followed by STEPD and HDDM-W. However, these detectors also presented a high number of false alarms, bringing down their F1-score. We notice a decrease in IVDD's TPR under delayed feedback, which we attribute to its monitoring of IV structures related to the target variable if an IV structure including it is available (updated only when the label becomes available). Even though MMDEW achieved the best detection metrics in these experiments, IVDD has still shown dominance over it in the real-world evaluation, as highlighted in our experiments in Section 5.

Table 10: Overall accuracy results on real-world data streams with $L = 1000$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.712	0.695	0.669	0.679	0.713	0.681	0.694	0.688	0.679	0.687	0.678
Airlines	0.648	0.652	0.577	0.576	0.648	0.630	0.635	0.594	0.624	0.593	0.612
Electricity	0.716	0.641	0.465	0.460	0.639	0.526	0.585	0.475	0.515	0.520	0.474
Insects-AB	0.565	0.558	0.555	0.562	0.566	0.569	0.570	0.566	0.574	0.571	0.534
Insects-GB	0.594	0.593	0.595	0.598	0.589	0.604	0.591	0.606	0.611	0.592	0.582
Insects-IB	0.545	0.533	0.536	0.535	0.533	0.533	0.528	0.544	0.533	0.547	0.536
Covertype	0.774	0.748	0.592	0.590	0.655	0.626	0.648	0.526	0.534	0.599	0.596
Rialto	0.280	0.192	0.138	0.142	0.272	0.234	0.183	0.172	0.185	0.188	0.137
Average	**0.604**	0.576	0.516	0.518	0.577	0.550	0.554	0.521	0.532	0.537	0.519

Table 11: Full overall Cohen's kappa results in real-world data streams with immediate feedback.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.326	0.392	0.010	0.004	0.347	0.358	0.377	0.335	0.346	0.346	0.278
Airlines	0.260	0.282	0.080	0.082	0.272	0.267	0.262	0.258	0.260	0.245	0.256
Electricity	0.649	0.679	0.656	0.651	0.707	0.699	0.700	0.699	0.706	0.688	0.670
Insects-AB	0.585	0.562	0.578	0.597	0.536	0.584	0.578	0.591	0.588	0.572	0.529
Insects-GB	0.582	0.612	0.639	0.639	0.638	0.649	0.618	0.647	0.619	0.612	0.570
Insects-IB	0.470	0.460	0.442	0.441	0.453	0.483	0.463	0.479	0.467	0.464	0.462
Covertype	0.752	0.759	0.732	0.733	0.789	0.781	0.738	0.788	0.793	0.759	0.754
Rialto	0.320	0.375	0.504	0.505	0.368	0.496	0.411	0.352	0.456	0.401	0.433
Average	0.493	**0.515**	0.455	0.457	0.514	0.540	0.518	0.519	0.530	0.511	0.494

Table 12: Full overall Cohen's kappa results in real-world data streams with $L = 100$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.293	0.368	0.230	0.255	0.302	0.271	0.310	0.201	0.273	0.258	0.245
Airlines	0.259	0.278	0.226	0.229	0.274	0.258	0.252	0.232	0.253	0.235	0.249
Electricity	0.504	0.444	0.364	0.360	0.407	0.339	0.423	0.177	0.221	0.304	0.396
Insects-AB	0.570	0.574	0.592	0.588	0.567	0.581	0.581	0.578	0.588	0.586	0.524
Insects-GB	0.582	0.601	0.637	0.637	0.574	0.619	0.597	0.627	0.576	0.611	0.567
Insects-IB	0.466	0.458	0.481	0.482	0.474	0.478	0.469	0.470	0.470	0.479	0.463
Covertype	0.700	0.577	0.270	0.268	0.647	0.378	0.355	0.135	0.135	0.201	0.325
Rialto	0.248	0.253	0.294	0.298	0.254	0.271	0.256	0.256	0.259	0.233	0.254
Average	**0.453**	0.444	0.387	0.389	0.437	0.399	0.405	0.335	0.347	0.363	0.378

Table 13: Full overall Cohen's kappa results in real-world data streams with $L = 500$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.286	0.348	0.194	0.251	0.321	0.235	0.259	0.188	0.222	0.215	0.220
Airlines	0.257	0.280	0.132	0.129	0.272	0.258	0.255	0.190	0.235	0.189	0.223
Electricity	0.450	0.344	0.077	0.063	0.265	0.193	0.205	0.021	0.046	0.113	0.094
Insects-AB	0.520	0.522	0.522	0.532	0.528	0.533	0.540	0.529	0.536	0.537	0.496
Insects-GB	0.562	0.565	0.574	0.573	0.552	0.551	0.555	0.581	0.563	0.567	0.532
Insects-IB	0.449	0.450	0.451	0.451	0.465	0.468	0.448	0.461	0.464	0.462	0.455
Covertype	0.691	0.575	0.375	0.383	0.618	0.482	0.518	0.307	0.352	0.452	0.403
Rialto	0.180	0.100	0.064	0.062	0.125	0.114	0.088	0.094	0.115	0.105	0.072
Average	**0.424**	0.398	0.298	0.305	0.393	0.354	0.358	0.296	0.316	0.330	0.312

Table 14: Full overall Cohen's kappa results in real-world data streams with $L = 1000$.

Dataset	IVDD	OPTWIN	DD-SCC	DD-KRC	DDM	RDDM	ADWIN	STEPD	HDDMA	HDDMW	MMDEW
NOAA	0.312	0.274	0.201	0.229	0.300	0.260	0.269	0.230	0.238	0.249	0.190
Airlines	0.259	0.279	0.112	0.113	0.269	0.234	0.246	0.161	0.221	0.161	0.188
Electricity	0.425	0.297	0.041	0.030	0.296	0.114	0.212	0.043	0.093	0.111	0.053
Insects-AB	0.478	0.469	0.466	0.475	0.479	0.483	0.484	0.479	0.489	0.485	0.441
Insects-GB	0.518	0.512	0.514	0.518	0.507	0.525	0.510	0.527	0.533	0.510	0.498
Insects-IB	0.455	0.439	0.443	0.441	0.439	0.439	0.434	0.452	0.440	0.457	0.443
Covertype	0.640	0.593	0.329	0.327	0.444	0.395	0.428	0.228	0.249	0.352	0.335
Rialto	0.167	0.102	0.042	0.046	0.191	0.148	0.093	0.081	0.095	0.098	0.041
Average	**0.407**	0.371	0.269	0.272	0.366	0.325	0.334	0.275	0.295	0.303	0.274

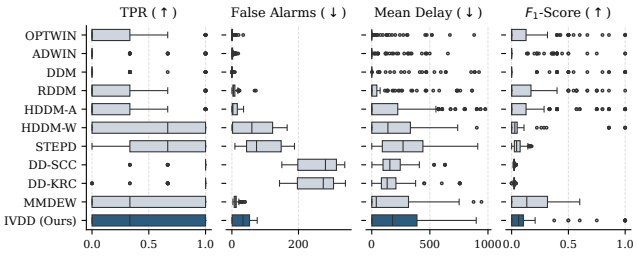


Figure 6: Averaged metrics on 120 synthetic datasets generated using CaDrift with different types of causal drift: Immediate feedback.

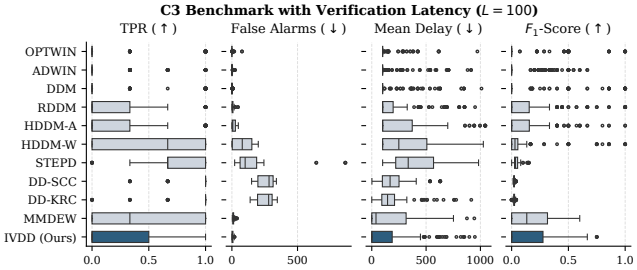


Figure 7: Averaged metrics on 120 synthetic datasets generated using CaDrift with different types of causal drift: $L = 100$.

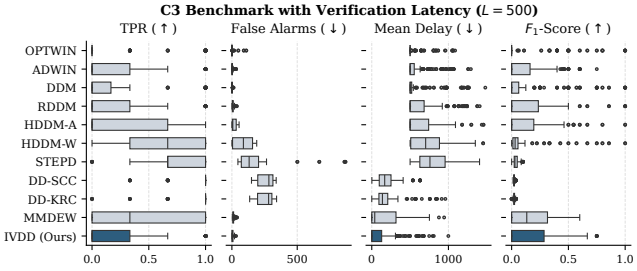


Figure 8: Averaged metrics on 120 synthetic datasets generated using CaDrift with different types of causal drift: $L = 500$.

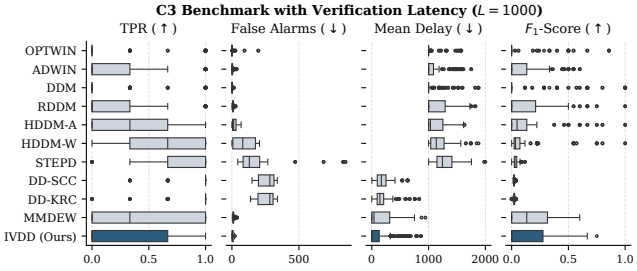


Figure 9: Averaged metrics on 120 synthetic datasets generated using CaDrift with different types of causal drift: $L = 1000$.