Concept Drift from a Causal Perspective

Anonymous authors
Paper under double-blind review

Abstract

Concept drift is a common phenomenon in real-world data streams, in which changes in the data-generating distribution can degrade predictive model performance. Most existing definitions characterize drift as changes in the joint distribution $P(\mathbf{x}, y)$, without distinguishing which component of the data-generating process has changed. In this work, we introduce a causal perspective on concept drift based on Structural Causal Models (SCMs). We propose a taxonomy that categorizes drift events by their causal origin, including changes in exogenous variables, endogenous mechanisms, confounders, and target-generating processes. Building on this framework, we develop an SCM-based data stream generator that simulates controlled mechanism-level drift events. Our experiments empirically characterize the distributional effects of each drift type and show that drifts with different causal origins induce distinct patterns of distribution shift and predictive behavior. Furthermore, by integrating causal discovery methods, we use our framework to construct data streams grounded in real-world dependency structures, enabling more realistic and informative evaluation scenarios. We also demonstrate that leveraging the generated data can improve downstream performance. These results highlight the importance of accounting for causal structure when studying and evaluating adaptive learning methods, and establish a foundation for causally-aware evaluation in non-stationary environments.

1 Introduction

Concept drift is a pervasive challenge in real-world data streams, where the statistical properties of data evolve over time (Lu et al., 2019; Hinder et al., 2024). Such changes can significantly degrade the performance of predictive models deployed in dynamic environments, including census analysis (Chakrabarty & Biswas, 2018), fraud detection (Hernandez Aros et al., 2024), and social media analysis (Yogi et al., 2024). As a result, a large body of research has focused on detecting, characterizing, and adapting to concept drift in streaming settings (Barboza et al., 2025; Paim & Enembreck, 2025; Kurian & Allali, 2024).

Most existing definitions characterize concept drift as any change in the joint distribution $P(\mathbf{x}, y)$ (Gama et al., 2014; Lu et al., 2019; Hinder et al., 2024). While this probabilistic view captures a broad family of distribution shifts, it does not reveal which components of the data-generating process have changed. Consequently, drift events stemming from fundamentally different causes may appear indistinguishable at the level of $P(\mathbf{x}, y)$, even though their implications for learning and adaptation can be substantially different.

This limitation is closely related to challenges studied in out-of-distribution (OOD) generalization (Arjovsky et al., 2020). A growing body of work argues that robust generalization to distribution shifts requires identifying invariant causal mechanisms rather than relying on spurious correlations that may change across environments (Arjovsky et al., 2020; Schölkopf et al., 2021; Eastwood et al., 2022). A classic example is that of an image classifier trained to recognize cows, but that inadvertently relies on the presence of green pastures (Arjovsky et al., 2020). Such a model may fail on images of cows standing on beaches, not because the concept has changed, but because the causal structure that generated the data differs across environments.

We can build this intuition about mechanism changes using the graph in Figure 1. Consider that T is a treatment variable corresponding to a drug dosage, and the outcome Y is the effect of the dosage on a patient. The confounder C (e.g., patient demographics) creates spurious correlations by influencing both the dosage T

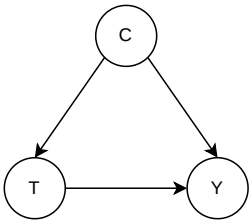


Figure 1: A graph exemplifying a treatment-outcome relationship with a confounder variable.

and the outcome Y. With continued use, a patient may develop physiological resistance to the drug (English & Gaur, 2010), which characterizes a change in the cause–effect relationship between T and Y.

These observations suggest that concept drift is fundamentally a *causal* phenomenon: it arises from changes in the structural mechanisms that generate the data. Within the framework of Structural Causal Models (SCMs) (Peters et al., 2017), data are generated by a set of structural equations defining how each variable depends on its causes. From this perspective, drift corresponds to changes in one or more components of this system, such as the distribution of exogenous variables, the functional mechanisms relating variables, or the structure of the causal graph itself. Different types of mechanism changes may induce similar distributional shifts in $P(\mathbf{x}, y)$, yet they may require fundamentally different adaptation strategies. Existing approaches categorize drift through statistical discrepancies or model-dependent signals without distinguishing between changes in the underlying data-generating mechanism. As a result, they provide limited insight into what type of drift occurred and how learning algorithms should respond.

In this work, we propose a causal taxonomy of concept drift grounded in SCMs. We categorize drift events according to their causal origin, distinguishing between exogenous drift, confounder drift, endogenous drift, target drift, and structural drift. Each category corresponds to a specific type of change in the underlying data-generating mechanisms.

Building on this taxonomy, we present an SCM-based data stream generator that enables controlled simulation of mechanism-level drift events, which we call *Causal Drift Generator* (CaDrift). Unlike existing synthetic generators, which typically simulate drift by arbitrarily modifying decision boundaries or feature centroids (Street & Kim, 2001; Bifet et al., 2009; Komorniczak, 2025), our approach models drift as changes in structural mechanisms of the data-generating process, and simulates time dependence.

Finally, we empirically characterize the distributional effects of each drift type using CaDrift. Our experiments show that drift events with different causal origins induce distinct patterns of change in marginal and conditional distributions, as well as differing impacts on predictive performance. Furthermore, we show that leveraging CaDrift for data augmentation improves downstream performance on real-world datasets. These results highlight the importance of reasoning about causal mechanisms when studying concept drift and evaluating adaptive learning methods. We argue that a causal perspective enables more principled analysis of distributional changes and can inform the development of more robust learning algorithms under non-stationarity (Schölkopf et al., 2021). CaDrift's source code is available in our GitHub repository[1].

Our main contributions are as follows:

- We introduce a causal taxonomy of concept drift grounded in structural causal models.
- We develop a synthetic data stream generator capable of simulating mechanism-level drift events and temporal dependencies.
- We empirically validate the proposed taxonomy, showing that different drift types induce qualitatively different effects on marginal and conditional distributions, as well as on downstream model performance.

[1]Code available in supplementary material during review.

- We demonstrate the practical utility of our framework by applying it to data augmentation, showing improved performance in streaming scenarios.

2 Background

Notation. Let $\mathbf{x} = (X_1, \ldots, X_d) \in \mathcal{X} \subseteq \mathbb{R}^d$ denote a d-dimensional feature vector, and let $y \in \mathcal{Y}$ denote the target variable (or label). A data stream is defined as an ordered sequence of instances

$$S = (\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots,$$

where $(\mathbf{x}_t, y_t)$ corresponds to the observation arriving at time step t. We denote by $P^{(t)}(\mathbf{x}, y)$ the joint data-generating distribution at time t, with corresponding marginals $P^{(t)}(\mathbf{x})$ and conditionals $P^{(t)}(y \mid \mathbf{x})$.

Concept Drift. *Concept drift* (Gama et al., 2014) is a prevalent phenomenon in data stream mining, characterized by changes in the data distribution over time. Formally, drift occurs when $P^{(t)}(\mathbf{x}, y) \neq P^{(t+\delta)}(\mathbf{x}, y)$, for some $\delta > 0$ (Lu et al., 2019; Hinder et al., 2024). As a convention, we denote data distributions at different moments in time as $P'(\mathbf{x}, y)$.

Concept drift is usually divided into two types (Gama et al., 2014; Lu et al., 2019): *real concept drift*, also known as *distributional shift*; and *virtual concept drift*, or *covariate shift*. A *distributional shift* happens when $P(y \mid \mathbf{x}) \neq P'(y \mid \mathbf{x})$ (Lu et al., 2019). This implies that the relationship between features and label evolves over time. Consequently, a model trained under distribution $P(\mathbf{x}, y)$ may become suboptimal under $P'(\mathbf{x}, y)$, even if the feature distribution remains unchanged. In this case, the predictive function itself must adapt.

Covariate Shift happens when $P(\mathbf{x}) \neq P'(\mathbf{x})$, i.e., the data distribution changes in the feature space but the posterior probability $P(y \mid \mathbf{x})$ remains unaffected (Lu et al., 2019). As an example, think of an object detection model that has been trained to detect cars. If this model were trained using data only from sunny days, it would never see cars in rainy or snowy conditions. However, the "true" concept definition of what a car is remains unchanged regardless of weather conditions. Other types of *concept drift* are usually subtypes of either *distributional* or *covariate shift*.

Structural Causal Models. SCMs Peters et al. (2017) are defined as follows:

Definition 2.1 (Structural Causal Model)**.** A *structural causal model* (SCM) $\mathcal{M}$ over a set of variables $V = \{X_1, \ldots, X_d\}$ consists of a collection of structural assignments

$$X_i := f_i(\mathrm{pa}_i, U_i), \quad i = 1, \ldots, d, \tag{1}$$

where $\mathrm{pa}_i \subseteq V \setminus \{X_i\}$ denotes the set of parents of X_i in the associated causal graph, and U_i are jointly independent exogenous noise variables such that $U_i \perp\!\!\!\perp U_j, \forall i \neq j$.

SCMs allow us to model complex cause–effect relationships between variables and the target, guided by deterministic effect mapping functions f_i.

Interventions. With SCMs, we can reason about the effects of interventions on variables in the causal graph. Broadly, interventions can be categorized into two types: *hard interventions* and *soft* (or *structural*) interventions. Hard interventions correspond to forcibly assigning a value to a variable, thereby removing the influence of its parents and effectively cutting all incoming edges in the causal graph. Considering an original graph structure in Figure 2a, a hard intervention is exemplified in Figure 2b, where we apply a hard intervention to the feature X_2. This is formalized using Pearl's do-notation (Pearl, 2009): $P(y \mid \mathrm{do}(X_2 = x))$, where the do-operator indicates that the variable X_2 is set to the value x, independently of its usual causal mechanisms.

With soft interventions, the causal edges are not cut off from the graph. Instead, we modify the structural assignment of a variable, i.e., we replace f_i with a different function f_i', while keeping the same set of parents, as illustrated in Figure 2c. This results in a new data-generating process where the mechanisms relating causes to effects have changed, but the underlying causal graph remains intact.

The framework of SCMs allows us to reinterpret concept drift through a causal lens. In this view, different types of drift correspond to changes in specific components of the causal model. For instance, covariate shift can be associated with (soft) interventions on upstream variables (i.e., causes of $\mathbf{x}$), which modify $P(\mathbf{x})$ while leaving the conditional mechanism $P(y \mid \mathbf{x})$ unchanged. On the other hand, distributional shift can be understood as interventions on the structural mechanism generating y, directly affecting $P(y \mid \mathbf{x})$. We explore this further in the next section.

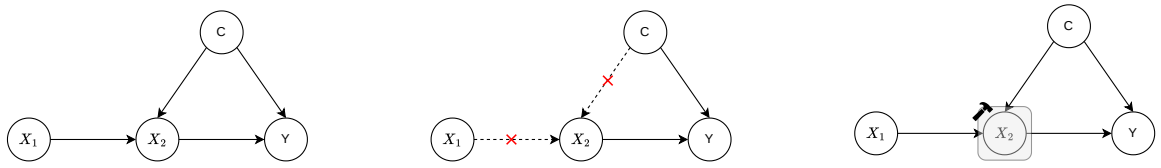
3 Concept Drift from a Causal Perspective

In this work, we consider that concept drift may originate from either observable or unobservable (latent) factors. We define *explicit concept drift* as changes affecting observable variables, since these directly alter the cause–effect relationships among the observed components of the data-generating process.

In contrast, *latent concept drift* originates from unobserved variables that are not accessible to the learner. Although these changes occur in latent factors, they can still influence observable associations indirectly. For example, price fluctuations or user preferences may be driven by numerous unmeasured factors, making it impractical to fully specify all relevant variables in real-world environments.

Both *explicit* and *latent drift* can induce downstream changes in the causal graph, altering the joint distribution $P(\mathbf{x}, y)$ in different manners depending on which node has drifted. Within a Structural Causal Model (SCM), concept drift occurs when either the distribution of exogenous variables or the mechanisms governing endogenous variables change. Formally, this corresponds to $\mathcal{M}t \neq \mathcal{M}t + \delta$.

This formulation reflects the idea that, when the underlying data-generating process evolves, the corresponding SCM must also change. We categorize different types of concept drift within the SCM framework according to which variables are affected by the drift. The categories can be found in Figure 3. Our explanations highlight how different forms of drift emerge through alterations to the cause--effect relationships encoded in the original graph.



(a) Original graph structure.

(b) Example of a hard intervention on the node X_2.

(c) Example of a soft intervention on the node X_2.

Figure 2: Graphs exemplifying hard and soft interventions on the node X_2

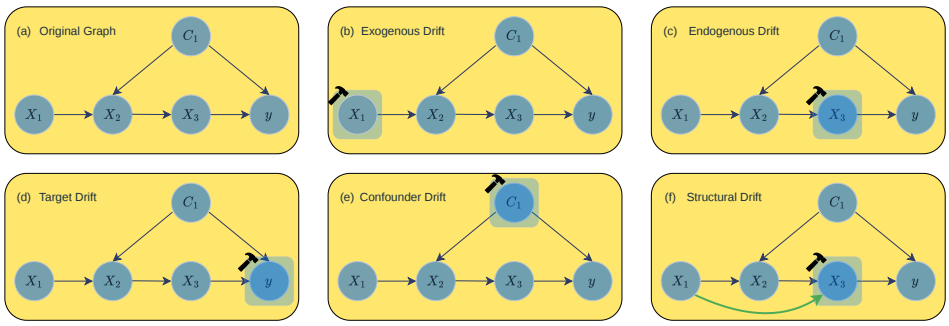


Figure 3: Causal taxonomy of concept drift.

Exogenous drift occurs when the distribution of exogenous variables changes over time, i.e., $P(U) \neq P'(U)$, where U denotes an exogenous variable in the SCM. In practice, exogenous variables are typically unobserved. Therefore, we focus on drift events that affect observable root nodes, whose values are directly determined by exogenous inputs. Modeling drift at root nodes allows us to represent changes in upstream environmental factors while keeping the structural mechanisms between variables unchanged.

This type of drift resembles *virtual concept drift*, or *covariate shift*. The graph representation of *exogenous drift* is illustrated in Figure 3b, where the mechanism sampling the root node X_1 changes. Importantly, the cause–effect relationships between variables remain intact.

Endogenous drift occurs when the mechanism of internal, or endogenous variables, changes, or in other words, there is a change in how one variable affects another. To define this mathematically, we should consider the effect function of a node X_i: $f_i^{(t)}(pa_i)$, such that pa_i are the parents of the node X_i. *Endogenous drift* takes place when the function $f_i(\cdot)$ changes. Hence, $f_i^{(t)}(pa_i) \neq f_i^{(t+\delta)}(pa_i)$. This is illustrated in Figure 3c. This type of drift produces a downstream effect in the causal graph, which produces effects on the marginal distribution of the X_i-th node descendants.

Essentially, when $P(X_i \mid pa)$ changes, the perturbation naturally propagates to the descendants. By the law of total probability, both *exogenous* and *endogenous drifts* can change the marginal distribution:

$$P(X_j) = \sum_{pa_j} P(X_j \mid pa_j) P(pa_j), \quad \forall X_j \in \text{descendants}(X_i), \tag{2}$$

because either the conditional mechanism $P(X_i \mid pa_i)$ changes (*endogenous drift*) or the upstream marginal $P(pa_i)$ changes (*exogenous drift*), and both propagate through the factorization of the joint distribution.

Endogenous drift potentially changes $P(y \mid a)$, where a is the set of ancestor nodes of X_i, such that X_i is the drifted node. In contrast, $P(y \mid X_j), \forall X_j \in \text{descendants}(X_i)$ are kept intact, assuming the target mechanism itself remains unchanged. To clarify this, we use the following SCM as an example:

$$\text{temperature}(X_1) \rightarrow \text{plant growth}(X_2) \rightarrow \text{yield}(y).$$

Suppose an *endogenous drift* occurs at node X_2, changing how temperature influences plant growth. Formally, we can say that $P(X_2 \mid X_1) \neq P'(X_2 \mid X_1)$. Since y is a descendant of X_2, its conditional distribution given X_1 changes: $P(y \mid X_1) \neq P'(y \mid X_1)$, while the mechanism for the target $P(y \mid X_2)$ remains unaltered. Thus, *conditional distributions of descendants of the drifted nodes only change when conditioned to the ancestors of the drifted node*. This leads us to the following proposition:

Proposition 3.1 (Propagation of Endogenous Drift)**.** Consider an SCM $\mathcal{M}$ and suppose an *endogenous drift* at node X_i, such that the structural mechanism f_i changes while all other mechanisms remain invariant. Assume in particular that the target mechanism f_y is unchanged. Then:

1. For any ancestor $a \in \text{ancestors}(X_i)$, the conditional distribution $P(y \mid a)$ may change across concepts due to the altered mediation through X_i, unless there is directional separation (d-separation) between a and X_i.

2. The structural conditional distribution $P(y \mid pa_y)$ remains invariant.

The proof can be found in Appendix A.

Target drift (Figure 3d) is a special case of *endogenous drift* in which the affected node is the target variable itself, i.e., $f_y(pa_y) \neq f'_y(pa_y)$ – hence, we discard the second assumption in Proposition 3.1. This causes the posterior probability conditioned on the target node changes for every feature, i.e., $P(y \mid \mathbf{x}) \neq P'(y \mid \mathbf{x})$, such that $\mathbf{x} = \{X_1, \ldots, X_d\}$, and d is the number of nodes in the graph that are not the target variable. If the target y is not a leaf, it also changes $P(\mathbf{x})$. *Target drift* produces more drastic changes to the decision process, as the change takes place directly on how direct parents of the target variable influence the outcome:

Corollary 3.1 (Target Drift)**.** Suppose an *endogenous drift* occurs at the target node y, changing its structural mechanism f_y. Let $\mathbf{x}$ be the set of all other observable variables. If y is a leaf node (i.e., it has no descendants), then the conditional distribution $P(y \mid \mathbf{x})$ changes across concepts, while the marginal distribution of the features $P(\mathbf{x})$ remains invariant. Conversely, if y is not a leaf node, $P(\mathbf{x})$ may also change, as the drift propagates to any descendants of y contained within $\mathbf{x}$.

Confounder drift occurs when the drift affects a variable that lies on a backdoor path (Pearl, 2009) to the target node. In Figure 3e, the mechanism sampling the confounder C_1 changes, which might induce a spurious change in the observed association between the node X_3 and the target y. Since C_1 induces a spurious (non-causal) association between X_2 and y, changes in its mechanism alter the observed correlations without modifying the direct causal mechanism of the target, given its true causes.

When such confounders are unobserved, no learner can explicitly condition on an appropriate adjustment set to block the backdoor path (Pearl, 2009). Consequently, models may adapt to shifts in spurious correlations rather than to genuine changes in the target-generating mechanism. Even when confounders are observed, most stream learners optimize predictive performance without causal constraints, and may therefore rely on associations induced by the backdoor path.

A practical example arises in medical diagnosis. Suppose X represents a treatment and y patient recovery, while an unobserved confounder C corresponds to disease severity. Patients with more severe conditions are both more likely to receive the treatment and less likely to recover. If the distribution of disease severity changes over time, the observed association between treatment and recovery may change even though the causal effect of the treatment remains unchanged.

From a causal standpoint, confounder drift alters observed associations while leaving the causal effect of true parents on the target invariant. However, when the confounder is unobserved, this invariance is not statistically identifiable from observational data alone (Pearl, 2009).

Proposition 3.2 (Effects of Confounder Drift)**.** Consider an SCM where an unobserved confounder C acts as a common cause for an observable variable X and the target y, forming a backdoor path $X \leftarrow C \rightarrow y$. Suppose a drift occurs such that the marginal distribution of the confounder changes $P(C) \neq P'(C)$, while the structural mechanisms f_X and f_y remain invariant. Hence:

1. The observable conditional distribution $P(y \mid X)$ changes across concepts due to the altered spurious association.
2. The conditional causal effect, denoted by the interventional distribution $P(y \mid \mathrm{do}(X), C = c)$ remains invariant.

The proof referring to the propagation of *confounder drift* can also be found in Appendix A.

Remark. While the conditional causal effect remains invariant under *confoudner drift*, it is worth noting that the marginal causal effect (or average treatment effect) does change under confounder drift. The interventional distribution is defined by Pearl's backdoor adjustment (Pearl, 2009):

$$P(y \mid \mathrm{do}(X)) = \sum_{c} P(y \mid X, C = c) P(C = c). \tag{3}$$

Because $P(C)$ shifts to $P'(C)$, the outcome of an intervention $P(y \mid \mathrm{do}(X))$ changes, even though the specific mechanism generating y remains fundamentally unchanged.

Structural Drift happens when edges in the causal graph change, i.e., the set of parents that cause a node changes. It may involve adding or removing causal edges between variables. Mathematically, we write this as $pa_i^{(t)} \neq pa_i^{(t+\delta)}$.

For example, a new tax policy may begin to affect consumer demand for a product. Initially, demand D may depend only on the product price P, i.e., $P \rightarrow D$. After the policy change, the tax variable T may also directly affect demand, yielding the updated structure $P \rightarrow D \leftarrow T$. In this case, the parent set of D changes

from $pa_D = \{P\}$ to $pa_D = \{P, T\}$, which characterizes a structural drift event. Figure 3f shows an example in which the node X_1 becomes a cause of X_3.

Most drift types in our taxonomy can be interpreted as soft interventions on the underlying SCM, in which structural mechanisms change while the graph structure remains fixed. Structural drift, in contrast, corresponds to interventions that modify the graph itself by altering parent sets.

In Table 1, we position the drift types defined through the causal perspective in contrast to classic and well-known groups of *concept drift*, from the probabilistic perspective.

Table 1: Contrasting classical concept drift taxonomy with concept drift from a causal perspective.

Classical concept drift		Causal perspective
$P(X) \neq P'(X)$		
Virtual concept drift Covariate shift	$\leftrightarrow$	Exogenous Drift
$P(y \mid X) \neq P'(y \mid X)$		
Real concept drift Distributional shift	$\leftrightarrow$	Endogenous drift
	$\leftrightarrow$	Target drift
	$\leftrightarrow$	Confounder drift
	$\leftrightarrow$	Structural drift

Rate of change. In addition to drift categorization in terms of its intricate factors, the literature also categorizes drifts according to their rate of change, i.e., drifts may happen abruptly, incrementally, or gradually (Lu et al., 2019).

Abrupt drift corresponds to an instantaneous change in the underlying SCM, i.e., $\delta = 1$. *Incremental* and *gradual drifts* involve a transition period ($\delta > 1$), during which the data-generating process evolves over time.

From a causal perspective, *incremental drift* corresponds to a smooth, continuous change in the parameters of one or more structural equations, such that the SCM $\mathcal{M}$ is slightly modified at each time step. In contrast, under *gradual drift*, two distinct SCMs coexist: a source model $\mathcal{M}_o$ and a target model $\mathcal{M}_n$ corresponding to the new concept. During the transition period δ, data samples are generated by either $\mathcal{M}_o$ or $\mathcal{M}_n$.

Concept drift might also present recurrence, such as the change of seasons during the year. Under the SCM framework, recurrence is simulated by retrieving a previously observed state of the data-generating process, i.e., by reinstating an earlier SCM $\mathcal{M}_{t'}$. This corresponds to the reactivation of past causal mechanisms, thereby making previously valid cause-and-effect relationships relevant again.

4 Generating Data Streams with Time-dependent Structural Causal Models

Building on the SCM framework, we simulate time-dependence across generated data samples by using autoregressive (AR) noise, sinusoidal seasonality, and an exponentially weighted moving average (EWMA) (Roberts, 1959). All of these components induce continuous non-stationarity on synthetic data samples. With them, we introduce CaDrift, which generates synthetic time-dependent data streams capable of simulating the causal drift events defined in Section 3.

Definition 4.1 (Time-dependent Structural Causal Model)**.** A *time-dependent SCM* $\mathcal{M}$ over a set of variables $V = \{X_1, \ldots, X_d\}$ is defined by the structural assignments

$$X_i := f_i(\mathrm{pa}_i, U_i^{(t)}), \quad i = 1, \ldots, d, \tag{4}$$

where the noise variables follow an autoregressive process:

$$U_i^{(t)} = \rho U_i^{(t-1)} + \epsilon_i^{(t)}, \quad \epsilon_i^{(t)} \sim \mathcal{N}(0, \sigma^2), \quad \rho \in [0, 1]. \tag{5}$$

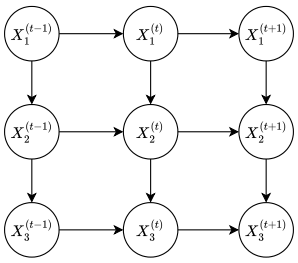


Figure 4: Scheme of a time-dependent SCM.

Here, ρ controls the temporal smoothness of the noise, determining the degree of dependence between consecutive samples. Higher values of ρ induce stronger temporal correlation, resulting in smoother transitions over time. The impact of this parameter is further analyzed in the experimental section.

In practice, this process induces a temporal causal dependence on two subsequent data samples, as shown in Figure 4 – hence, generated data samples are non-iid.

Next, let us define EWMA:

$$z_t = (1 - \alpha)z_{t-1} + \alpha X_t, \tag{6}$$

such that z_t denotes the current average, $\alpha \in [0, 1]$ is the smoothing parameter, and X_t the current observation at time t. We use EWMA at the root nodes to induce temporal correlation in the feature space. In order to introduce the autoregressive noise $U_i^{(t)}$ on the SCM definition, we define time-dependent SCM:

Lastly, we incorporate seasonality into the SCM framework to capture periodic patterns commonly observed in real-world data streams. Seasonality refers to recurring fluctuations that occur at regular intervals, such as daily or yearly cycles.

To model this behavior, we augment each structural assignment with an additive periodic component. Specifically, for variables exhibiting seasonality, we define:

$$X_i^{(t)} := f_i(\mathrm{pa}_i^{(t)}, U_i^{(t)}) + s_i^{(t)}, \tag{7}$$

where $s_i^{(t)}$ represents the seasonal component, defined as:

$$s_i^{(t)} = A_i \sin\left(\frac{2\pi t}{T_i} + \phi_i\right). \tag{8}$$

Here, A_i denotes the amplitude of the seasonal effect, T_i the period, and ϕ_i an optional phase shift (Hyndman & Athanasopoulos, 2018).

By combining autoregressive noise, EWMA smoothing, and seasonal components, the resulting time-dependent SCM is capable of generating non-stationary data streams exhibiting both incremental non-stationarity and recurring temporal patterns, in addition to the causal drift events presented in Section 3. This enables the simulation of realistic scenarios where both distributional and structural properties evolve over time.

Algorithm 1, in Appendix B shows the pseudocode for CaDrift's synthetic generation process. As input, CaDrift receives a graph $\mathcal{G}$, which can be any Directed Acyclic Graph (DAG). Furthermore, for drift events, it receives a drift schedule $\mathcal{D}$ containing the type of drift, as defined in Section 3, the time point t at which it happens, and the drift length $\delta \geq 1$.

5 Related Work

Concept Drift Adaptation. Standard approaches typically define concept drift as any change in the joint distribution $P(\mathbf{x}, y)$ (Lu et al., 2019; Hinder et al., 2024). In practice, this broad definition encompasses shifts in the marginal distribution $P(\mathbf{x})$, the conditional distribution $P(y \mid \mathbf{x})$, or both, each of which may impact predictive performance differently. A large body of work focuses on *error-driven adaptation*, where drift is inferred through changes in model performance. These methods monitor fluctuations in the prediction error over time (Agrahari & Singh, 2022), triggering adaptation mechanisms such as incremental retraining or ensemble updates (Gomes et al., 2017; Paim & Enembreck, 2025; Barboza et al., 2025). For instance, the Drift Detection Method (DDM) (Gama et al., 2004) monitors the online classification error and raises alarms based on statistically derived thresholds, distinguishing between warning and drift levels. Similarly, the Adaptive Windowing (ADWIN) detector (Bifet & Gavaldà, 2007) maintains a variable-length sliding window and continuously tests for significant differences between the averages of two sub-windows, providing guarantees on false positive rates. The McDiarmid Drift Detection Method (MDDM) (Pesaranghader et al., 2018) extends this idea by assigning higher weights to more recent instances and leveraging McDiarmid's inequality to detect changes. Despite their effectiveness, error-driven methods present notable limitations. Because they rely on labeled data, their responsiveness is inherently delayed in scenarios where labels are scarce or arrive with latency.

These limitations have motivated the development of *unsupervised* drift detection approaches, which operate directly on the input data distribution. For example, DD-SCC and DD-KRC (Agrahari & Singh, 2024) measure changes in feature relationships using the Spearman Correlation Coefficient (SCC) and the Kendall Rank Correlation (KRC), respectively. These methods assume that concept drift manifests as changes in the dependency structure among features, captured by correlation statistics computed over sliding windows. However, such correlation-based methods may be vulnerable to changes in the underlying causal relationship, such as *confounder drift*, which can trigger false alarms when spurious correlations shift. Another line of work explores *uncertainty-based* detectors. Methods such as Uncertainty Drift Detector (UDD) (Baier et al., 2022) estimate predictive uncertainty from neural networks and monitor its evolution over time as a proxy for drift. PUDD (Lu et al., 2025) introduces the Prediction Uncertainty Index (PU-index), which monitors the classifier's predictive uncertainty to identify concept drift before it manifests as significant drops in accuracy. However, like error-driven approaches, uncertainty-based methods rely on model-dependent signals that may not fully capture changes in the underlying data-generating process.

Most drift detectors operate by comparing statistics, such as error rates or feature distributions, across consecutive windows. While effective at identifying distributional changes, these approaches treat the data-generating process as a black box. Gower-Winter et al. (2026) show that perceived drift may arise from window partitioning artifacts rather than genuine changes in the underlying process. Moreover, existing detectors primarily indicate *when* drift occurs, but provide limited insight into *where* and *how* the data-generating mechanism has changed. Taken together, these methods characterize drift through statistical discrepancies or model-dependent signals, without explicitly modeling the causal mechanisms that generate the data. Our causal taxonomy provides the foundation to address this limitation by explicitly defining drift as mechanistic changes in the underlying system.

Causality and Concept Drift. Recent work has begun to study concept drift through a causal lens, leveraging distributional changes to gain insight into underlying data-generating mechanisms. For example, Komnick et al. (2025) propose a framework for post-hoc explanation of drift events, attributing observed performance changes to shifts in specific causal mechanisms and providing actionable insights to users. Causal mechanisms that vary across environments have also been studied in the context of causal discovery. Zhang et al. (2017) introduce Causal discovery from nonstationary/heterogeneous data (CD-NOD), which exploits distributional changes to recover the underlying DAG, under the assumption that such changes correspond to shifts in structural mechanisms. Similarly, Latent Intervened Non-stationary learning (LIN) considers non-stationary settings with latent interventions, where changes in the data-generating process are not directly observed but can be inferred from distributional variation (Liu & Kuang, 2023). While these approaches use distributional changes to explain drift or recover causal structure, they do not provide a systematic characterization of different types of drift. In contrast, our work introduces a causal taxonomy of concept

drift, explicitly modeling drift as interventions on SCMs, enabling controlled analysis of their effects on distributions and learning performance.

Causal domain generalization. Causal approaches to domain generalization assume that data are generated by an underlying SCM, and that certain mechanisms, in particular the target-generating mechanism, remain invariant across environments (Arjovsky et al., 2020; Yao et al., 2025). Methods such as Invariant Causal Prediction (ICP) (Peters et al., 2016) and Invariant Risk Minimization (IRM) (Arjovsky et al., 2020) aim to identify predictors that remain stable under distribution shifts by exploiting these invariances. Subsequent work has relaxed strict invariance assumptions through risk extrapolation, quantile-based risk minimization, and representation learning approaches (Krueger et al., 2021; Eastwood et al., 2022; Rosenfeld et al., 2021; Schölkopf et al., 2021). Within our taxonomy, such assumptions correspond to settings where the target-generating mechanism is preserved, such as *exogenous* or *confounder drift*, and, in some cases, *endogenous* and *structural drift*. In these regimes, invariance-based strategies are well-justified and can yield robust generalization. In contrast, *target drift* explicitly violates the invariance of the predictive mechanism, highlighting that the effectiveness of domain generalization methods depends critically on the underlying type of causal drift – an aspect not explicitly captured in existing frameworks.

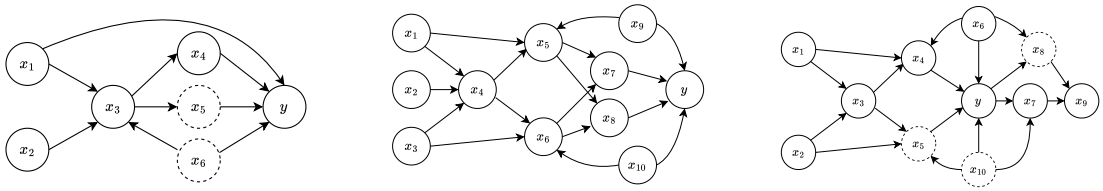
To address shifts that violate strict invariance, recent work builds on the Independent Causal Mechanism (ICM) principle and the Sparse Mechanism Shift (SMS) hypothesis (Schölkopf et al., 2021; Chen et al., 2024). The SMS hypothesis posits that distribution shifts typically affect only a small subset of structural mechanisms, while others remain stable, suggesting that robust generalization requires localized adaptation rather than global retraining (Bengio et al., 2019). However, most domain generalization methods assume access to discrete, labeled environments during training, whereas real-world concept drift usually happens continuously without explicit boundaries between environments. By formalizing drift as mechanism-level interventions, our taxonomy and the CaDrift framework set the foundation for controlled evaluation of sparse adaptation and causal transfer strategies in continuous streaming settings.

Synthetic Data Stream Generators. Synthetic data generators are widely used to evaluate learning algorithms under controlled concept drift scenarios. Classical generators, such as SEA (Street & Kim, 2001), Hyperplane (Hulten et al., 2001), and RandomRBF (Bifet et al., 2009), simulate drift by modifying ideal decision boundaries, feature distributions, or class priors over time. While these tools enable the evaluation of predictive accuracy under *concept drift*, they rely purely on probabilistic or geometric manipulations. Consequently, they lack the structural semantics necessary to simulate targeted causal interventions. In contrast, synthetic data generation based on Structural Causal Models SCMs has been widely adopted in causal inference and discovery, where data are generated from user-specified directed acyclic graphs and structural equations. Examples include simulation frameworks used in libraries such as DoWhy (Sharma & Kiciman, 2020), as well as benchmarking platforms like CauseMe (Runge et al., 2019), which enable controlled evaluation across diverse causal structures and functional mechanisms. However, these approaches are typically limited to static settings and do not explicitly model temporal dynamics or evolving data streams. CauKer (Xie et al., 2026) is an SCM-based generator that incorporates seasonality for training classification time-series foundation models. However, it lacks a formal taxonomy to distinguish among specific causal origins of drift.

6 Experiments

We evaluate whether the proposed SCM-based framework (CaDrift) generates data streams that reflect well-defined and distinguishable forms of concept drift. In particular, we aim to validate three key properties: **(i)** that different causal drift types induce distinct distributional changes in the data, **(ii)** that these changes can be identified through marginal and conditional analyses, and **(iii)** that these changes have a measurable impact on predictive performance. To this end, we analyze the statistical properties of the generated data streams and relate them to classifier behavior under drift.

After assessing the distributional impact of each causal drift event, we evaluate the serial correlation of samples generated by CaDrift in Section 6.2. Finally, in Section 6.3 we perform a case study in which we synthesize the real-world ELEC2 dataset with CaDrift, and use it for data augmentation in stream learners.



(a) DAG #1. (b) DAG #2. (c) DAG #3.

Figure 5: DAGs used to generate data samples for distribution analysis. Dashed nodes represent latent (unobservable) variables.

Dataset Generation Protocol. Synthetic datasets are generated using CaDrift. We use the handcrafted DAGs presented in Figure 5, which gives the basis for us to evaluate the effect of each drift type defined in Section 3. Each node X_i of the DAGs is associated with a structural equation:

$$X_i^{(t)} := f_i^{(t)}(\mathrm{pa}_i^{(t)}) + U_i^{(t)} + s_i^{(t)}. \tag{9}$$

To generate the target y for classification tasks, we use a prototype-based mapper, in which prototypes are randomly sampled according to the distribution of pa_y, such as in other SCM generators (Hollmann et al., 2025). Each class $y \in \mathcal{Y}$ is assigned to one or more prototypes. For each data sample, the classes are assigned to their nearest prototypes based on the Euclidean distance computed on pa_y. This design enables flexible, non-linear class boundaries while preserving dependence on the parent variables. Further details on prototype generation are provided in Appendix C, and drift event details performed for the experiments in this section can be found in Appendix C.1. Unless otherwise stated, drift events are abrupt.

Distributional impact analysis. To assess whether different drift types induce distinguishable changes in the data distribution, we analyze the evolution of marginal distributions over time by computing the Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) using sliding windows of 200 samples. For each window, we compare the distribution of samples from the initial (reference) concept with that in the current window.

For this analysis, we generate streams of 20,000 samples with a drift event introduced every 2,000 samples. This configuration provides sufficient observations per concept to obtain stable empirical estimates of MMD, while keeping the computational cost tractable given the quadratic complexity of kernel-based MMD estimation.

Conditional distribution analysis. Since marginal analysis alone cannot capture changes in the predictive relationship between features and the target, we additionally analyze shifts in the conditional distribution $P(y \mid \mathbf{x})$. To do so, we estimate predictive posteriors using a multinomial logistic regression classifier implemented in scikit-learn with the *lbfgs* solver and L2 regularization to approximate the conditionals $P(y \mid \mathbf{x})$. For each sliding window, we train a model on the current window and compute the conditional KL divergence (Lee & Lee, 2024; Kurian & Allali, 2024) between its predicted posterior probabilities and those produced by a model trained on the initial concept. The KL divergence is averaged over a fixed reference set to ensure comparability across windows.

This procedure captures changes in the predictive relationship between features and the target, allowing us to distinguish drift types that are indistinguishable under marginal analysis (e.g., target drift).

Performance evaluation. To evaluate the practical implications of the induced drift, we measure the predictive performance of the Hoeffding Tree (HT) classifier (Domingos & Hulten, 2000) under different drift scenarios in a test-then-train manner, considering a prompt label availability after the test, as well as when label availability is delayed by 100 samples, as in previous works (Gomes et al., 2017). We also couple the HT with the classic DDM detector (Gama et al., 2004).

For taxonomy validation, we restrict the analysis to stationary concepts without intra-concept temporal dependence (i.e., excluding autoregressive noise, EWMA, and seasonality). This ensures that observed

distributional changes arise exclusively from structural interventions associated with each drift type, isolating the causal impact of the induced drift.

6.1 The impact of causal drift events on distribution and performance

We analyze the effect of drift events using three complementary measures: marginal divergence (MMD), conditional divergence (KL), and prequential accuracy. The evolution of MMD, conditional KL divergence, and prequential accuracy over time for each DAG is presented in Figure 6. Results are averaged over 10 datasets generated by each DAG per drift type. Drift events are random adjustments to the weights of the linear functions that map each node, and the graph state is reverted to the original concept before each new drift event, allowing direct comparison with the first concept in the stream.

Exogenous drift (Figure 6a) impacts the marginal distribution measured by MMD, as well as the KL divergence, even though the cause–effect relationships between nodes remain intact. We observe an impact on prequential accuracy due to *exogenous drift* induced in DAG #2, whereas for DAGs #1 and #3 it usually remains stable, except for minor fluctuations when delayed feedback is applied.

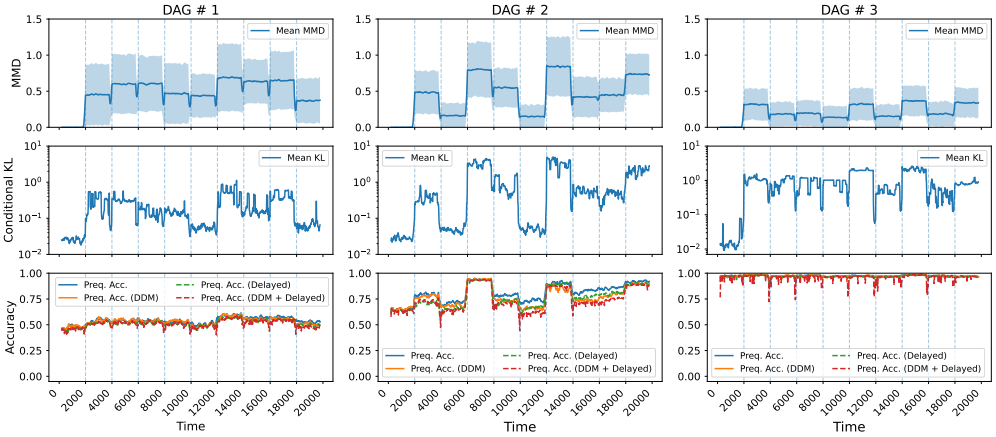
The drop in accuracy is more pronounced when we apply *endogenous drift* (Figure 6b). In this type of drift, we notice drops in accuracy followed by recovery, as the HT classifier incorporates sufficient samples from the new concept. Using DDM to actively adapt to *endogenous drift* gives different behaviors. We hypothesize that the effectiveness of active adaptation via DDM is determined by the degree of distributional overlap between concepts. In some instances of *endogenous drift*, the change creates an inconsistency between the concepts, where the optimal decision boundary for the new concept directly contradicts the previously learned one. In these cases, adaptation through detectors such as DDM is essential to clear the model's "memory" of the obsolete concept. In contrast, other events may simply push the data into a previously unobserved region of the feature space where the underlying causal mechanism remains locally consistent with the global rule. In such a scenario, the model faces a sample-complexity issue rather than a structural failure – therefore, incremental learning from new samples is more effective.

Confounder drift, in Figure 6c, also produces measurable marginal shifts. Its effect resembles that of exogenous drift with respect to the observed covariates, but its impact on the target variable is revealed only when analyzing conditional distributions. Drops in accuracy are observed, even though the inner and target nodes remained invariant. This highlights that marginal measures alone are insufficient to characterize the causal nature of drift, as *confounder drift* alters statistical associations without modifying the underlying generating mechanisms. Consequently, performance degradation under this type of drift is primarily a symptom of a model having learned statistical correlations rather than the invariant causal mechanisms.

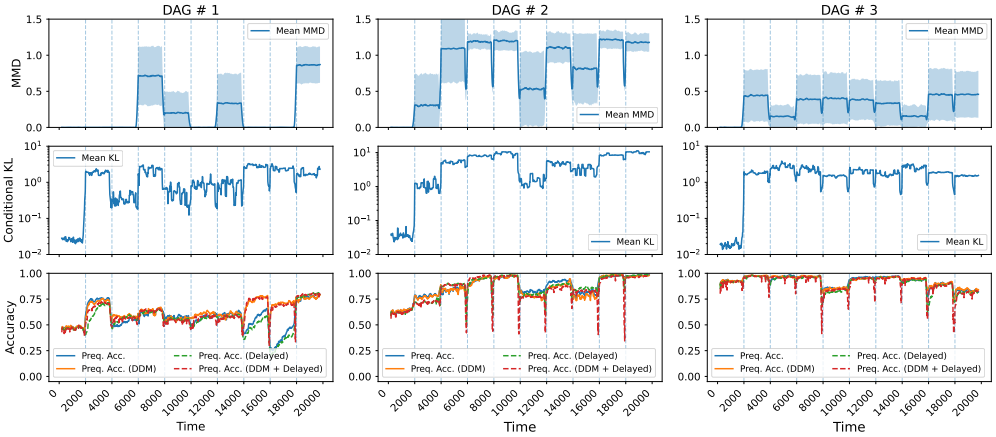
As expected, *target drift* (Figure 6d) does not modify covariate distributions, since the intervention affects only the conditional mechanism of the target variable. The effect of this drift type is perceived in the conditional KL divergence, and on the accuracy performance – an increase in KL divergence is associated with a decrease in accuracy on datasets generate by DAGs #1 and #2. Interestingly, *target drift* on data generated by DAG #3 produced no substantial accuracy degradation. As a matter of fact, DAG #3 seems to produce less challenging environments for the HT classifier. We attribute this to the presence of causal descendants of the target, which act as noisy proxies of y and may be exploited by the HT classifier during splitting, thus explaining why *target drift* did not lead to degradation in accuracy. This suggests that the presence of observable descendants of the target can partially mask *target drift*, reducing its impact on predictive performance.

Under *structural drift* (Figure 6e), we observe varying effects on both MMD and KL divergence: in some cases, both remain low despite noticeable drops in accuracy. In DAG #3, for instance, a sharp accuracy decline between samples 8,000 and 10,000 coincides with the removal of the edge $y \rightarrow X_7$. This supports the hypothesis that observable causal descendants can simplify the learning problem and, when present, mask the effects of *target drift*. Additional experiments on interventions in inner nodes are provided in Appendix F.

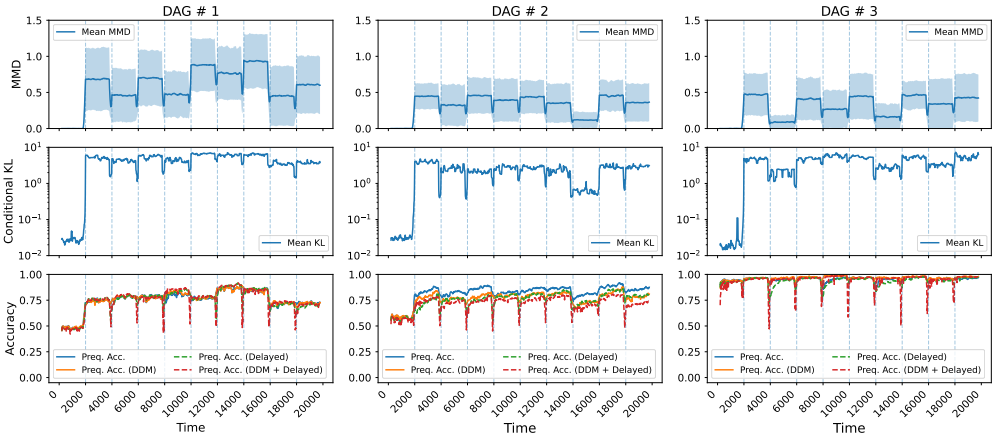
Overall, the results reveal that each drift type induces a characteristic and interpretable signature across marginal divergence, conditional divergence, and predictive performance, empirically validating the proposed causal taxonomy. More importantly, no single metric is sufficient to fully characterize drift: marginal measures



(a) Exogenous drift.

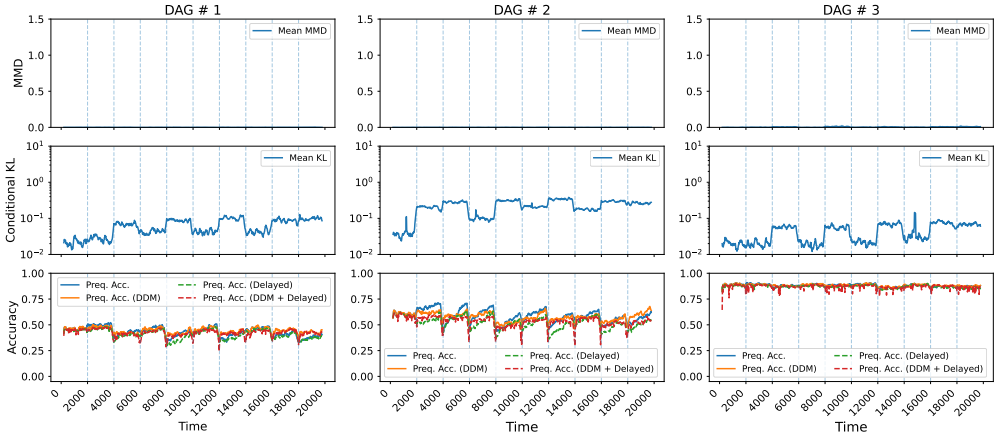


(b) Endogenous drift.

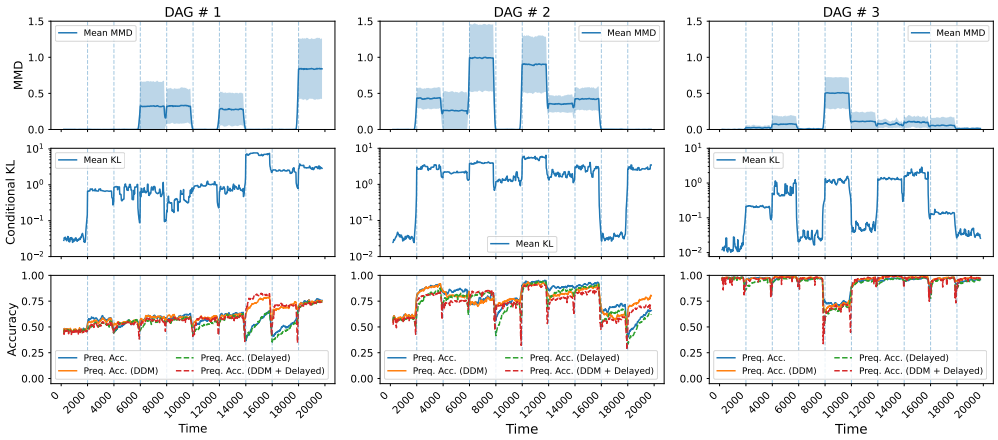


(c) Confounder drift.

Figure 6: MMD, KL divergence, and prequential accuracy over time in each DAG on causal drift types.



(d) Target drift.



(e) Structural drift.

Figure 6: MMD, KL divergence, and prequential accuracy over time in each DAG on causal drift (continued).

such as MMD fail to capture purely conditional changes, while conditional divergence alone does not reflect shifts in the input distribution, and predictive performance conflates both effects with model adaptation dynamics. This highlights the necessity of a joint analysis to correctly diagnose the nature of drift. In Table 2, we summarize the observed signatures in MMD, KL divergence, and accuracy performance of each drift type.

From a causal perspective, these findings demonstrate that the impact of drift on learning systems depends not only on the magnitude of distributional change, but on where the intervention occurs in the data-generating process. As a result, two drift events with similar statistical signatures may require fundamentally different adaptation strategies. This reinforces the central premise of our framework: understanding and modeling concept drift through a causal lens is essential for disentangling its effects and designing robust adaptive learning systems.

6.2 Autocorrelation Analysis

To analyze the temporal dependence induced by the proposed components, we generate synthetic streams from DAG #2 under four configurations: no temporal dependence, autoregressive (AR), EWMA, and seasonality. For each configuration, we generate 10,000 samples and compute the autocorrelation function (ACF) (Box et al., 2015) for the generated features averaged over 10 datasets, shown in Figure 7.

Table 2: Observed signatures of each causal drift type.

Drift Type	MMD	KL	Accuracy Impact
Exogenous	↑	↑	low to moderate
Endogenous	↑	↑	moderate to high
Confounder	↑	↑	moderate
Target	none	↑	moderate to high
Structural	variable	variable	variable

These plots illustrate the presence of serial correlation in the generated streams and how temporal dependence propagates through the causal structure. Because the target variable is generated from its parents through structural equations, temporal correlations present in upstream variables naturally propagate to the labels.

As expected, larger values of ρ increase the autocorrelation of both features and the target variable at early lags, reflecting stronger dependence between consecutive samples. The autocorrelation then gradually decreases with increasing lag, consistent with autoregressive processes in which the influence of past observations decays over time.

The EWMA parameter α also plays an important role in shaping the temporal dynamics. Small values of α (e.g., $\alpha = 0.05$) produce longer memory in the smoothed series, resulting in persistent autocorrelation across a larger number of lags. Conversely, larger values of α reduce this persistence by assigning greater weight to recent observations, thereby diminishing the time dependence.

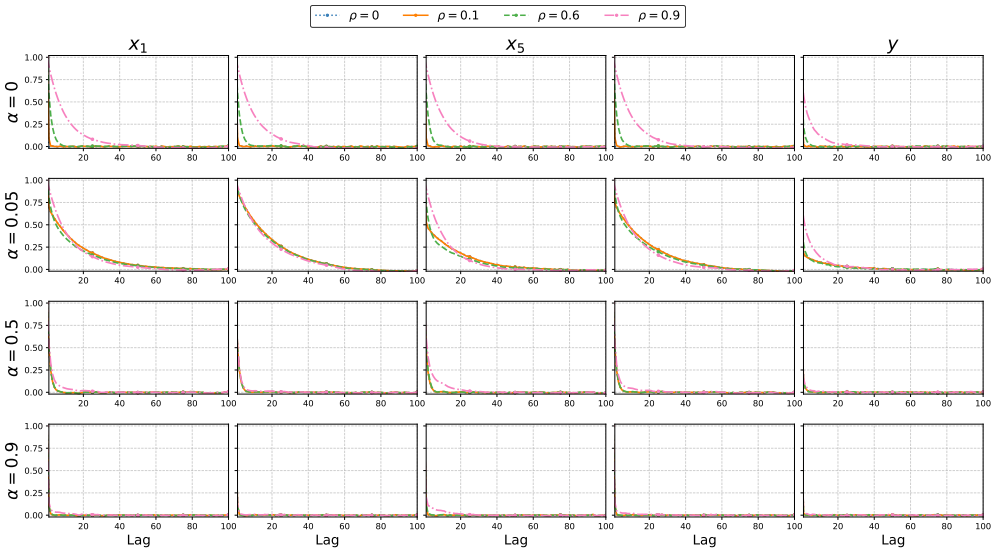


Figure 7: The impact of the α and ρ variables on the lagged autocorrelation function in DAG #2. Each row refers to a different value for α, and each column a different feature (X_1, X_4, X_5, X_8 and y). The y-axis refers to the autocorrelation, and the x-axis to the lag.

Figure 8 shows the Autocorrelation Function (ACF) plots when seasonality is introduced in the generation process. The hyperparameters of the seasonality component (Equation 8) for these plots are set to $A_i = 0.2$, $T_i = 50$, and $\phi = 0$. The resulting autocorrelation patterns exhibit periodic peaks characteristic of seasonal time series.

Because EWMA smoothing is applied to the root nodes of the DAG, the resulting seasonal patterns propagate through the causal structure while being progressively attenuated along the causal chain, due to the combined effect of intermediate transformations and noise terms. In practice, larger values of α tend to smooth the

seasonal signal more aggressively, reducing the strength of the induced temporal dependence in downstream variables.

Overall, the combination of autoregressive noise, EWMA smoothing, and seasonal components allows the proposed framework to generate a wide variety of temporally dependent data streams, capturing several forms of non-stationarity commonly observed in real-world streaming scenarios. This makes CaDrift a suitable framework for evaluating models under non-stationary streaming data.

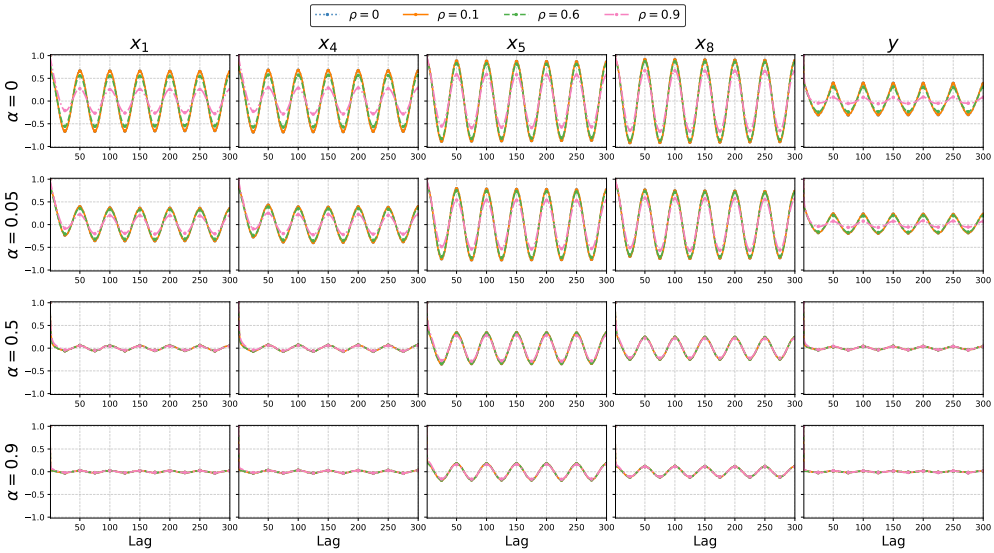


Figure 8: ACF plot on features with seasonality in DAG #2. The y-axis refers to the autocorrelation, and the x-axis to the lag.

To further investigate temporal dependence in data generated by CaDrift, we apply the Ljung–Box (LB) test (Ljung & Box, 1978), with detailed results for DAG #2 reported in Appendix H. The LB test rejects the null hypothesis of no autocorrelation when AR, EWMA, or seasonal components are included, confirming the presence of temporal dependence. In contrast, when these components are absent, the test does not reject the null, indicating no evidence of autocorrelation and consistency with i.i.d. samples.

6.3 Case Study: Synthesizing the Electricity Market dataset

In this section, we investigate whether causally synthesized data can improve the performance of online learners on the ELEC2 dataset. The Electricity Market dataset (ELEC2) (Harries et al., 1999) is widely used to evaluate data stream learning methods, in which the task is to predict whether the price will increase or decrease given demand- and supply-related features. The dataset contains time-ordered instances collected at 30-minute intervals, exhibiting temporal dependencies. Following common practice in data stream mining Losing et al. (2016), we ignore the features *date* and *nswprice*.

To infer the DAG for the ELEC2 dataset, we employ the Peter-Clark (PC) algorithm (Spirtes et al., 2000), a classic constraint-based causal discovery method. Although more recent approaches have been proposed, PC remains one of the most widely studied and commonly used algorithms in empirical causal discovery research, serving as a standard reference method in recent surveys and benchmark studies (Glymour et al., 2019; Hasan et al., 2024). The DAG inferred from the ELEC2 dataset can be found in Figure 9. We use the first half of the dataset (22,656 samples) to learn the DAG structure via the PC algorithm. To detect seasonality in the features, we use the Fast Fourier Transform (FFT) (Musbah et al., 2019).

For each inner node X_i in the DAG, we train a feedforward neural network (NN) that maps its parent variables to X_i, thereby approximating the causal mechanisms in the data. When the node is continuous (all endogenous nodes in the ELEC2 DAG), we fit an NN regressor. Otherwise (the target node y), we fit a classifier. Further, these learned mappers are used to sample instances in the generated stream. Details of the

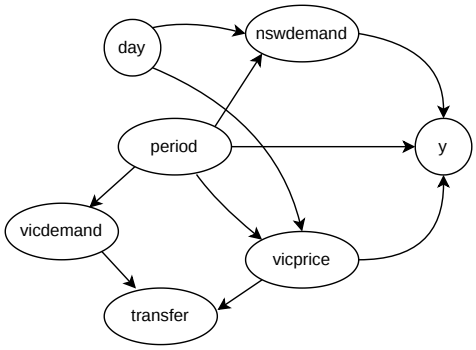


Figure 9: DAG inferred from the ELEC2 dataset through the PC algorithm.

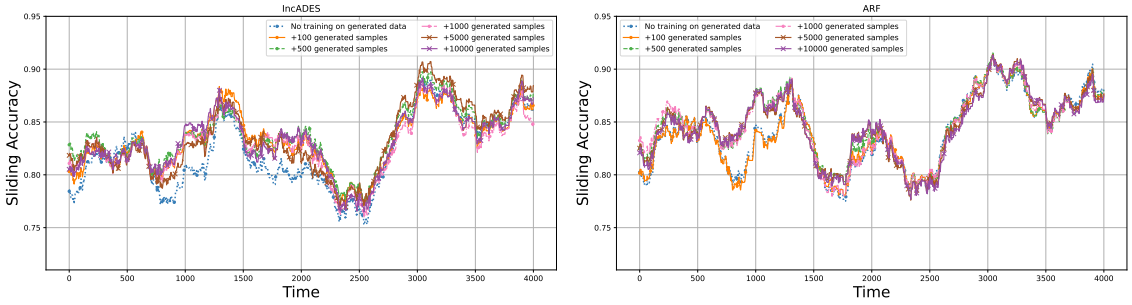
neural network hyperparameters used to map the features are provided in Appendix D. For the root nodes (*day* and *period*), we used different functions to simulate the behavior observed in the real-world ELEC2 dataset. These details, as well as additional experimental details in this section, are available in Appendix I. We provide ACF plots for the synthesized data compared to the original ELEC2 dataset in Appendix I.1, where we see that the root nodes closely follow the ACF plots of the real-world ELEC2. However, for many features, the data synthesized by CaDrift do not follow the same seasonal trend – the FFT only detected seasonality for the *period* feature, but not for other features that present seasonality.

We use two competitive online learners: **IncA-DES** (Barboza et al., 2025), an online dynamic ensemble selection method, and **Adaptive Random Forest (ARF)** (Gomes et al., 2017), an online ensemble. In addition to the online learners, we also include **TabPFNv2.5** (Grinsztajn et al., 2026), a transformer-based tabular classifier. For TabPFN, we adopt the data-stream setup described by Lourenço et al. (2025), which maintains a sliding context window. The learners' hyperparameters are reported in Appendix I.3. To isolate the effect of causal data augmentation, all experimental conditions are kept identical across both settings, with the only difference being the inclusion of n additional synthetic samples generated by CaDrift before starting the evaluation.

In Figure 10, we present the windowed prequential accuracy (Gama et al., 2014) without data augmentation and with CaDrift-based augmentation for different values of n. IncA-DES (Figure 10a), ARF (Figure 10b), and TabPFN (Figure 10c) exhibit improved predictive performance when trained with synthetic data when no delay is applied. For ARF, the prequential accuracy curves eventually converge, as the HT base learners are progressively updated with sufficient real data from the ELEC2 stream. The same happens with TabPFN, as the context window receives more samples from the real ELEC2 dataset. In contrast, the improvements for IncA-DES persist for longer periods.

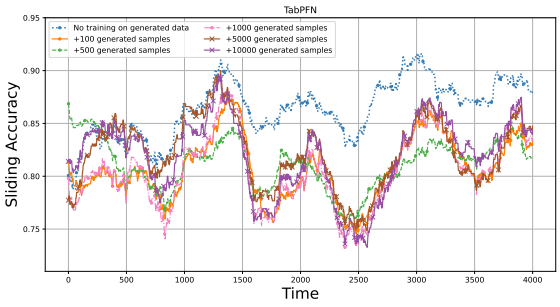
However, when we apply delayed feedback to the stream, the performance gain persists for fewer data samples, as we observe in Figure 11. Since the gain in accuracy is more short-term under delayed feedback, we plot a smaller subset of the stream. This result suggests that the generator synthesizes samples based on an outdated concept, causing the model to overfit to obsolete cause–effect relationships. Figure 12 supports this observation: although a large volume of synthetic data boosts early accuracy (within the first 100–200 data points), it eventually hinders adaptation, causing performance to degrade faster than the non-augmented baseline. Consequently, causal augmentation is most beneficial for immediate recovery but requires continuous real-world supervision to prevent long-term reliance on outdated distributions.

In Table 3, we present the average accuracy after data augmentation on the 500 subsequent samples (approximately 10 days in the ELEC2 dataset), and 100 subsequent samples on the delayed protocol. The improvement in accuracy achieved by data augmentation is evident: under immediate feedback, when $n = 1,000$, we observe increases of 3.1 percentage points in accuracy for ARF, 4.4 percentage points for IncA-DES when $n = 500$, and of 6.8 percentage points for TabPFN, also when $n = 500$. In this setup, performance peaks at moderate augmentation levels ($n = 500 - 1,000$). Interestingly, TabPFNv2.5's initial



(a) IncA-DES (immediate feedback).

(b) ARF (immediate feedback).



(c) TabPFN (immediate feedback).

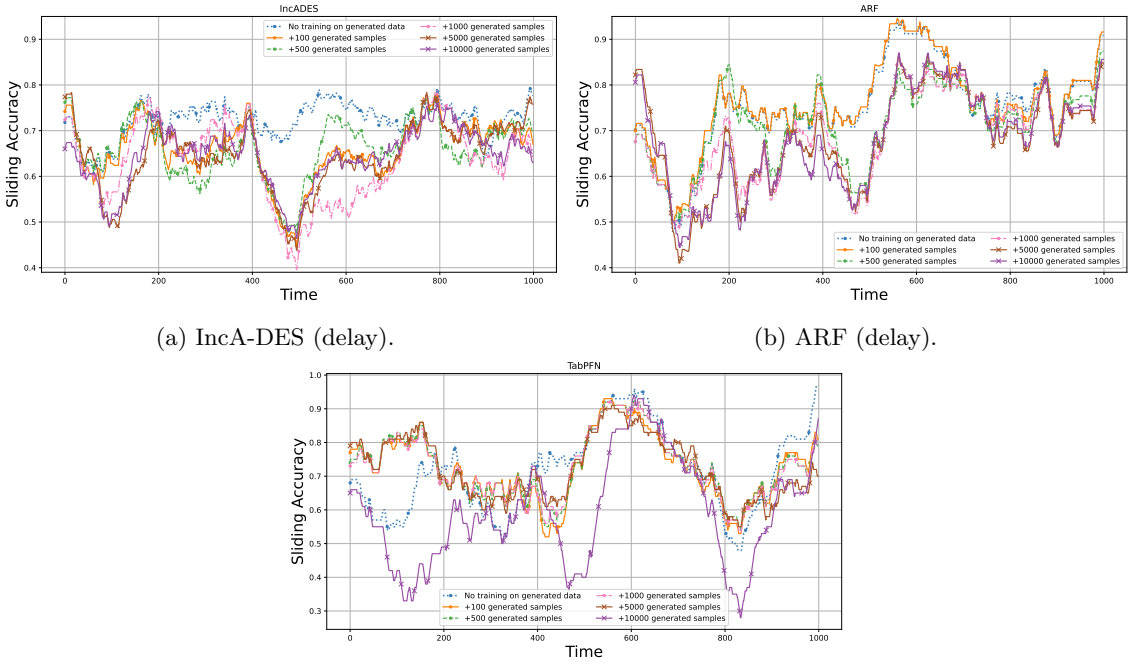
Figure 10: The impact of samples generated by CaDrift on IncA-DES, ARF, and TabPFN (immediate feedback).

performance when the context window consists only of synthetic samples $n = 10,000$ is greater than when using only the original ELEC2 ($n = 0$) – but eventually, the setup when $n = 0$ surpasses it. When we apply delayed feedback, the average accuracy on the 100 subsequent samples (2 days in the ELEC2 dataset) increases by 12 percentage points for ARF, 5.6 percentage points for IncA-DES, and 10.9 percentage points for TabPFNv2.5, all when $n = 5,000$. However, if the supervision of recent data samples is not provided, overfitting to a previous concept may degrade the model's performance more rapidly on IncA-DES and ARF. This behavior reflects the stability–plasticity dilemma in streaming settings. Synthetic data can improve short-term stability by reinforcing previously learned causal mechanisms, but it may reduce plasticity by anchoring the model to outdated concepts once the data-generating process changes. As a result, effective causal data augmentation requires balancing the use of generated samples for rapid recovery with the incorporation of new observations to enable timely adaptation.

It is important to note that causal discovery methods such as the PC algorithm (Spirtes et al., 2000) used in our experiments rely on assumptions (e.g., causal sufficiency and model specification) and may not recover the true underlying causal structure in all cases. As a result, the learned graph should be interpreted as an approximation of the data-generating process. Nevertheless, this approach allows us to construct structurally grounded generators that preserve the main dependencies present in the data. In some applications, human interventions can be done to mitigate potential inaccuracies in the recovered DAG structure – domain expertise is used to validate edges and orient directions that remain ambiguous.

6.4 Discussion

Our results show that causal drift events induce distinct, interpretable effects on both the data distributions and model performance. Moreover, the combination of marginal and conditional measures proved essential for disentangling different drift types, as several scenarios were indistinguishable when considered from a single perspective. *Confounder drift* demonstrates that accuracy degradation does not necessarily arise from changes in $P(y \mid \mathbf{x})$, but can instead be driven by shifts in spurious associations between the target and observed variables.



(a) IncA-DES (delay).

(b) ARF (delay).

(c) TabPFN (delay).

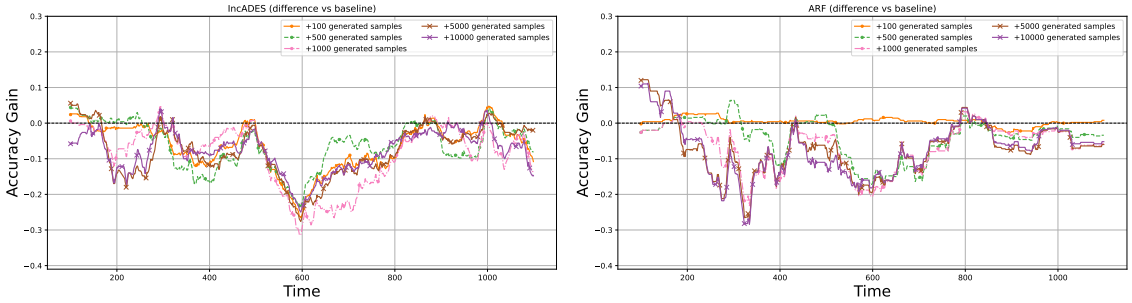
Figure 11: The impact of samples generated by CaDrift on IncA-DES, ARF, and TabPFN (delayed feedback).

Table 3: Average accuracy (%) with and without data augmentation. We report results on the next 500 samples for immediate supervision and on the next 100 samples for delayed supervision. The baseline corresponds to no data augmentation ($n = 0$).

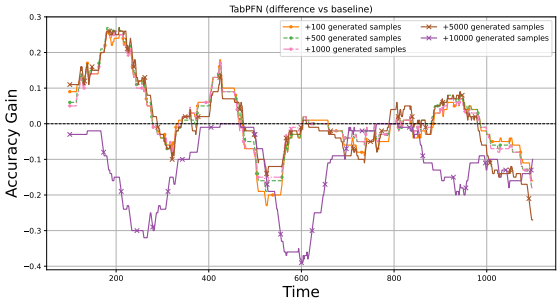
Model	n					
	0	100	500	1,000	5,000	10,000
test-then-train						
ARF	80.2	80.2(+0.0)	82.5(+2.3)	**83.2**(+3.1)	82.5(+2.3)	82.1(+1.9)
IncA-DES	78.4	80.3(+1.9)	**82.8**(+4.4)	81.1(+2.7)	81.8(+3.4)	80.5(+2.1)
TabPFNv2.5	80.1	81.3(+1.2)	**86.9**(+6.8)	79.8(-0.3)	77.7(-2.4)	81.4(+1.3)
delay						
ARF	70.2	70.0(-0.2)	67.6(-2.6)	67.6(-2.6)	**82.2**(+12.0)	80.6(+10.4)
IncA-DES	71.8	74.2(+2.4)	72.4(+0.6)	76.2(+4.4)	**77.4**(+5.6)	66.0(-5.8)
TabPFNv2.5	68.3	77.2(+8.9)	74.3(+6.0)	73.3(+5.0)	**79.2**(+10.9)	65.4(-2.9)

Furthermore, the presence of direct causal descendants can mask the impact of *target drift* on accuracy. In such cases, models exploit these descendants as predictive shortcuts rather than learning the underlying causal mechanism. While this can yield strong performance under stationary conditions, it leads to performance degradation when the relationship between the target and its descendants changes (as in Figure 6e). As the presence of such causal descendants strongly separates the target variable, the HT tends to rely on them when constructing its splits. Consequently, it does not recover the underlying causal relationship that generated the data, but instead exploits more predictive spurious dependencies that are easier to capture.

In the case study in Section 6.3, models trained on samples generated by CaDrift using a graph inferred from the ELEC2 dataset improved the accuracy of subsequent data samples in the stream. This improvement is particularly evident in the early stages of learning, especially a short-term gain under delayed feedback. This suggests that CaDrift can be used for post-drift data augmentation. By generating samples that respect the



(a) IncA-DES. (b) ARF.



(c) TabPFN.

Figure 12: The difference in prequential accuracy of learners when using CaDrift for data augmentation vs. when no data augmentation is used under delayed feedback.

underlying causal structure, CaDrift provides informative training instances that accelerate adaptation to new concepts. However, this augmentation carries the risk of overfitting the model to a previous concept. We leave further exploration of post-drift data augmentation through SCM-based generation for future work.

Overall, our results highlight that the causal structure of the data plays a central role in determining both the detectability of drift and its impact on stream learners. In particular, confounding and causal descendants can obscure or amplify the observable effects of drift, influencing both statistical measures and predictive behavior. These findings reinforce the importance of incorporating causal reasoning into drift analysis, especially in streaming settings where distributional changes are frequent and heterogeneous.

7 Conclusion

In this work, we introduced a causal taxonomy of concept drift, moving beyond purely probabilistic characterizations of distribution shift. By grounding our framework in SCMs, we categorize drift events according to their causal origins, including exogenous, endogenous, target, confounder, and structural drift.

To operationalize this perspective, we proposed CaDrift, an SCM-based data stream generator that simulates controlled, mechanism-level drift events while incorporating realistic temporal dynamics, such as autoregressive noise and seasonality.

Our empirical analysis shows that drift types with different causal origins induce distinct patterns of change in marginal and conditional distributions, leading to qualitatively different impacts on predictive performance. For instance, target drift can degrade accuracy without altering covariate distributions, whereas confounder drift changes observed associations while preserving underlying causal effects. Distinguishing between these causal signatures provides a rigorous foundation for assessing model reliability beyond simple performance metrics. This insight enables researchers to move beyond "black-box" drift detection and develop more interpretable diagnostic tools that explain why a model is failing, a critical requirement for deploying machine learning systems in evolving domains.

By integrating causal discovery algorithms to synthesize real-world data streams, CaDrift bridges the gap between causal theory and practical evaluation in non-stationary environments, even under limited or delayed feedback. This work provides a foundation for the development of causally-aware approaches that can more effectively reason about and adapt to the underlying mechanisms of change in data streams.

References

Supriya Agrahari and Anil Kumar Singh. 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. ISSN 1319-1578.

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.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2020.

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

Eduardo V.L. Barboza, Paulo R. Lisboa de Almeida, Alceu de Souza Britto, Robert Sabourin, and Rafael M.O. Cruz. Inca-des: An incremental and adaptive dynamic ensemble selection approach using online k-d tree neighborhood search for data streams with concept drift. *Information Fusion*, 123:103272, 2025. ISSN 1566-2535.

Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Rosemary Ke, Sébastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to disentangle causal mechanisms, 2019.

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

Albert Bifet, Geoffrey Holmes, Bernhard Pfahringer, Richard Kirkby, and Ricard Gavaldà. New ensemble methods for evolving data streams. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 139–148, 06 2009.

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Navoneel Chakrabarty and Sanket Biswas. A statistical approach to adult census income level prediction. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 207–212, 2018.

Tianyu Chen, Kevin Bello, Francesco Locatello, Bryon Aragam, and Pradeep Ravikumar. Identifying general mechanism shifts in linear causal representations. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 42405–42429. Curran Associates, Inc., 2024.

Pedro Domingos and Geoff Hulten. Mining high-speed data streams. *Association for Computing Machinery*, pp. 71–80, 2000.

Cian Eastwood, Alexander Robey, Shashank Singh, Julius von Kügelgen, Hamed Hassani, George J. Pappas, and Bernhard Schölkopf. Probable domain generalization via quantile risk minimization. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 17340–17358. Curran Associates, Inc., 2022.

B. Keith English and Aditya H. Gaur. *The Use and Abuse of Antibiotics and the Development of Antibiotic Resistance*, pp. 73–82. Springer New York, New York, NY, 2010. ISBN 978-1-4419-0981-7.

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.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37, 2014.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Front Genet*, 10:524, June 2019.

Heitor Murilo Gomes, Albert Bifet, Jesse Read, Jean Paul Barddal, Fabrício Enembreck, Bernhard Pfahringer, Geoff Holmes, and Talel Abdessalem. Adaptive random forests for evolving data stream classification. *Machine Learning*, 106:1–27, 10 2017.

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

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, Mihir Manium, Rosen Yu, Felix Jablonski, Shi Bin Hoo, Anurag Garg, Jake Robertson, Magnus Bühler, Vladyslav Moroshan, Lennart Purucker, Clara Cornu, Lilly Charlotte Wehrhahn, Alessandro Bonetto, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. Tabpfn-2.5: Advancing the state of the art in tabular foundation models, 2026.

Michael Harries, New South Wales, et al. Splice-2 comparative evaluation: Electricity pricing. *University of New South Wales, School of Computer Science and Engineering*, 1999.

Uzma Hasan, Emam Hossain, and Md Osman Gani. A survey on causal discovery methods for i.i.d. and time series data, 2024.

Ludivia Hernandez Aros, Luisa Ximena Bustamante Molano, Fernando Gutierrez-Portela, John Johver Moreno Hernandez, and Mario Samuel Rodríguez Barrero. Financial fraud detection through the application of machine learning techniques: a literature review. *Humanities and Social Sciences Communications*, 11 (1):1130, Sep 2024. ISSN 2662-9992.

Fabian Hinder, Valerie Vaquet, and Barbara Hammer. One or two things we know about concept drift—a survey on monitoring in evolving environments. part a: detecting concept drift. *Frontiers in Artificial Intelligence*, 7:1330257, 2024. ISSN 2624-8212.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, Jan 2025. ISSN 1476-4687.

Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pp. 97–106, New York, NY, USA, 2001. Association for Computing Machinery. ISBN 158113391X.

Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

David Komnick, Kathrin Lammers, Barbara Hammer, Valerie Vaquet, and Fabian Hinder. Causal explanation of concept drift – a truly actionable approach, 2025.

Joanna Komorniczak. Synthetic non-stationary data streams for recognition of the unknown, 2025.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5815–5826. PMLR, 18–24 Jul 2021.

Jeomoan Francis Kurian and Mohamed Allali. Detecting drifts in data streams using kullback-leibler (kl) divergence measure for data engineering applications. *Journal of Data, Information and Management*, 6 (3):207–216, Sep 2024. ISSN 2524-6364.

Sanghyuk Lee and Eunmi Lee. Score function design for decision making using conditional kullback-leibler divergence. In *2024 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pp. 216–221, 2024.

Chenxi Liu and Kun Kuang. Causal structure learning for latent intervened non-stationary data. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21756–21777. PMLR, 23–29 Jul 2023.

Greta M Ljung and George EP Box. On a measure of lack of fit in time series models. *Biometrika*, 65(2): 297–303, 1978.

Viktor Losing, Barbara Hammer, and Heiko Wersing. Knn classifier with self adjusting memory for heterogeneous concept drift. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 291–300, 2016.

Afonso Lourenço, João Gama, Eric P. Xing, and Goreti Marreiros. In-context learning of evolving data streams with tabular foundational models, 2025.

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.

Pengqian Lu, Jie Lu, Anjin Liu, and Guangquan Zhang. Early concept drift detection via prediction uncertainty. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(18):19124–19132, Apr. 2025.

Hmeda Musbah, Mo El-Hawary, and Hamed Aly. Identifying seasonality in time series by applying fast fourier transform. In *2019 IEEE Electrical Power and Energy Conference (EPEC)*, pp. 1–4, 2019.

Aldo M. Paim and Fabrício Enembreck. Adaptive random tree ensemble for evolving data stream classification. *Knowledge-Based Systems*, 309:112830, 2025. ISSN 0950-7051.

Judea Pearl. *Causality*. Cambridge university press, 2009.

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

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: Identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 11 2016. ISSN 1369-7412.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

S. W. Roberts. Control chart tests based on geometric moving averages. *Technometrics*, 1(3):239–250, 1959. ISSN 00401706.

Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization, 2021.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, Jonas Peters, Rick Quax, Markus Reichstein, Marten Scheffer, Bernhard Schölkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in earth system sciences. *Nature Communications*, 10(1):2553, Jun 2019. ISSN 2041-1723.

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.

Amit Sharma and Emre Kiciman. Dowhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216*, 2020.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

W. Nick Street and YongSeog Kim. A streaming ensemble algorithm (sea) for large-scale classification. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pp. 377–382, New York, NY, USA, 2001. Association for Computing Machinery. ISBN 158113391X.

Shifeng Xie, Vasilii Feofanov, Jianfeng Zhang, Themis Palpanas, and Ievgen Redko. Cauker: Classification time series foundation models can be pretrained on synthetic data. In *The Fourteenth International Conference on Learning Representations*, 2026.

Dingling Yao, Dario Rancati, Riccardo Cadei, Marco Fumero, and Francesco Locatello. Unifying causal representation learning with the invariance principle, 2025.

Kottala Sri Yogi, Dankan Gowda V, Mouna K M, L.R. Sujithra, KDV Prasad, and P Midhun. Scalability and performance evaluation of machine learning techniques in high-volume social media data analysis. In *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pp. 1–6, 2024.

Kun Zhang, Biwei Huang, Jiji Zhang, Clark Glymour, and Bernhard Schölkopf. Causal discovery from nonstationary/heterogeneous data: skeleton estimation and orientation determination. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pp. 1347–1353. AAAI Press, 2017. ISBN 9780999241103.

A Deferred Proofs from Section 3

Proposition A.1 (Propagation of Endogenous Drift)**.** Consider an SCM $\mathcal{M}$ and suppose an *endogenous drift* at node X_i, such that the structural mechanism f_i changes while all other mechanisms remain invariant. Assume in particular that the target mechanism f_y is unchanged. Then:

1. For any ancestor $a \in \text{ancestors}(X_i)$, the conditional distribution $P(y \mid a)$ may change across concepts due to the altered mediation through X_i, unless a is d-separated from X_i.
2. The structural conditional distribution $P(y \mid pa_y)$ remains invariant.

Proof. Let P and P' denote the joint distributions induced by the SCM before and after an endogenous drift at node X_i. By the Markov factorization property of SCMs (Pearl, 2009),

$$P(V) = \prod_{X_j \in V} P(X_j \mid pa_j). \tag{10}$$

By assumption, only the structural mechanism of X_i changes, so

$$P(X_i \mid pa_i) \neq P'(X_i \mid pa_i),$$

while for all $j \neq i$,

$$P(X_j \mid pa_j) = P'(X_j \mid pa_j).$$

Since, by assumption, the structural mechanism of the target node is invariant, we state that

$$P(y \mid pa_y) = P'(y \mid pa_y).$$

Now let $a \in \text{ancestors}(X_i)$. By marginalization,

$$P(y \mid a) = \sum_{X_i} P(y \mid X_i, a) P(X_i \mid a) \tag{11}$$

Although $P(y \mid X_i, a)$ remains invariant (as it conditions on X_i and downstream mechanisms are unchanged), the term $P(X_i \mid a)$ may differ across concepts due to the modified structural mechanism of X_i. Therefore, $P(y \mid a)$ may change across concepts. Since the descendants of X_i do not change, according to Equation 10, the prior distribution of the subset of descendants of X_i conditioned to their causal parents also remains unchanged. Still, the global distribution changes.

Proposition A.2 (Effects of Confounder Drift)**.** Consider an SCM where an unobserved confounder C acts as a common cause for an observable variable X and the target y, forming a backdoor path $X \leftarrow C \rightarrow y$. Suppose a drift occurs such that the marginal distribution of the confounder changes $P(C) \neq P'(C)$, while the structural mechanisms f_X and f_y remain invariant. Hence:

1. The observable conditional distribution $P(y \mid X)$ changes across concepts due to the altered spurious association.
2. The conditional causal effect, denoted by the interventional distribution $P(y \mid \text{do}(X), C = c)$ remains invariant.

Proof. Let P and P' denote the joint distributions before and after the drift. By the law of total probability, the observable conditional distribution of y given X can be found by marginalizing over the unobserved confounder C:

$$P(y \mid X) = \sum_{c} P(y \mid X, C = c) P(C = c \mid X). \tag{12}$$

By assumption, the target's structural mechanism remains invariant. Therefore, the probability of y given its direct parents (X and C) does not change:

$$P(y \mid X, C = c) = P'(y \mid X, C = c).$$

However, the posterior distribution of the confounder given the feature, $P(C \mid X)$, is dependent on the prior $P(C)$. Using Bayes' theorem:

$$P(C \mid X) = \frac{P(X \mid C)P(C)}{P(X)}. \tag{13}$$

Since the drift alters $P(C)$, the term $P(C = c \mid X)$ will differ between concepts. Consequently, substituting this back into Equation 12 shows that $P(y \mid X) \neq P'(y \mid X)$, thus proving the first claim.

For the second claim, we apply the rules of do-calculus (Pearl, 2009). Because conditioning on C completely blocks the backdoor path from X to y, the conditional interventional distribution is equivalent to the observational conditional distribution:

$$P(y \mid \text{do}(X), C = c) = P(y \mid X, C = c).$$

Since the structural mechanism mapping X and C to y is invariant across domains ($P(y \mid X, C = c) = P'(y \mid X, C = c)$), it follows that $P(y \mid \text{do}(X), C = c) = P'(y \mid \text{do}(X), C = c)$, proving the second claim.

B CaDrift Pseudocode

CaDrift's pseudocode can be found in Algorithm 1.

C Additional Details on Sampling DAGs

The datasets in Section 6.1 are sampled using the DAGs in Figure 5. In these DAGs, the mapping functions of inner nodes are linear functions. Here, we present the linear mapping equations for each variable on those DAGs. We omit the exogenous noise terms $U_i^{(t)}$ in the equations. For DAG #1, these are the mapping functions:

- $X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$
- $X_2 \sim \mathcal{U}(a_2, b_2)$
- $X_3 := w_{31}X_1 + w_{32}X_2 + w_{33}X_6$
- $X_4 := w_{41}X_3$
- $X_5 := w_{51}X_3$
- $X_6 \sim \mathcal{N}(\mu_6, \sigma_6^2)$
- $y := f_y(X_4, X_5, X_6)$

For DAG #2, we have:

- $X_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$
- $X_2 \sim \mathcal{U}(a_2, b_2)$
- $X_3 \sim \mathcal{N}(\mu_3, \sigma_3^2)$
- $X_4 := w_{41}X_1 + w_{42}X_2 + w_{43}X_6$
- $X_5 := w_{51}X_1 + w_{52}X_4$
- $X_6 := w_{61}X_3 + w_{62}X_4$
- $X_7 := w_{71}X_5 + w_{72}X_6$
- $X_8 := w_{81}X_5 + w_{82}X_6$
- $X_9 \sim \mathcal{N}(\mu_9, \sigma_9^2)$
- $X_{10} \sim \mathcal{N}(\mu_{10}, \sigma_{10}^2)$
- $y := f_y(X_7, X_8, X_9, X_{10})$

Algorithm 1 CaDrift: Time-Dependent SCM Data Stream Generation

Input: Causal graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$, structural functions $\{f_i\}_{i=1}^{d}$, number of time steps N, AR parameter ρ, EWMA parameter α, seasonal parameters $\{(A_i, T_i, \phi_i)\}$, drift schedule $\mathcal{D}$.
Output: Non-stationary data stream $\mathcal{S} = \{(x^{(t)}, y^{(t)})\}_{t=1}^{N}$
Initialize $z_i^{(0)} = 0$ and $U_i^{(0)} = 0$ for all $i \in \{1, \ldots, d\}$
$\mathcal{S} \leftarrow \emptyset$
for $t = 1$ **to** N **do**
  if $t \in \mathcal{D}$ **then**
    Apply soft intervention to SCM ▷ Induce causal drift
  end if
  for each node $X_i \in \mathbf{V}$ in topological order **do**
    Sample Gaussian noise $\epsilon_i^{(t)} \sim \mathcal{N}(0, \sigma^2)$
    $U_i^{(t)} \leftarrow \rho U_i^{(t-1)} + \epsilon_i^{(t)}$ ▷ Autoregressive noise
    $s_i^{(t)} \leftarrow A_i \sin\left(\frac{2\pi t}{T_i} + \phi_i\right)$ ▷ Seasonal component
    $X_i^{(t)} \leftarrow f_i(pa_i^{(t)}, U_i^{(t)}) + s_i^{(t)}$ ▷ Base structural assignment
    if $pa_i^{(t)} = \emptyset$ **then** ▷ If X_i is a root node
      $z_i^{(t)} \leftarrow (1-\alpha) z_i^{(t-1)} + \alpha X_i^{(t)}$ ▷ EWMA smoothing
      $X_i^{(t)} \leftarrow z_i^{(t)}$
    end if
  end for
  Extract feature vector $\mathbf{x}^{(t)}$ and target variable $y^{(t)}$ from $\mathbf{V}^{(t)}$
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{(\mathbf{x}^{(t)}, y^{(t)})\}$
end for
Return $\mathcal{S}$

And, finally, for DAG #3:

- $X_1 \sim \mathcal{N}(\mu, \sigma^2)$
- $X_2 \sim \mathcal{U}(a_2, b_2)$
- $X_3 := w_{31} X_1 + w_{32} X_2$
- $X_4 := w_{41} X_1 + w_{42} X_3 + w_{43} X_6$
- $X_5 := w_{51} X_3 + w_{52} X_2$
- $X_6 \sim \mathcal{N}(\mu, \sigma^2)$
- $X_7 := w_{71} y + w_{72} X_{10}$
- $X_8 := w_{81} X_6 + w_{82} y$
- $X_9 := w_{91} X_7 + w_{92} X_8$
- $X_{10} \sim \mathcal{N}(\mu, \sigma^2)$
- $y := f_y(X_4, X_5, X_6, X_{10})$

All of the linear function weights w_{ij} are initialized randomly by following a uniform distribution $\mathcal{U}(-1, 1)$. The parameter μ_i of the Gaussian distributions sampling exogenous variables is randomly set to a value in the range $[-1, 1]$, while the σ parameter is set to a value in the range $[0.5, 1.5]$. The hyperparameters a and b in the Uniform distributions, also sampling exogenous variables, are sampled in the ranges $[-2, 0]$ and $[0, 2]$, respectively.

Endogenous drift is simulated by randomly modifying the weights of the linear functions, while *exogenous drift* and *confounder drift* are simulated by changing the parameters of the Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$.

The target variable y is defined by a prototype-based mapping f_y, where each class is associated with a prototype vector in the space of the parent variables of y. Given an input sample x, the class is assigned according to the nearest prototype under the Euclidean distance:

$$f_y(\mathbf{x}) = \arg \min_{k \in \{1, \ldots, K\}} \|\mathbf{x} - \mathbf{c}_k\|_2.$$

To initialize the prototypes, we compute the empirical mean μ_{pa_y} and standard deviation σ_{pa_y} of the parent variables of y. Each prototype $\mathbf{c}_k$ is then sampled independently from a Gaussian distribution:

$$\mathbf{c}_k \sim \mathcal{N}\left(\mu_{pa_y},\ \beta^2\,\mathrm{diag}\big(\sigma^2_{pa_y}\big)\right),$$

where β is a scaling factor controlling the dispersion of the prototypes, which has been set to $\beta = 1$. This results in class centers that are aligned with the distribution of the causal parents while allowing variability across classes.

Target drift is simulated by moving the prototypes $\{\mathbf{c}_k\}_{k=1}^K$, thereby modifying the decision boundaries induced by f_y.

Lastly, *structural drift* is simulated by changing the parents of a node, which might include either removing or adding nodes to the mapping function f_i. When adding a new node as a parent, its weight on the linear function is also randomly initialized through a uniform distribution.

C.1 Drift events

In this section, we describe the drift events in each DAG, divided by drift type, for the experiments performed in Section 6.1.

DAG #1. First we start with DAG #1. The exogenous nodes are X_1 and X_2, and *exogenous drift* events are simulated by randomly changing the parameter μ of the Gaussian distributions. Since the node X_2 is the only exogenous node that is not a confounder, *exogenous drift* events in DAG #1 are simulated by modifying the Gaussian distribution sampling X_2.

For the *endogenous drift* events, the order of drifted nodes is as follows:

$$X_3 \to X_3 \to X_4 \to X_5 \to X_3, X_4 \to X_4, X_5 \to X_5 \to X_3 \to X_4$$

Next, we have two confounder nodes in DAG #1: X_1 (observable) and X_6 (latent). The order of nodes modified for *confounder drift* is:

$$X_1 \to X_6 \to X_1 \to X_6 \to X_1 \to X_1, X_6 \to X_1, X_6 \to X_6 \to X_1$$

Structural drift is simulated by changing the edges of the DAGs. In the linear function, this is done by adding a new node with a random weight. We expose the events by writing the linear function of the nodes that have drifted:

1. $X_5 := w_{51}X_3 + w_{52}X_2$
2. $X_5 := w_{51}X_3 + w_{52}X_1$
3. $X_4 := w_{41}X_3 + w_{42}X_2$
4. $X_3 := 0X_1 + w_{32}X_2 + w_{33}X_6$
5. $X_5 := w_{51}X_3 + w_{52}X_6$
6. $X_4 := w_{41}X_3 + w_{42}X_6$
7. $y := f_y(X_4, X_5, X_6, X_3)$
8. $y := f_y(X_4, X_5, X_6, X_2)$
9. $X_4 := w_{41}X_3 + w_{42}X_2$ &
   $X_3 := w_{31}X_1 + w_{32}X_2 + \tilde{w}_{33}X_6$

In the fourth *structural drift* event, by multiplying X_1 by zero we mean that the edge $X_1 \to X_3$ has been removed. *Target drift* is simulated by simply randomly shifting the centroids on all of the DAGs.

DAG #2. Next, we describe drift events in DAG #2. The exogenous (non-confounder) nodes in this DAG are X_1, X_2, and X_3. The *exogenous drift* events are:

$$X_1 \to X_2 \to X_3 \to X_1 \to X_2 \to X_3 \to X_1 \to X_2 \to X_3$$

As for the *endogenous drift* events, they are:

$$X_4 \to X_6 \to X_8 \to X_7 \to X_4 \to X_5 \to X_6 \to X_8 \to X_7$$

The confounder nodes in DAG #2 are X_9 and X_{10}, both observable. The nodes modified in each *confounder drift* event are:

$$X_9 \to X_{10} \to X_9 \to X_{10} \to X_9 \to X_{10} \to X_9 \to X_{10} \to X_9$$

Lastly for DAG #2, we write the equations that changed in each *structural drift* event:

1. $X_8 := w_{81}X_5 + w_{82}X_6 + w_{83}X_4$
2. $X_8 := w_{81}X_5 + w_{82}X_6 + w_{83}X_1$
3. $X_6 := w_{61}X_3 + w_{62}X_4 + w_{63}X_1$
4. $y := f_y(X_7, X_8, X_9, X_{10}, X_6)$
5. $X_7 := w_{71}X_5 + w_{72}X_6 + w_{73}X_8$
6. $X_8 := w_{81}X_5 + w_{82}X_6 + w_{83}X_4$
7. $X_8 := w_{81}X_5 + w_{82}X_6 + w_{83}X_3$
8. $X_5 := w_{51}X_1 + w_{52}X_4 + w_{53}X_2$
9. $y := f_y(X_7, X_8, X_9, X_{10}, X_1)$

DAG #3. Finally, we describe the drift events for DAG #3, in Figure 5c. The exogenous non-confounder nodes are X_1 and X_2. The order of nodes change to simulate *exogenous drift* events on DAG #3 is as follows:

$$X_1 \to X_2 \to X_1 \to X_2 \to X_1 \to X_2 \to X_1, X_2 \to X_1 \to X_1, X_2$$

The *endogenous drift* events on DAG #3 are as follows:

$$X_3 \to X_5 \to X_4 \to X_8 \to X_3 \to X_4 \to X_5 \to X_7 \to X_9$$

The confounder nodes in DAG #3 are X_6 as an observable confounder, and X_{10} as a latent confounder. Drift events are as follows:

$$X_6 \to X_{10} \to X_6 \to X_{10} \to X_6 \to X_{10} \to X_6, X_{10} \to X_6 \to X_6, X_{10}$$

Finally, completing the drift events for all of the DAGs, we write the mapping equations that map the nodes that were modified in each drift event:

1. $X_7 := w_{71}y + w_{72}X_{10} + w_{73}X_5$
2. $X_5 := w_{51}X_3 + w_{52}X_2 + w_{53}X_1$
3. $X_8 := w_{81}X_6 + w_{82}y + w_{83}X_4$
4. $X_7 := 0y + w_{72}X_{10}$
5. $X_9 := 0X_7 + w_{92}X_8$
6. $y := f_y(X_4, X_5, X_6, X_{10}, X_3)$
7. $y := f_y(X_4, X_5, X_6, X_{10}, X_1)$
8. $X_7 := w_{71}y + w_{72}X_{10} + w_{73}X_6$
9. $X_9 := w_{91}X_7 + w_{92}X_8 + w_{93}X_{10}$

On all of the DAGs, there are 9 drift events, one every 2,000 samples, totaling 20,000 data samples and 10 concepts per dataset.

D Generator hyperparameters

Table 4 reports the hyperparameter configuration of CaDrift used to generate the datasets for the experiments in Section 6.1. All non-stationarity parameters were set to zero, as these experiments aim to isolate the effects of causal drift events on the data distribution and predictive performance.

We employ a number of prototypes equal to ten times the number of classes to induce a non-linear mapping between the parents of y and the class labels. This choice increases the expressiveness of the data-generating process and allows different concepts to occupy distinct regions of the feature space.

Table 4: CaDrift hyperparameters to generate data samples for the experiments in Section 6.1.

Hyperparameter	Value
# classes	5
# prototypes	50
ρ	0
α	0
A_i	0
T_i	0
ϕ	0

Table 5 reports the hyperparameters of the neural network utilized to approximate cause–effect relationships on the dataset synthesized from the ELEC2 dataset.

Table 5: Neural network hyperparameters to map cause–effect relationships on the ELEC2 dataset.

Hyperparameter	Value
Learning Rate	0.001
Hidden layers	1
# neurons hidden layer	10
Optimizer	*adam*
max_iter	100
hidden layer $activation\_function$	ReLU

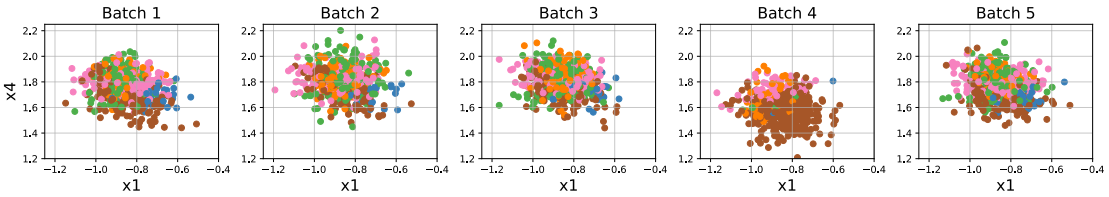
E Class distributions and MMD across causal drift events

In Figure 13, we show the class distributions across batches for datasets generated by the SCM framework with induced causal drift events, in which each color represents a different class. These are sampled from DAG #1, and the drift events are the same as described in Appendix C.1. We plot the features X_1 and X_4, two direct causes of the target y. Each color in the plots represents a class.

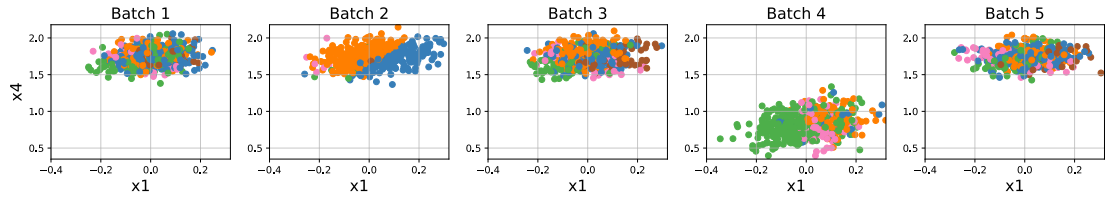
Under *exogenous drift* affecting the node X_2 (Figure 13a), we can notice the region in which X_4 is being sampled, moving, as the node X_2, which is being drifted, is one of its ancestors. The class distributions do not change in this type of drift.

In contrast, *endogenous drift*, in Figure 13b, leads to clear changes in the class distribution between concepts, reflecting modifications in the underlying mechanisms that map features to the target. As a result, decision rules learned under previous concepts generally become invalid and must be adapted.

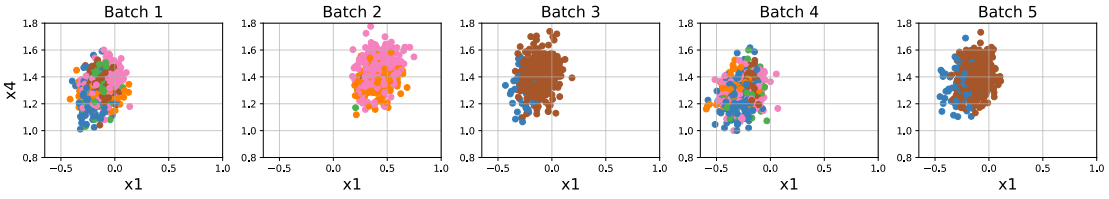
To induce *confounder drift*, we observe both the emergence of new regions in the feature space, as observed on *exogenous drift*, as well as changes in the class distribution across concepts, as observed in Figure 13c. The impact on the class distribution becomes more pronounced under *target drift* (Figure 13d), where changes in the target-generating mechanism directly alter the class boundaries, making differences between concepts apparent. To simulate these, we randomly reposition the prototypes that map the causal parents of y.



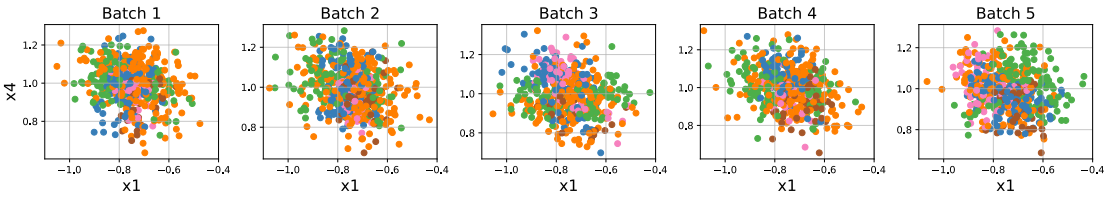
(a) Exogenous drift.



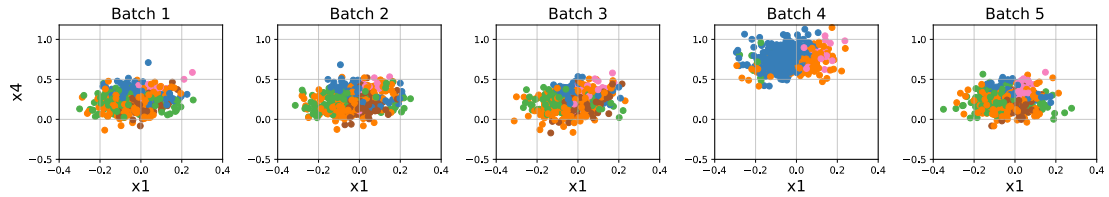
(b) Endogenous drift.



(c) Confounder drift.



(d) Target drift.



(e) Structural drift.

Figure 13: Class distribution across concepts under causal concept drift events. Each color represents a different class.

Finally, under *structural drift* in Figure 13e, it is observed that class distribution also changes. The impact of *structural drift* is closely related to the size of the causal graph. These plots use a DAG with six nodes, excluding the target node; consequently, even small structural modifications can produce large effects throughout the causal chain, leading to significant distributional changes.

F Impact of incremental endogenous drift on specific nodes

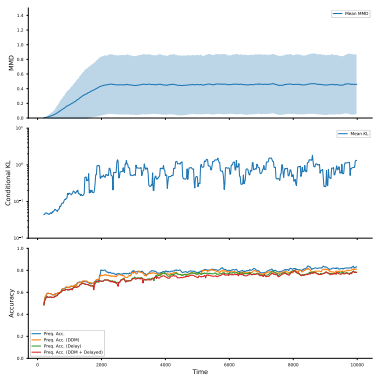
In Figure 14, we show the plots when we induce incremental updates on endogenous nodes of DAG #2 over time. The results are averaged over 10 datasets, with randomly initialized weights for the linear mappers.

We observe that the observable magnitude of drift depends not only on the structural location of the intervened node but also on the distributional regime of its ancestors. In particular, upstream variables with small means and small linear coefficients tend to attenuate perturbations as they propagate through successive structural equations. As a consequence, incremental changes introduced at early nodes in the graph may result in mild observable divergence at the level of $P(\mathbf{x})$ and $P(y \mid \mathbf{x})$.

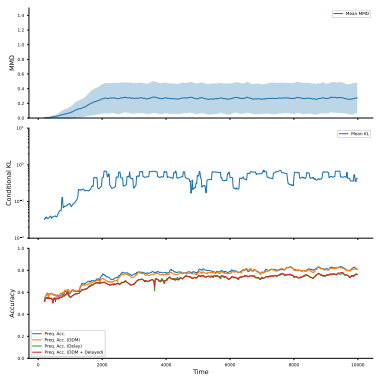
Interventions on all nodes produce measurable marginal MMD and conditional KL divergence. However, the magnitude and qualitative impact on predictive performance vary with the causal position of the drifted node.

Interestingly, long-term incremental drift on nodes leads to increases in predictive accuracy, occasionally approaching near-perfect classification when intervening on deeper nodes. This behavior arises when incremental parameter updates push the target's parents into regions of the feature space dominated by a single-class prototype, thereby reducing class overlap. Because the drift is applied smoothly at each time step, the classifier adapts to a progressively simplified decision surface before more substantial overlap is reintroduced. Thus, one should be careful when applying interventions to structural mechanisms in an SCM.

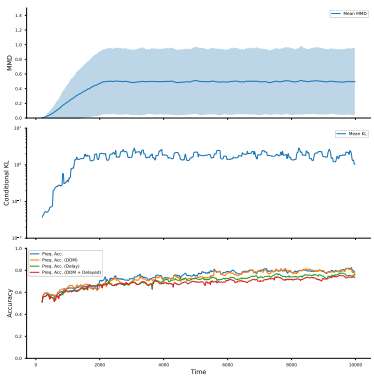
These results highlight that the effect of incremental endogenous drift is mediated by both causal distance to the target and by the interaction between structural coefficients and class-prototype geometry. Consequently, drift impact cannot be characterized solely by topological position in the graph. It also depends on how structural perturbations reshape the effective class-conditional distributions.



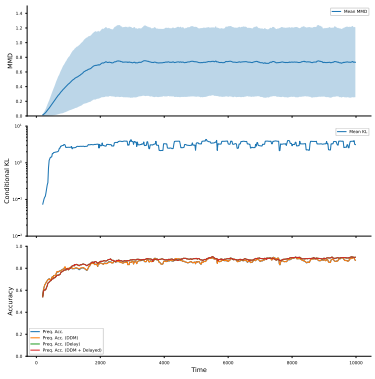
(a) Incremental Endogenous drift – Node X_4.



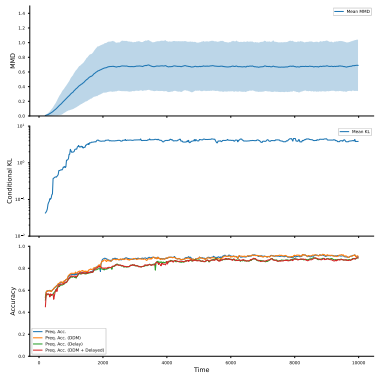
(b) Incremental Endogenous drift – Node X_5.



(c) Incremental Endogenous drift – Node X_6.



(d) Incremental Endogenous drift – Node X_7.



(e) Incremental Endogenous drift – Node X_8.

Figure 14: MMD, KL divergence, and prequential accuracy over time across concepts of incremental causal drift events on specific nodes (DAG #2).

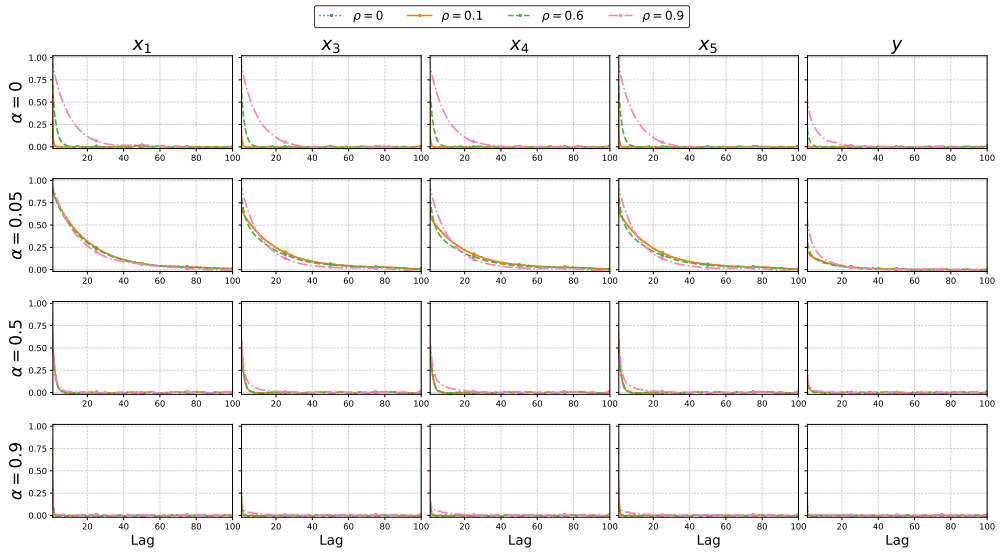


Figure 15: ACF plot on features with seasonality in DAG #1 without seasonality.

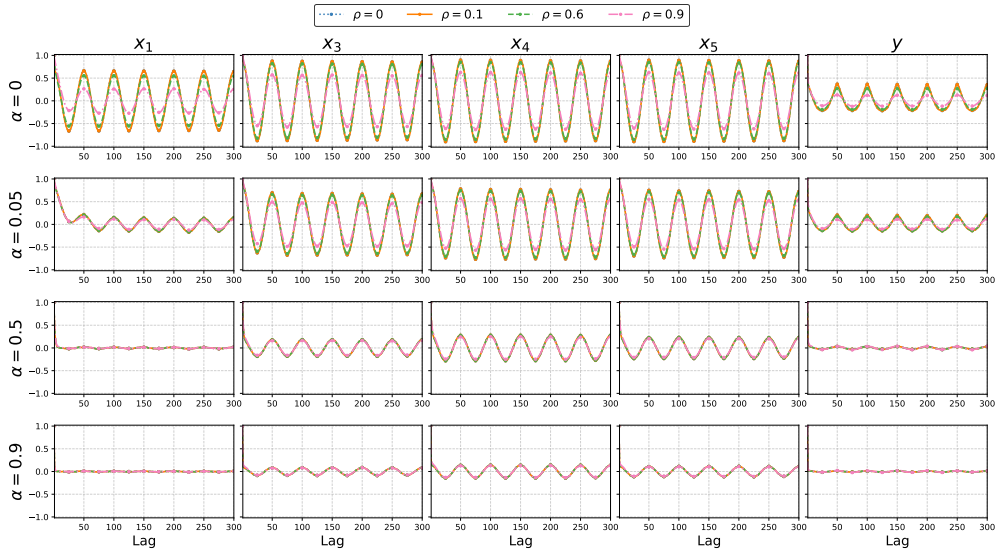


Figure 16: ACF plot on features with seasonality in DAG #1 with seasonality.

G ACF Plots for datasets generated by DAGs #1 and #3

The ACF plots for DAG #1 are shown in Figures 15 (without seasonality) and 16 (with seasonality). For DAG #3, the ACF plots are in Figures 17 (without seasonality) and 18 (with seasonality). The observations are the same as in the main text.

H Ljung-box test

We report the Ljung–Box (LB) test results for data generated by DAG #2 in Table 6. When no component inducing temporal dependence is active, we fail to reject the null hypothesis of no autocorrelation across all features. In contrast, when AR, EWMA, or seasonal components are included, the test consistently rejects the null hypothesis ($p < 0.001$), indicating the presence of temporal dependence in the generated samples.

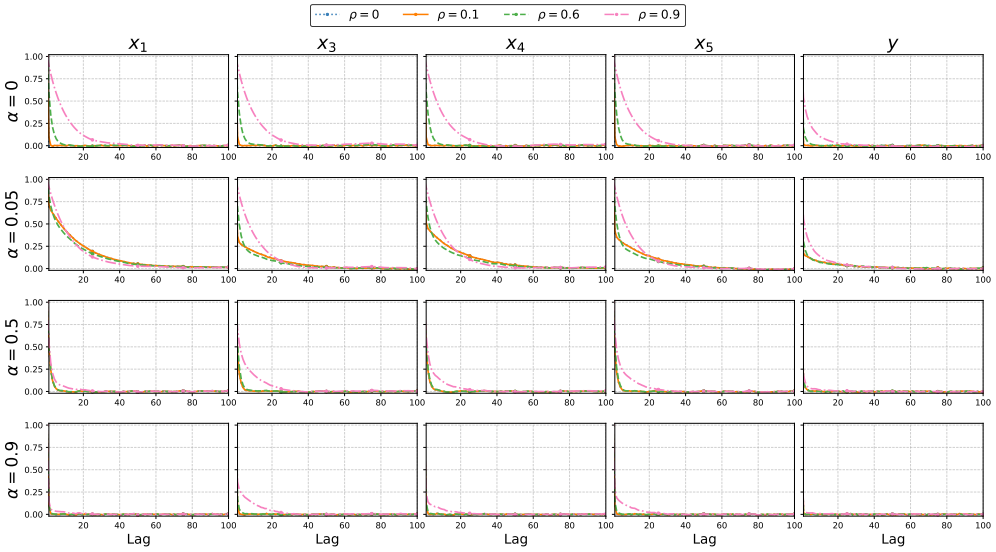


Figure 17: ACF plot on features with seasonality in DAG #3 without seasonality.

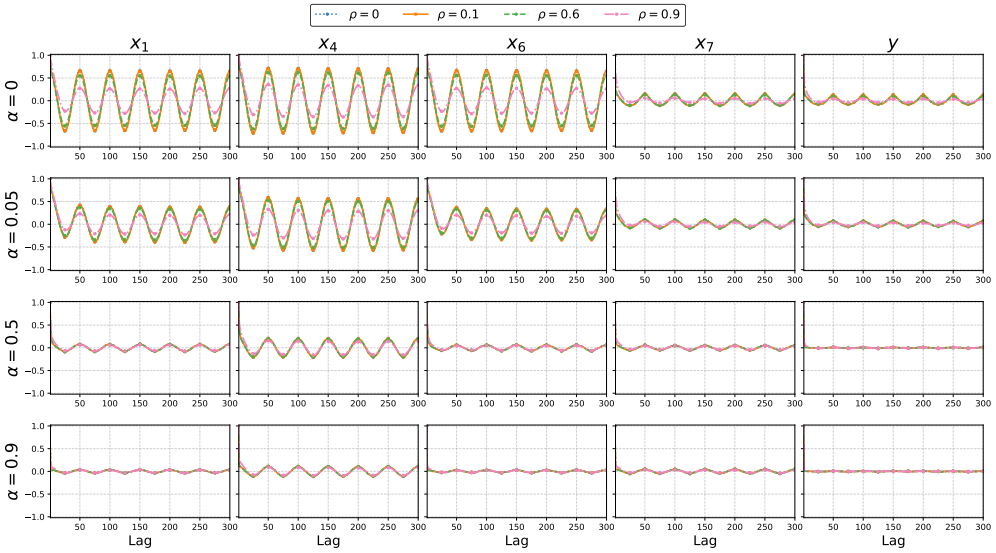


Figure 18: ACF plot on features with seasonality in DAG #3 with seasonality.

Because the temporal components are introduced through the same generation mechanisms across all DAG configurations, the qualitative effect of inducing serial correlation is consistent for DAGs #1 and #3. Therefore, for brevity, we report the LB analysis only for DAG #2.

Table 6: Ljung–Box test (lag $h = 10$) p-values for synthetic data generated by CaDrift using DAG #2 under different temporal dependence configurations.

	None (i.i.d.)	EWMA ($\alpha = 0.05$)	AR ($\rho = 0.6$)	EWMA+AR	Seasonality
X_1	0.83	< 0.001	< 0.001	< 0.001	< 0.001
X_2	0.54	< 0.001	< 0.001	< 0.001	< 0.001
X_3	0.56	< 0.001	< 0.001	< 0.001	< 0.001
X_4	0.45	< 0.001	< 0.001	< 0.001	< 0.001
X_5	0.56	< 0.001	< 0.001	< 0.001	< 0.001
X_6	0.83	< 0.001	< 0.001	< 0.001	< 0.001
X_7	0.79	< 0.001	< 0.001	< 0.001	< 0.001
X_8	0.85	< 0.001	< 0.001	< 0.001	< 0.001
X_9	0.15	< 0.001	< 0.001	< 0.001	< 0.001
X_{10}	0.85	< 0.001	< 0.001	< 0.001	< 0.001
y	0.63	< 0.001	< 0.001	< 0.001	< 0.001

I Additional experimental details: causal discovery on the ELEC2 dataset

The experiments in Section 6.3 follow a test-then-train protocol with immediate label availability (Gama et al., 2014), and one with a 100-sample delay in label availability. The results are averaged over 5 runs. This appendix provides additional detail regarding root mappers, ACF plots, and learners' hyperparameters.

The *day* feature is a categorical variable taking values in $[1, 7]$, where each value corresponds to a day of the week. Since each sample in the ELEC2 dataset is collected every 30 minutes, one day corresponds to 48 consecutive samples. To preserve this temporal structure in the synthetic stream, we define the root node as a deterministic periodic mapping from time:

$$X_{day}^{(t)} := \left(\left\lfloor \frac{t}{48} \right\rfloor \bmod 7 \right) + 1.$$

This mapping generates a weekly categorical cycle in which each day label remains constant for 48 samples (i.e., 24 hours), after which it advances to the next day and repeats after seven days. The *period* feature represents a day period, ranging in $[1, 48]$, and has been normalized to $[0, 1]$. To simulate this feature, we use the following equation:

$$X_{period}^{(t)} := \frac{t \bmod p}{p - 1},$$

where p is set to 48, referring to the period in which the feature changes.

These are the only two features for which we needed to explicitly define custom mappers to faithfully reproduce the temporal dynamics at the root nodes of the original dataset. Inner nodes and the target variable are mapped through learned small neural networks, fitted on the first 50% of the data, following the cause–effect relationships present in Figure 9.

Among the original features, the FFT identified a clear seasonal period only for the *period* node, with a period T_{period} of 48. Since no consistent seasonality was detected in the remaining nodes, no explicit seasonal components were added to their corresponding learned mappers. The only exception is the autoregressive noise term, for which the parameter ρ was set to 0.6 in order to induce moderate temporal dependence across consecutive samples. With this setup, we can synthesize the data samples.

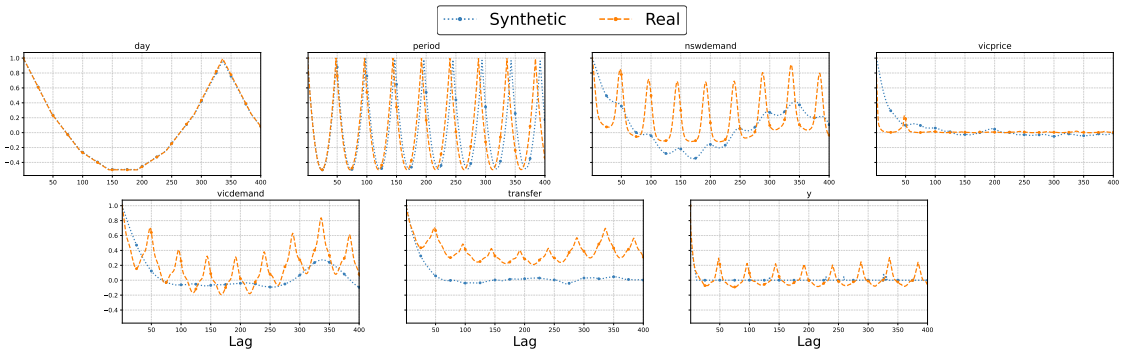


Figure 19: Synthetic vs real-world ACF plots for ELEC2. The y-axis refers to the autocorrelation, and the x-axis to the lag.

For prequential accuracy, we use a sliding window of 500 samples under immediate feedback, providing a smoother visualization of the long-term evolution of predictive performance over the stream. Under delayed feedback, the effects of augmentation are concentrated in the early stages following adaptation. Therefore, we use a smaller sliding window of 100 samples to better capture short-term performance changes that would otherwise be smoothed out by larger windows.

I.1 Additional experiments: Synthesized vs. original ELEC2

In Figure 19, we show the ACF plots for all features of the datasets generated by CaDrift and compare them to the real-world datasets. We notice that the seasonality on the features *day* and *period* is very similar. However, many features exhibit seasonality in the real-world dataset that the FFT did not detect. Nevertheless, we can still generate synthetic time-dependent samples that follow the learned cause–effect relationships from the source data.

I.2 MMD: synthesized vs. real-world ELEC2

In Figure 20, we present the evolution of the MMD over time for the real ELEC2 stream and the stream synthesized by CaDrift. We use two complementary MMD plots to assess the fidelity of the synthesized stream in terms of marginal distribution. First, the "Real vs Real" curve corresponds to the MMD computed between windows from the original ELEC2 dataset and a reference window containing the first 200 samples, capturing the natural distributional evolution of the real stream over time. Second, the "Real vs Generated" curve corresponds to the MMD computed between aligned windows from the real and synthesized streams at the same temporal positions. This design allows us to evaluate not only the similarity between real and generated samples, but also whether the synthesized stream reproduces the temporal distributional dynamics of the original data.

We observe that the synthesized stream follows the same seasonal MMD patterns present in the original ELEC2 dataset, with peaks occurring at similar temporal regions. Moreover, the divergence between aligned real and synthesized windows remains relatively stable over time, suggesting that CaDrift reproduces key temporal and distributional characteristics of the original stream, although the seasonal dynamics are not perfectly matched.

I.3 Online Learners Hyperparameters

Table 7 reports the hyperparameters of the online learners used in the experiments in Section 6.3. For TabPFN, we use a *context_window* of 10,000 samples. This implies that, when $n = 5,000$, 50% of the context window consist of synthetic data generated by CaDrift. When $n = 10,000$ the whole initial context window consist of synthetic data.

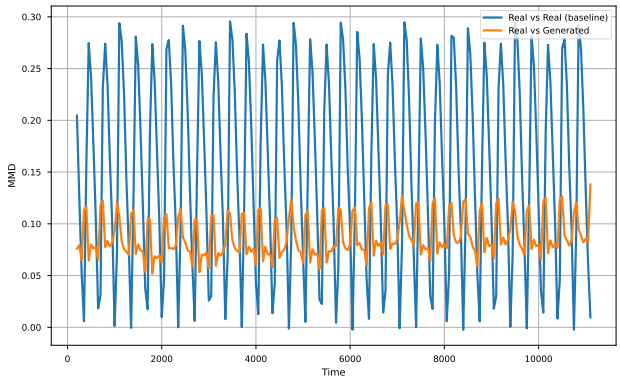


Figure 20: MMD over time: real vs. generated ELEC2.

Table 7: Learners' hyperparameters. All ensemble methods use an HT as a base classifier.

Method	Hyperparameters
ARF	$change\_detector : ADWIN$, $ensemble\_size = 100$
IncA-DES	$change\_detector : RDDM$, $pool\_size = 100$, $F = 200$, $k = 5$
TabPFN$_{v2.5}$[Stream]	$context\_window = 10,000$, $short\_term\_window = 7,500$, $long\_term\_window = 2,500$