A Data Generator to Explore the Interaction Between Anomalies and Concept Drift

Jongjun Park, Akanksha Nehete, Tammy Zeng, Fei Chiang

Computing and Software, McMaster University

Hamilton, Ontario

{parkj182,nehetea,zengt5,fchiang}@mcmaster.ca

ABSTRACT

As data changes, it is crucial that predictive models remain robust even in the presence of anomalies and concept drift. A significant challenge towards accurately identifying and differentiating between concept drift and anomalies is due to the lack of real, labeled datasets containing both types of events. Given the varied types of anomalies and concept drifts, and the varying distributions in which they occur in practice, obtaining labelled datasets is challenging. In this paper, we propose CanGene, a tool for anomaly injection, and concept drift generation into existing time-series data. CanGene allows users to specify the types, frequencies, locations, and interactions of injected anomalies and concept drifts according to selected distributions. We demonstrate CanGene's capability through a series of cases using electrocardiogram and weather datasets, illustrating CanGene's ability to synthetically replicate real world changes and anomalies in time series data.

VLDB Workshop Reference Format:
Jongjun Park, Akanksha Nehete, Tammy Zeng, Fei Chiang. A Data Generator to Explore the Interaction Between Anomalies and Concept Drift. VLDB 2024 Workshop: 13th International Workshop on Quality in Databases (QDB'24).

VLDB Workshop Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/mac-dsl/AnomalyDriftDetection.

1 INTRODUCTION

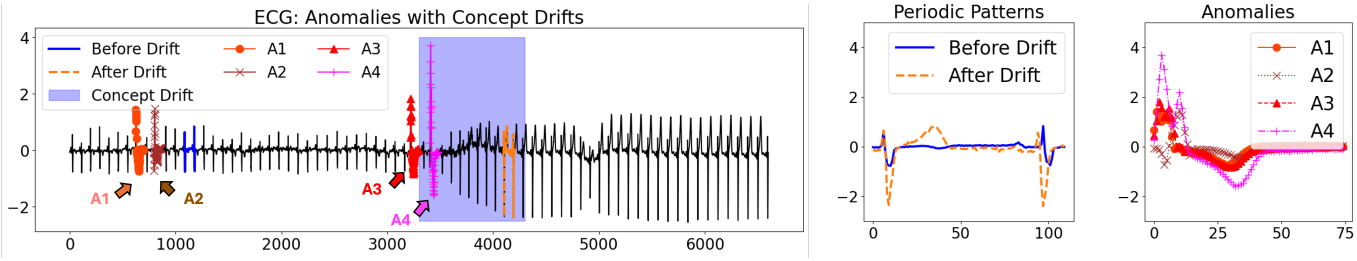
Data inevitably changes to reflect user activity and preferences, and changes in the environment. Identifying the inherent patterns to understand how data changes is a fundamental task in time series analysis and prediction. The rate at which data changes, the magnitude of change, and the time duration of the change, are characteristics used to determine whether a data change is normal or not. When an input data distribution changes from the original data distribution, this is often referred to as concept drift [9, 12, 19, 20]. Existing time series, anomaly detection techniques ignore concept drift, assuming time series concepts are stationary, and that data values follow a fixed probability distribution [6, 7, 25]. Clearly, this assumption does not hold in practice as applications and natural phenomena elicit changes in the underlying distribution. For example, temperature changes between seasons demonstrate a gradual increase from winter to spring, changes in workplace electricity usage from weekday to weekend exhibit an abrupt decrease due to a change in employee work patterns, and a company's stock price changes due to political and economic events, and investor sentiment and speculation.

Anomalies and concept drift commonly occur together in practice. For example, the temperature distribution in an aircraft engine fluctuates according to changes between take-off (climb), cruising altitude, turbulence, and landing. During any of these phases and transitions, mechanical failures (anomalies) may occur causing unexpected increases in temperature within the engine module. As another example, in fraud detection over online transactions, hackers frequently change the distribution of fraudulent transactions to gradually shift over time to avoid detection. However, fraudulent transactions may also contain anomalies, such as abnormally large/small transaction amounts, or transactions from unusual locations. The presence of concept drift poses challenges for anomaly detection in time series. While anomalies are caused by undesirable changes in the data, differentiating abnormal changes from varying normal behaviours is difficult due to differing frequencies of occurrence, varying time intervals when normal patterns occur, and identifying similarity thresholds to separate the boundary between normal vs. abnormal sequences.

Differentiating between concept drifts and anomalies is critical for accurate analysis as studies have shown that the compounding effects of error propagation in downstream data analysis tasks lead to lower detection accuracy and increased overhead due to unnecessary model updates [5, 6, 16]. Adopting anomaly detection methods for drift detection lead to mis-classification and an increased number of false positives. In contrast, existing drift detection methods assume a negligible amount of anomalies, or fail to consider them at all [3, 6, 12, 20].

Recent work by Le and Papotti study the problem of anomaly detection with change points (based on sudden, abrupt changes) from a group of distributionally similar sequential data points [17].

Proceedings of the VLDB Endowment. ISSN 2150-8097.



(a) Amplification of anomalies in the presence of drift. **(b) Varying normal baselines.** **(c) Anomaly streams.**

Figure 1: Concept drift and anomalies in real ECG data.

However, such methods assume that anomalies are short-lived and independent; and do not consider broader definitions of data drift. Concept drift and change detection methods have largely used windowing-based methods that compare overlapping or adjacent windows to detect significant changes beyond statistical (test) thresholds [3, 8]. Techniques for Seasonal Trend Decomposition are often used to differentiate among seasonal (periodic) patterns, trends over time, and residuals, but are often adapted to application-specific changes with respect to events [13], fraud [18], or patterns [2, 24]. Unfortunately, existing work has largely explored anomaly detection and concept drift detection in isolation [12, 14, 19].

Example 1.1. Figure 1a shows a real data stream of ambulatory electrocardiogram (ECG) recordings [22]. Heart rhythm problems, known as arrhythmias, occur when the electrical signals that coordinate the heart's beats cause the heart to beat too fast (tachycardia), too slow (bradycardia) or irregularly. However, normal activities such as exercise and sleep, cause the heart rate to increase and decrease, respectively. Differentiating between anomalous and normal baselines is important towards accurate diagnosis and life-saving treatment.

The data contains four collective anomalies (in red, labelled A1 - A4), showing an irregular heart rate, and a concept drift (shaded in purple), where the heart rate increases, and with similar pattern readings before and after the drift period. Figure 1b shows a zoomed-in view of a snippet of the ECG readings before and after the concept drift, occurring at approximately $x = 1100$ and $x = 4100$, respectively (Figure 1a). While the period remains the same, the concept drift has caused a change from the baseline pattern (shown in dark blue), to a pattern with larger amplitude and mean (at $x = 35,95$, shown in orange). Existing techniques misclassify these changes as anomalies, leading to an increase in false positives [6, 21, 26]. In contrast, Figure 1c shows a zoomed-in view of irregular heart rates, depicted as collective anomaly patterns A1 - A4. Given the similarity of these error distributions, occurring at irregular times, existing drift detection techniques, largely ignoring the presence of anomalies, miscategorize such instances as recurring data drifts [3, 12, 20].

One of the contributing factors to this problem is the lack of real, labelled datasets containing *both* concept drifts and anomalies. Such datasets are often inaccessible due to privacy constraints in medical or financial areas, expensive to obtain, or incomplete due to missing values or only a partial subset of data is available. We often then resort to synthetic data as an alternative. Generating synthetic data containing different types of concept drift and anomalies, with realistic distributions of co-occurrence frequency is challenging. First, we must understand the characteristics of different types of anomalies, and different types of concept drift that occur in practice, and define the corresponding user parameters. Drift can occur in several forms, each presenting distinct data generation challenges. Second, dependencies between anomalies, and between drift and anomalies introduce sequencing constraints, e.g., a drift may induce or amplify an anomaly to occur.

Anomaly and Concept Drift Types. We consider three types of anomalies:
(1) Point Anomaly: are individual data points that deviate significantly from the majority of points in the dataset, this is often characterized by a large difference in value(s) from nearby data points;
(2) Collective Anomaly: are a set of data points that *together* exhibit anomalous behaviour, while the individual values of these data points may not be considered anomalous;
(3) Periodic Anomaly: is a collective anomaly that occurs with an expected regularity (period). We characterize periodic anomalies as a sub-type of collective given its proliferation in application settings, e.g., network traffic, and patient health monitoring.

We model three types of concept drifts that commonly occur in practice:
(1) Gradual Drift: changes happen slowly over time where the source concept increasingly transitions to the target concept;
(2) Abrupt Drift: characterized by sudden shifts in data patterns often occurring within a short time period; and .
(3) Recurring Drift: changes from a source concept to a target concept are generated from a distribution that was previously observed, often with an expected period.

While there exist a wealth of anomaly labelled, real time-series data [1, 10, 15, 23], these lack labelled instances of concept drift. Conversely, real data that contain drift are often unlabelled, and synthetic generators for concept drift, such as MOA [4], do not have the inherent capabilities to inject different types of anomalies and

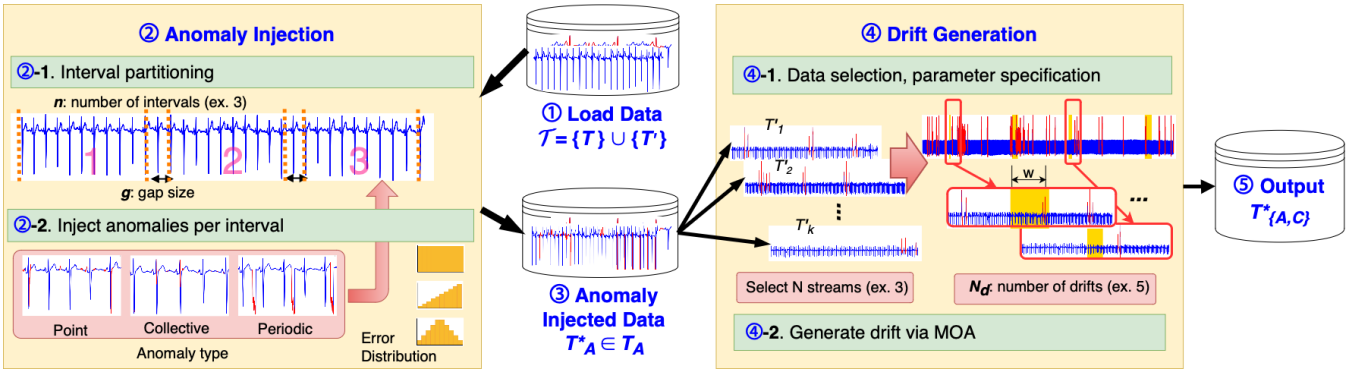


Figure 2: CanGene overview.

their unique properties. With the proliferation of machine learning models in data quality assessment and re-evaluation, these underlying models require realistic datasets containing both concept drift and anomalies to adapt to evolving data patterns to improve robustness, and towards more accurate training and validation.

To address these challenges, we introduce CanGene, a Concept drift and anomaly data Generator, a tool for injecting anomalies and concept drift into existing time series data, with the following features:

Type-based anomaly injection: users can specify up to three types of anomalies: point, collective, and periodic. Anomalies may be injected independently or drawing from a set of distributions with varying likelihood.

Type-based concept drift injection: users can specify a number of abrupt or gradual drifts that span a proportion of the dataset, with varying lengths, and relative to one or more anomalies. The latter case is particularly useful when studying interactions between drift and anomalies, and when changes in normality occur.

Declarative specification: CanGene provides parameterized specification of anomalies, and *the location of drifts relative to anomalies*. Users are able to control the interaction of anomalies and concept drift by customizing their type, location, frequency and duration. This specification is important for localized drift and/or anomaly detection when the local context differs from the global context. We provide a technical overview and architecture of CanGene, and describe system features in Section 2, followed by the demonstration scenarios in Section 3.

2 SYSTEM OVERVIEW

CanGene provide users with an integrated toolkit for anomaly injection and concept drift generation. The occurrence of anomalies and drifts, especially in close proximity in a time series, influences the accuracy of their detection. For example, the accuracy of detecting an abnormal heart rate depends on whether a person is starting to transition from walking to running, or whether they are in a resting state. CanGene provides features to allow users to further explore these interactions by not only providing users the ability to specify the type, location and quantity of anomalies and concept drift, but also *the position of generated concept drift relative to the position of anomalies*. This enables users to generate datasets where drift occurs within a window (either before or after) of an existing anomaly. We start by introducing the architecture of CanGene followed by a discussion of each of its components.

Overview. Figure 2 presents the overall architecture of CanGene, consisting of two main components: (1) anomaly injection and (2) concept drift generation. Given a set of input time series $\mathcal{T} = \{T\} \cup \{T'\}$, where $\{T\}$ represents an input set of time series to be injected with anomalies, and $\{T'\}$ denotes the set of time series from which we will use to generate concept drift. CanGene partitions a given $T^* \in \{T\}$ into n intervals (each separated by a gap size g). Users specify the quantity of desired anomalies (as a percentage of the interval size). For each interval, anomalies are injected according to a specified type, and error distribution, creating a new (anomalous) dataset $T^*_A \in \{T_A\}$.

To inject concept drift into a selected $T^*_A \in \{T_A\}$, we select k time series $\{T'_1, T'_2, \ldots T'_k\} \in \{T'\}$, and for each pair (without loss of generality) $\{T'_1, T'_2\}$, CanGene creates a drift in T^*_A that transitions from a subset of T'_1 to T'_2. We consider two types of drift: (i) abrupt, which occurs during a single time point, and (ii) gradual, which occurs over a window of w time points. The likelihood of selecting data points in w from either T'_1 or T'_2 is governed by a sigmoid function. Users specify the percentage of T^*_A that will compose a concept drift, the duration of each drift, and its location (relative to existent anomalies). We use the Massive Online Analysis (MOA) framework to augment the drift generation process, particularly for gradual drifts, where a source stream is gradually replaced by a target stream over the duration of the drift [4]. CanGene generates a new time series, $T^*_{\{A,C\}}$ containing both anomalies and concept drift.

2.1 Anomaly Injection

We define an anomaly as follows [25].

Definition 2.1. A time series *anomaly* is a sequence of one or more data points $T_{i,j}$ where its length $|T_{i,j}|$ equals $j - i + 1$, and it's

Table 1: Anomaly-injection parameters (defaults in bold).

Sym.	Description	Values
n	number of intervals	[**1**, $\|T^*\|$]
g	gap size	[**1**, $\|T^*\| - 1$]
a	% anomalies in an interval	(0.00, 1.00] (**0.01**)
d	error distribution	**Uniform**, Gaussian, Skew-normal
λ	anomaly subsequence length	(1, $\|T^*\|$]
$s \in T'$	start of periodic anomaly	[1, $\|T^*\|$]
η	additive noise factor	[0, 100] (**0.5**)

value, pattern, or behaviour deviates from the remaining patterns in time series T.

CanGene considers three types of anomalies:

- *Point anomaly*: are individual data points that deviate significantly from the majority of points in the dataset, and $|T_{i,j}| = 1$. For example, a point anomaly in online credit card transactions occurs when a single transaction has an abnormally large dollar value compared to all other transactions. Clearly, identifying such point anomalies is important towards recognizing fraudulent behaviour.
- *Collective anomaly*: are anomalies that involve a group of data points exhibiting anomalous behavior when considered together, with $|T_{i,j}| > 1$. The actual values of these points may not be anomalous, but the presence of the collective set of values indicate the anomaly.
- *Periodic anomaly*: are a consecutive set of point anomalies that repeatedly occur with some period with $|T_{i,j}| > 1$. Consider two examples: (1) in electrocardiogram (ECG) data, a heart arrhythmia is a sequence of anomalies that occurs with a regular period, at every heartbeat; and (2) in mechanical subsystems in manufacturing, components exhibit periodic errors due to deteriorating components. Collective anomalies subsume periodic anomalies, and we consider periodic as a sub-type of collective. We define periodic anomalies separately given their presence in real datasets, and to allow users to declaratively define the expected regularity.

CanGene transforms existing data points for a given time series $T^* \in \{T\}$ to anomaly points. CanGene first partitions T^* into n intervals, separated by a gap g. For each interval, we define the percentage, a of the interval that is anomalous, the type of anomalies, and the corresponding parameters including distribution d (for point and collective anomalies). We assume all intervals are of equal length, and we inject anomalies into each interval according to a specified type.

Point Anomalies. We update values in T^* to point anomalies by drawing error values from a given distribution d, where d is one of Gaussian, Uniform (default), or Skew-normal. For the Gaussian distribution, for a given mean μ and standard deviation σ, an anomaly value will be generated to lie within a μ percentage above or below its original value. For example, for $\mu = 0.5$, anomaly values will be more likely 50% higher than its original value, and for $\sigma = 0.2$, approximately 68% of the anomaly values will lie within a 30% to 80% difference of their original values. For the Uniform distribution, users can specify lower and upper bounds, $[l, u]$, which restrict the allowed changes of an existing value $v \in T^*$ to compute $v' \in T^*_A$, such that $\frac{(v'-v)}{v} \in [l, u]$.

The Skew-normal distribution is a continuous probability distribution that generalises the normal distribution to allow for non-zero skewness. Anomalies often occur in practice with long-tailed distributions that are not achievable with the aforementioned distributions. The Skew-normal distribution with a skew parameter α and an upper bound of u. Specifically, our Skew distribution re-scales the random number to $[0, u]$, so when $u > 0$ and $\alpha > 0$, the probability density function (pdf) is left-skewed, and conversely, when $u < 0$ and $\alpha > 0$, the pdf is right-skewed. In both cases, with $\alpha > 0$, the pdfs are skewed near 0. To inject an anomaly, we select r numbers from the given distribution d, e.g., for Uniform distribution with bounds [0,1], we select r numbers from this range. We randomly select a value $v_r \in r$, and generate the anomaly value as $v_t \cdot (1 + v_r)$ for a $v_t \in T^*$.

Periodic Anomalies. CanGene creates periodic anomalies by first selecting a sequence of points starting at a given point $s \in T^*$ (within an interval), for a duration of length λ. We then add Gaussian white noise with mean $\mu = 0$, and standard deviation $\sigma = \eta$ to generate an anomaly sequence $T^*_{s,s+\lambda}$. This sequence $T^*_{s,s+\lambda}$ is randomly placed within the interval $a \cdot |T^*|/\lambda$ times, such that the total duration of all periodic anomalies does not exceed a.

Collective Anomalies. For an interval, CanGene randomly selects a starting point s_i for each collective anomaly, and selects all subsequent points up to length λ to modify. Similar to point anomalies, error values follow one of three distributions (Gaussian, Uniform or Skew-normal), where lower and upper bounds can be specified to restrict the range of error values for the Uniform and Skew-normal distributions. For each point $v_{t_i} \in T^*_{s_i,s_i+\lambda}$ in the collective anomaly, we generate error values by selecting r numbers from the given distribution d. We randomly select a value $v_r \in r$, and generate the anomaly value as $v_{t_i} * v_r$. The total length of all collective anomaly durations does not exceed a.

Summary. Table 1 summarizes the list of parameters used during anomaly injection. CanGene empowers users to control the type, location, and quantity of injected anomalies at a more fine-grained, interval-based manner. Data points are randomly selected for error injection according to a given distribution d, such that no more than a percentage of the interval is anomalous. CanGene allows users to generate realistic error scenarios containing different anomaly types, following different distributions at different points in time. For example, Figure 9 shows an anomaly injection case where point anomalies are injected for 1% of the $|T^*|$ following a Uniform distribution, then collective anomalies following a Gaussian distribution with ($\mu = 1, \sigma = 0.1$) for 1% of the first interval, and lastly, periodic anomalies for 5% of the second interval with a $\eta = 0.5$ noise factor. Anomalies often do not occur in isolation, but in conjunction with concept drift reflecting seasonality trends, changes in the environment, and user behaviour. We describe how CanGene generates concept drift relative to anomalies next.

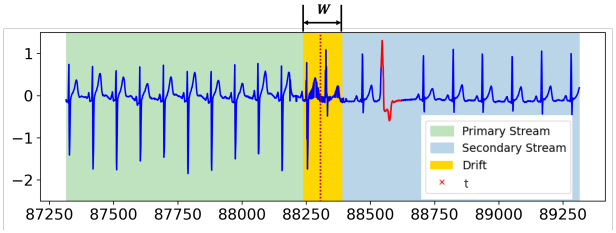


Figure 3: Gradual Drift, $w = 120$ and $t = 88300$ (ECG data).

2.2 Drift Generation

Given a time series T_A^* with anomalies, CanGene injects concept drift into T_A^* by combining two given data streams, without loss of generality, $\{T_1', T_2'\}$, a primary and a secondary stream, representing the source and target concepts, respectively. We define concept drift as follows [20].

Definition 2.2. A concept drift at time t is defined as the change of the joint probability of feature vector X and label y at time t, denoted as $\exists t : P_t(X, y) \neq P_{t+1}(X, y)$, where the joint probability can be decomposed as $P_t(X, y) = P_t(X) \cdot P_t(y|X)$.

Informally, a concept drift is defined as the transition from a source concept (primary data stream) to a target concept (secondary data stream). In CanGene, we define parameters to specify the centre of the drift at a time point t, and the drift spans width w, as shown in Figure 3. To study the interaction of anomalies and concept drift, CanGene enables users to specify the location of a concept drift relative to the position (before or after) of existent anomalies. CanGene considers three types of concept drift:

- *Abrupt drift*: An abrupt (or sudden) drift is defined as a concept drift that occurs suddenly at an exact timestamp, without a transitional period. In CanGene, an abrupt drift occurs when the transition from the source concept to the target occurs suddenly at a particular point in time ($w = 1$).
- *Gradual drift*: A gradual drift is the transformation from a starting concept to an ending concept over a certain period of time. During this period, intermediate concepts may appear, which can be selected either the starting or ending concept depending on their proximity. A gradual drift occurs when a new concept gradually replaces an old one over an extended period of time ($w > 1$).
- *Recurring drift*: A recurring drift is a phenomenon where statistical or sequential patterns change over time, and then re-appear after some duration. This occurs when the properties of the target concept change over time, but revert to a previous state (e.g., network traffic before and after the start of a new school year).

Methodology. Each concept drift is generated by randomly selecting a pair of distinct streams from $\{T'\}$. Let C denote the percentage of T_A^* that will be used to generate concept drift, n_C the number of concept drifts in T_A^*, and C_A denote the percentage of the number of drifts occurring before an anomaly. We randomly select candidate anomalies to generate drift before their occurrence in T_A^*. We generate drift before an anomaly by selecting a position t to inject drift that is within $\frac{w}{2}$ time units before an anomaly occurs. The sum of all concept drift durations over the time series length equals $C = \frac{\sum_{i=1}^{n_C} w_i}{|T_A^*|}$, e.g., the sum of the (yellow) durations as shown in Figure 4.

Table 2: Concept drift generation parameters (defaults in bold).

Sym.	Description	Values
n_C	number of concept drifts	any ≥ 1, **3**
C	percentage of drift in T_A^*	0.05, 0.2, **0.35**, 0.5, 0.65, 0.8
C_A	percentage of drift before anomalies	0, 0.25, **0.5**, 0.75, 1

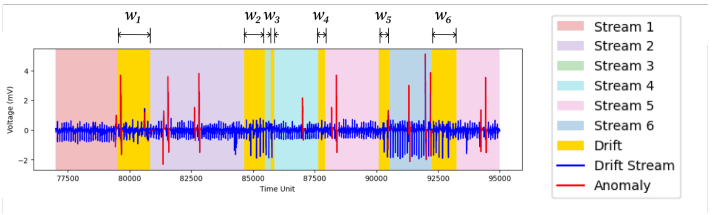


Figure 4: Generated drifts for $n_C = 6$ (ECG data).

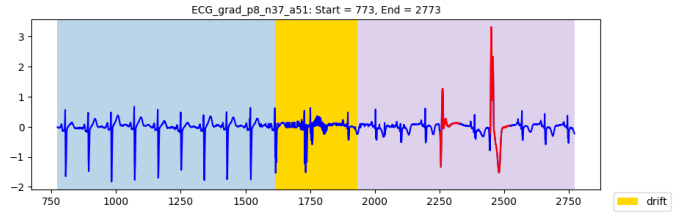


Figure 5: Sample drift before anomalies (ECG data).

To generate the transition from the source concept to the secondary concept, CanGene uses the Massive Online Analysis (MOA) platform, a software environment for implementing and testing evolving data streams [4]. Given two data streams, $\{T_1', T_2'\}$, a concept drift C is generated by joining $\{T_1', T_2'\}$ as $C = T_1' \bigoplus_t^w T_2'$. Based on the sigmoid function, we select C at time t with width w as one of T_1' or T_2' with probability according to Equation 1.

$$P[C(t_i) = T_1'(t_i)] = e^{-4(t_i - t)/w} / (1 + e^{-4(t_i - t)/w})$$
$$P[C(t_i) = T_2'(t_i)] = 1 / (1 + e^{-4(t_i - t)/w}) \quad (1)$$

Figure 5 shows a sample concept drift (denoted in yellow), occurring before the anomalies (denoted in red). The concept drift transitions gradually from the source concept (denoted in blue) towards an increasingly larger proportion of the target concept (shown in purple). Table 2 summarizes the parameters used for concept drift generation. The datasets and CanGene code are publicly available on Github[1].

[1]https://github.com/mac-dsl/AnomalyDriftDetection

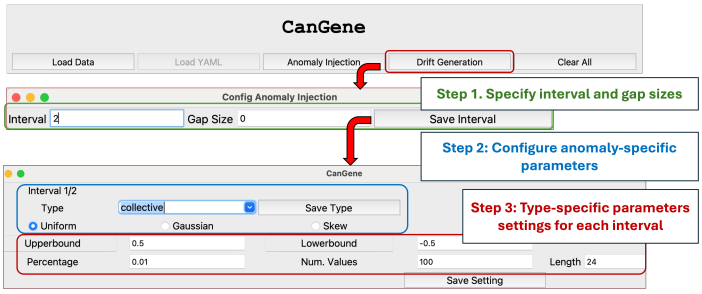


Figure 6: UI parameter configuration for anomaly injection.

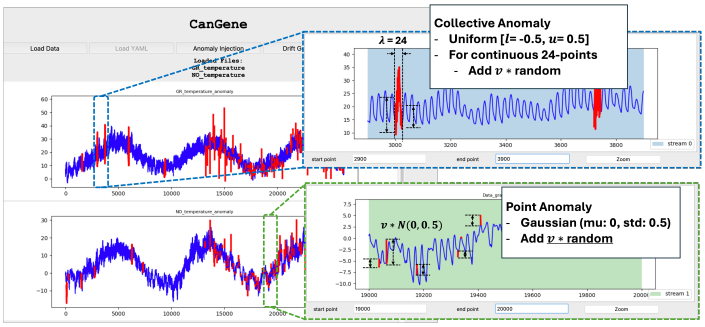


Figure 7: Anomaly injection.

3 DEMONSTRATION

We present three scenarios that will allow users to interact with CanGene to experience its anomaly injection and data generation features. We also demonstrate the utility of our datasets by evaluating three existing anomaly detection baselines. Our results show that the baseline performance decreases as the presence of concept drift increases in the data.

Datasets. Our demonstration will feature the use of two real datasets: (1) ECG dataset: describes patient electrocardiograms obtained from the MIT-BIH Arrhythmia database [22]. Anomalies represent ventricular premature contractions.
(2) Weather dataset: hourly, geographically aggregated temperature recording of European countries obtained from the NASA MERRA-2 [11]. The dataset records timestamp, average temperature, and radiation levels from 1960 to 2020. However, for our demo, we will focus on each country's temperature reading from 2017 to 2020. To better exemplify the transition (drift) between seasons, we selected countries with different climates: Greece (GR, warm), Norway (NO, cool), and Germany (DE, moderate). CanGene reads input streams in either `*.csv` or `*.arff` formats.

We first introduce users on how to setup and configure CanGene parameters. We then present the first scenario to add point anomalies over the entire time series T^* to obtain T^*_A. We will then divide T^*_A into two intervals, injecting collective anomalies in the first interval, followed by periodic anomalies in the second interval. We aim to generate anomalies that replicate severe weather events such as hurricanes, or snow storms occurring at a single time point

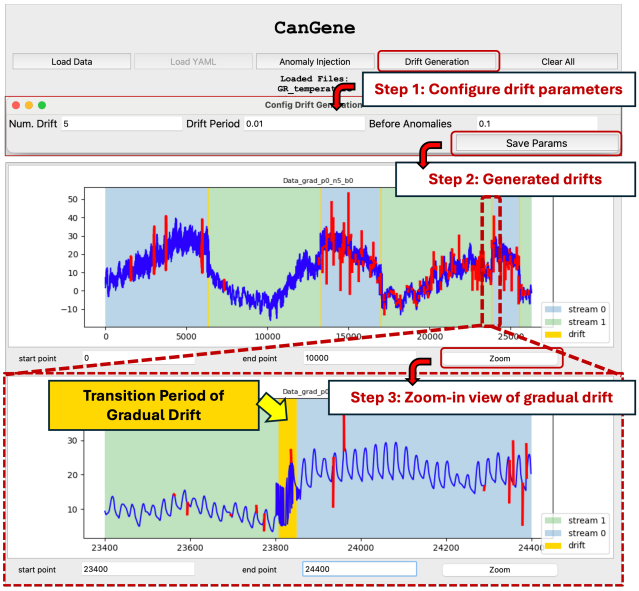


Figure 8: UI parameter configuration for drift generation.

(point anomalies), and weather events such as rainstorms, and heat waves, which span for longer durations (collective anomalies), and re-occur each year (periodic anomalies). In the second scenario, we demonstrate how drifts can be generated between these climates to simulate changes between the seasons, from cool to moderate climates (Fall season), and from cool to warm temperatures (Summer season). Lastly, we evaluate three recent baseline anomaly detection algorithms [6, 7, 21], showing their sensitivity and decreasing performance as the presence of concept drift increases in the data.

User Interface. Figure 6 shows the CanGene user interface. Users first load their input data in either `*.csv` or `*.arff` format, and then specify anomaly injection parameters such as the interval and gap sizes (Step 1). For each interval, users specify the anomaly type,

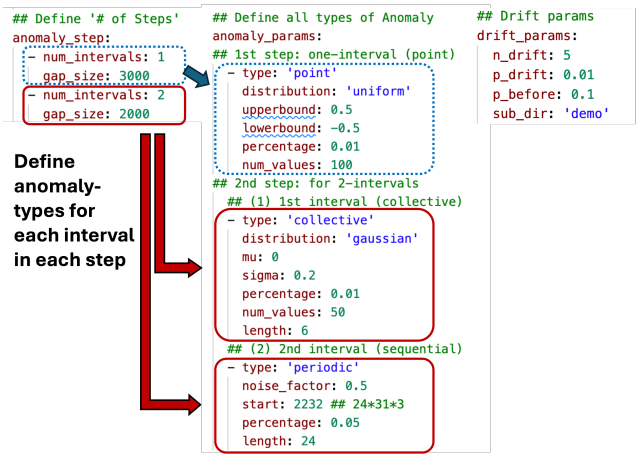
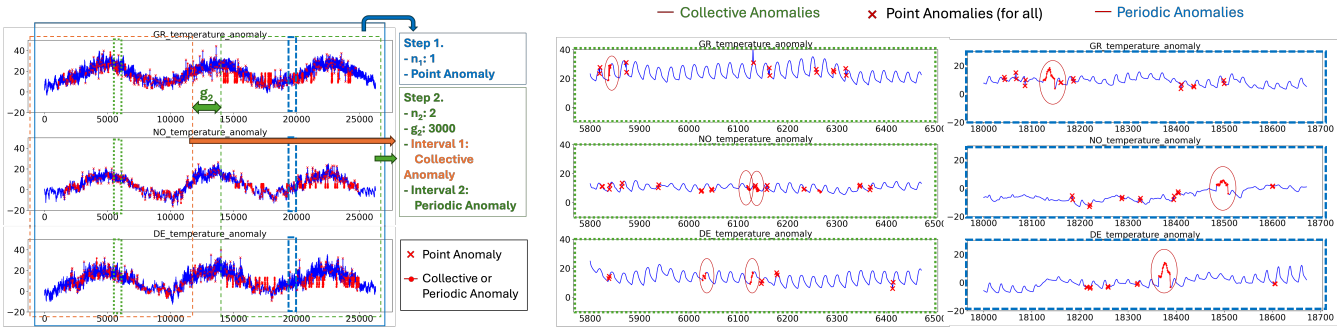


Figure 9: Example configuration file (`config.yaml`).



(a) Point, collective, periodic anomalies over 3-years.

(b) 1^{st}-half (point and collective) (left), 2^{nd}-half (point and periodic) (right)

Figure 10: Injected anomalies reflecting anomalous weather events.

the error distribution, the percentage of errors to inject, and the lower and upper bounds (Step 2 and Step 3). Figure 7 shows sample generated collective anomalies with uniform distribution (top), and point anomalies with Gaussian distribution (bottom). These (labelled) anomaly injected time-series are then used to generated concept drifts. Figure 8 shows the parameter configuration for drift generation where users specify the type of drift, quantity, and the percentage of the drift that occurs before an anomaly (Step 1). The generated drifts are shown in Step 2, and a zoomed-in version shows a gradual drift (Step 3).

Parameter Configuration. CanGene also provides a configuration file where users may specify and tune parameters, as shown in Figure 9. Each interval may be specified via 'anomaly_step' with 'num_intervals' and 'gap_size'. All anomaly types and their corresponding parameters are defined in 'anomaly_params'. Drift generation parameters are defined in 'drift_params'. Users may input the configured *.yaml configuration file into the CanGene user interface.

Case 1: Anomalous weather events. We simulate anomalous weather events in a two-phase approach: (1) injecting point anomalies to represent extreme weather events such as thunderstorms, snow storms, extreme heat that cause severe changes in temperature; and (2) seasonal weather events such as monsoons and heatwaves which span longer time duration, and re-occur within a season or within a year, represented by collective and periodic anomalies, respectively.

We will demonstrate adding point anomalies throughout T^*, and then divide the resulting T_A^* into two intervals to inject collective and periodic anomalies to simulate more seasonal and recurring weather events. Figure 10a shows injected anomalies over the Greece (GR), Norway (NO), and Germany (DE) temperature data over a three-year span, and the collective and periodic anomalies in the first and latter 1.5 years, respectively. Figure 10b (left) shows a zoomed-in view of a 4-week time frame (shown via the green dotted line in Figure 10a) highlighting the change in temperature due to the *point and collective anomalies*. Similarly, Figure 10b (right) highlights the change in temperature due to injected *point and periodic anomalies*.

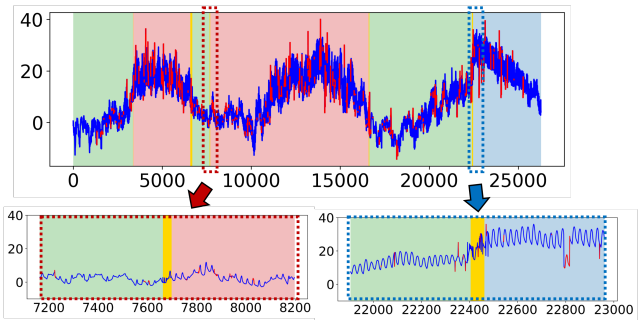
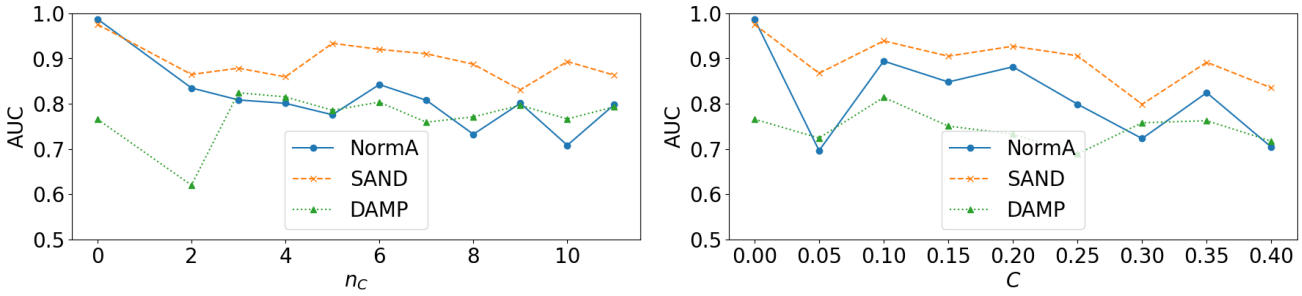


Figure 11: Drift generation reflecting seasonal weather transitions.

In this example, we use point anomalies with $a = 0.01$, $r = 100$, Uniform distribution. For the collective anomalies, we use two intervals, $g = 2000$, $r = 50$, Gaussian distribution with ($\mu = 0, \sigma = 0.2$), $\lambda = 6$, $a = 0.01$. For periodic anomalies, we used similar settings except with a Gaussian distribution ($\mu = 0$, $\sigma = 0.5$), and $a = 0.05$. Users will interactively adjust parameter values to visualize changes in the data due to varying anomaly types, anomaly percentage, distribution, and their co-occurrence frequency.

Case 2: Seasonal transitions. In the second case study, we demonstrate CanGene's drift generation features. We will guide a user through the drift generation process of selecting streams to use as concepts, and evaluate varying parameter values for the number of drifts, the percentage of data to use to simulate short vs. long duration weather events (heavy rain lasting a few hours vs. weeks-long heat waves), and the occurrence of these events in conjunction with extreme weather events. We will show how the above anomaly-injected streams are used as inputs to generate recurring and abrupt concept drifts due to seasonal weather events and heat waves, respectively.

Figure 11 shows five injected drifts (in yellow) where the blue, green and red background indicate temperature data streams from Greece, Norway, and Germany, respectively. Recurring drifts can be generated in CanGene by injecting a larger number of drifts than the



(a) AUC vs. n_C ($C = 0.2$). **(b) AUC vs. C ($n_C = 10$).**

Figure 12: Comparative performance of anomaly detection baselines for varying n_C and C.

available number of input data streams, e.g., transition from Norway (green) to Germany (red) and back to Norway. Figure 11 (bottom-left) show zoomed-in details of the drift reflecting how temperatures from Norway (cool weather) transition to moderate temperatures in Germany, depicting the transition to the Autumn season. Similarly, Figure 11 (bottom-right) shows a drift from temperatures in Norway to warm temperatures in Greece, reflecting the transition to the sudden, increased temperatures of a heat wave.

Table 3: ECG dataset characteristics.

	ECG803	**ECG805**	**ECG806**
data size	230,400	230,400	230,400
# anomalies	11,025 (4.8%)	20200 (8.8%)	5,214 (2.3%)

Case 3: Impact of concept drift on anomaly detection. We demonstrate the utility of our generated drift(s) dataset by evaluating a set of baseline anomaly detection algorithms. We implement the following algorithms using the TSB-UAD benchmark containing univariate time-series anomaly detection methods [23]. We use three ECG datasets with anomaly labels, with data characteristics shown in Table 3 [22].

NormA [6]: NormA derives to keep multiple normal patterns to identify anomalies, based on their frequency of appearance and similarity. It computes the anomaly score as the weighted sum of differences from all normal patterns.
SAND [7]: SAND extends NormA online by updating normal patterns in real-time batches, helping to understand changes in these patterns. Due to the batch processing, it takes time to adapt to changes, as it needs to re-balance the weight of each normal patterns based on its frequency.
DAMP [21]: DAMP compares an incoming subsequence with the previously seen time series data in the backward direction, to find the similar subsequences. It computes the anomaly score based on the distance of the most similar subsequence within a certain time range.

For each ECG dataset, we vary two drift parameters: (1) n_C: the number of drifts from [2, 11] in increments of two; and (2) C: the percentage of drift from [0.05, 0.4] in 0.05 increments. We computed the Area Under the Curve (AUC) to eliminate the impact of the detection threshold. Figure 12a and Figure 12b show the comparative AUC comparison for varying n_C and C, respectively. To establish a starting point (with no generated concept drift), we run the three anomaly detection algorithms on the original datasets (without concept drift), and report the averaged AUC over 5-runs as $n_C = C = 0$. We observe that for NormA and SAND, the AUC decreases as n_C and C increase. SAND shows higher AUC than NormA due to its ability to differentiate between recent vs. historical normal patterns in online settings. However, SAND incurs a significantly higher computational overhead than NormA (greater than 20x in our evaluation). DAMP, demonstrated more stable overall performance in the presence of increasing concept drift. However, DAMP showed high sensitivity to small data fluctuations that were not anomalies (noise) leading to more false positives.

4 CONCLUSION

Although anomalies and concept drifts commonly occur in real time series data, there are few datasets that include labels for both types of events. In this work, we present CanGene, a data generation tool that addresses the lack of tools for *both* anomaly injection and concept drift generation. By leveraging existing real data, CanGene supports injecting point, collective, and periodic anomalies according to uniform, Gaussian, or skew-normal random distributions. In addition, using these anomaly-labeled time series data, CanGene supports generation of abrupt, gradual and recurring drifts relative to these anomalies. We demonstrate three case studies showing the data utility of the generated data, and the comparative performance of three time series, anomaly detection baseline methods in the presence of concept drift.

REFERENCES

[1] 2017. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* 262 (2017), 134–147.
[2] Cesare Alippi, Giacomo Boracchi, and Manuel Roveri. 2016. Hierarchical change-detection tests. *IEEE transactions on neural networks and learning systems* 28, 2 (2016), 246–258.
[3] Albert Bifet and Ricard Gavaldà. 2007. Learning from Time-Changing Data with Adaptive Windowing. *Proceedings of the 2007 SIAM International Conference on Data Mining (SDM)* (2007), 443–448.

[4] Albert Bifet, Geoff Holmes, Richard Kirkby, and Bernhard Pfahringer. 2010. MOA: Massive Online Analysis. *Journal of Machine Learning Research* 11, 52 (2010), 1601–1604.

[5] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A. Lozano. 2021. A Review on Outlier/Anomaly Detection in Time Series Data. *Comput. Surveys* 54, 3 (Apr 2021), 33.

[6] Paul Boniol, Michele Linardi, Federico Roncallo, Themis Palpanas, Mohammed Meftah, and Emmanuel Remy. 2021. Unsupervised and Scalable Subsequence Anomaly Detection in Large Data Series. *The VLDB Journal* 30, 6 (nov 2021), 909–931.

[7] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. 2021. SAND: streaming subsequence anomaly detection. *Proceedings of the VLDB Endowment* 14, 10 (2021), 1717–1729.

[8] Rodolfo C. Cavalcante, Leandro L. Minku, and Adriano L. I. Oliveira. 2016. FEDD: Feature Extraction for Explicit Concept Drift Detection in time series. In *2016 International Joint Conference on Neural Networks (IJCNN)*. 740–747.

[9] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide and Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (DLRS 2016)*. 7–10.

[10] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[11] Ronald Gelaro, Will McCarty, Max J. Suárez, Ricardo Todling, Andrea Molod, Lawrence Takacs, Cynthia A. Randles, Anton Darmenov, Michael G. Bosilovich, Rolf Reichle, Krzysztof Wargan, Lawrence Coy, Richard Cullather, Clara Draper, Santha Akella, Virginie Buchard, Austin Conaty, Arlindo M. da Silva, Wei Gu, Gi-Kong Kim, Randal Koster, Robert Lucchesi, Dagmar Merkova, Jon Eric Nielsen, Gary Partyka, Steven Pawson, William Putman, Michele Rienecker, Siegfried D. Schubert, Meta Sienkiewicz, and Bin Zhao. 2017. The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2). *Journal of Climate* 30, 14 (2017), 5419 – 5454.

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

[13] Aditya Gopalan, Braghadeesh Lakshminarayanan, and Venkatesh Saligrama. 2021. Bandit Quickest Changepoint Detection. In *Advances in Neural Information Processing Systems*, Vol. 34.

[14] Guolin Ke, Zhenhui Xu, Jia Zhang, Jiang Bian, and Tie-Yan Liu. 2019. DeepGBM: A Deep Learning Framework Distilled by GBDT for Online Prediction Tasks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Anchorage, AK, USA) *(KDD '19)*. 384–394.

[15] E. Keogh. 2021. Multi-dataset Time-Series Anomaly Detection Competition. *SIGKDD 2021* (2021).

[16] Pratibha Kumari and Mukesh Saini. 2021. Anomaly Detection in Audio with Concept Drift using Adaptive Huffman Coding. *ArXiv* abs/2102.10515 (2021).

[17] Kim-Hung Le and Paolo Papotti. 2020. User-driven Error Detection for Time Series with Events. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. 745–757.

[18] Mengchu Li and Yi Yu. 2021. Adversarially Robust Change Point Detection. In *Advances in Neural Information Processing Systems*, Vol. 34. 22955–22967.

[19] Wendi Li, Xiao Yang, Weiqing Liu, Yingce Xia, and Jiang Bian. 2022. DDG-DA: Data Distribution Generation for Predictable Concept Drift Adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence* 36, 4 (Jun. 2022), 4092–4100.

[20] Jie Lu, Anjin Liu, Fan Dong, Feng Gu, João Gama, and Guangquan Zhang. 2019. Learning under Concept Drift: A Review. *IEEE Transactions on Knowledge and Data Engineering* 31, 12 (2019), 2346–2363.

[21] Yue Lu, Renjie Wu, Abdullah Mueen, Maria A Zuluaga, and Eamonn Keogh. 2022. Matrix profile XXIV: scaling time series anomaly detection to trillions of datapoints and ultra-fast arriving data streams. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1173–1182.

[22] G.B. Moody and R.G. Mark. 2001. The impact of the MIT-BIH Arrhythmia Database. *IEEE Engineering in Medicine and Biology Magazine* 20, 3 (2001), 45–50.

[23] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: An End-to-End Benchmark Suite for Univariate Time-Series Anomaly Detection. *Proc. VLDB Endow.* 15, 8 (apr 2022), 1697–1711.

[24] Abdulhakim A. Qahtan, Basma Alharbi, Suojin Wang, and Xiangliang Zhang. 2015. A PCA-Based Change Detection Framework for Multidimensional Data Streams: Change Detection in Multidimensional Data Streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 935–944.

[25] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. *Proc. VLDB Endow.* 15, 9 (may 2022), 1779–1797.

[26] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Zachary Zimmerman, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. 2017. Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile. *Data Mining and Knowledge Discovery* (2017), 1–41.