Estimating Mutual Information between Time Series and Temporal Event Sequences Across Diverse Analysis Tasks

Haoji Hu*
huxxx899@umn.edu
University of Minnesota - Twin Cities
Minneapolis, MN, USA

Huaqing Mao*
maohqphy@hotmail.com
University of Minnesota - Twin Cities
Minneapolis, MN, USA

Yijun Lin
lin00786@umn.edu
University of Minnesota - Twin Cities
Minneapolis, MN, USA

Xiaowei Jia
xiaowei@pitt.edu
University of Pittsburgh
Pittsburgh, PA, USA

Jinwei Zhou
zhou1909@umn.edu
University of Minnesota - Twin Cities
Minneapolis, MN, USA

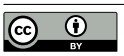
Minoh Jeong
minohj@inha.ac.kr
Inha University
Incheon, Incheon, Republic of Korea

Yao-Yi Chiang
yaoyi@umn.edu
University of Minnesota - Twin Cities
Minneapolis, MN, USA

Abstract

Pairwise dependence measures such as correlation and causality are fundamental to temporal data mining, yet there is still no principled and robust way to quantify dependence between heterogeneous data types, especially between continuous time series and discrete temporal event sequences. Existing approaches rely on ad hoc transformations or mutual-information estimators that are highly sensitive to quantization, repeated values, and event redundancy, leading to biased or unstable results in practice. We propose a nonparametric mutual information estimator that directly measures the dependence between time series and event sequences without data transformation, learning, or ad hoc discretization. Our method models the continuous–discrete duality of real-world time series to handle quantization and repeated-value artifacts and introduces a latent event clustering strategy to mitigate bias from event co-occurrence and redundancy. Together, these yield a robust and unified framework that bridges discrete and continuous mutual information. We evaluate the proposed estimator on four representative tasks: discrete–continuous time-delayed mutual information for causality analysis, global and local temporal repetition discovery, discrete covariate selection for time series forecasting, and continuous feature selection for classification. Experiments on synthetic and real-world datasets show consistent improvements over existing methods in accuracy, robustness, and interpretability, positioning our approach as a general-purpose dependence operator for heterogeneous temporal data, similar to Pearson correlation for homogeneous time series. Code: https://github.com/HaojiHu/Multimodal-Temporal-Data-Quantification

*Both authors contributed equally to this research. This paper has been accpeted to KDD 2026.



KDD '26, Jeju Island, Republic of Korea

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

CCS Concepts

• **Mathematics of computing** → **Information theory**; • **General and reference** → Measurement.

Keywords

Mutual Information, Multimodal Temporal Data, Time Series, Temporal Event Sequences, Seasonality, Feature Selection, Covariate

ACM Reference Format:
Haoji Hu, Huaqing Mao, Yijun Lin, Xiaowei Jia, Jinwei Zhou, Minoh Jeong, and Yao-Yi Chiang. 2026. Estimating Mutual Information between Time Series and Temporal Event Sequences Across Diverse Analysis Tasks. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Time series and temporal event sequences are two of the most common and fundamental forms of temporal data in real-world applications. Time series represent continuous-valued measurements evolving over time, such as temperature, traffic volume, or magnetic field intensity, whereas temporal event sequences consist of discrete events characterized by timestamps and symbolic event types, such as system alerts, transactions, or scientific phenomena. These two data types naturally coexist in many domains, where continuous physical or socioeconomic processes are influenced by, or give rise to, discrete events, and vice versa.

Quantifying pairwise relationships between **time series** and **temporal event sequences** is a core problem in temporal data mining and supports a wide range of downstream tasks, including correlation and causality analysis, temporal pattern discovery, and feature or covariate selection. For example, in solar physics, estimating the time-delayed mutual information (TDMI) between discrete solar event sequences and continuous geomagnetic field measurements is crucial to understanding how and when solar activity influences Earth's magnetic field [9]. Similar heterogeneous

temporal relationships appear in transportation systems (traffic volume versus calendar events) and sales forecasting (sales time series versus promotions), highlighting their broad practical importance.

Most existing dependence measures focus on homogeneous temporal data. Pearson correlation and its variants quantify relationships between two time series (Figure 1(a)), while point-wise mutual information measures dependencies between two temporal event sequences (Figure 1(b)). These methods are effective within a single data type but do not directly support cross-type temporal analysis, such as measuring the interaction between a continuous time series and a discrete temporal event sequence (Figure 1(c)). A common workaround is to transform one temporal data type into another and then apply standard homogeneous measures. Typical approaches include discretizing time series into symbolic sequences via binning [23], or converting event sequences into numerical series by assigning integer identifiers (IDs) to symbols [29]. Such transformations introduce ad hoc design choices (e.g., bin width or symbol ordering) and impose non-existent artificial structures, leading to biased estimates and reduced interpretability. Some prior work has explored event–time series relationships using hypothesis testing that analyzes signal behavior around event occurrences [24], but does not explicitly quantify dependence. More importantly, existing approaches fail to account for real-world characteristics of temporal data, such as sensor quantization effects, repeated values in time series, and correlations among events in temporal event sequences.

Mutual information (MI) is a model-free, principled measure of statistical dependence that applies to both discrete and continuous variables, making it a natural choice for cross-type temporal analysis. However, existing MI estimators do not adequately handle heterogeneous temporal data in practice. They face two fundamental challenges: (1) real-world time series often violate strict continuity assumptions of continuous variables due to finite precision and repeated values; for example, temperature recorded at integer or low-decimal resolution usually takes identical values across many timestamps. (2) Event types are typically treated as independent symbols, ignoring redundancy in their induced distributions over continuous values. These mismatches between theoretical assumptions and real data lead to unstable and biased estimates.

In this paper, we propose a nonparametric MI estimator that directly quantifies dependence between a time series and a temporal event sequence, without data transformation, discretization heuristics, or learning-based modeling. Our method introduces two key ideas: (1) a *continuous–discrete duality representation* that explicitly models the coexistence of continuous variation and repeated values caused by finite measurement precision; and (2) an optional *latent event clustering* strategy that aggregates redundant or highly correlated event types into representative latent events. Together, our proposed estimator unifies classical discrete and continuous MI and serves as a general-purpose dependence operator for heterogeneous temporal data. We demonstrate the generality and effectiveness of the proposed estimator through four representative tasks involving time series and temporal event sequences.

(1) Time-delayed mutual information estimation, extending the quantification of lagged dependencies from homogeneous time series to time series–event sequence pair;

(2) Temporal repetition analysis, identifying repeated temporal structures in time series, including both global seasonality and localized, event-conditional repetition patterns;

(3) Covariate analysis for time series forecasting, ranking discrete covariates by their relevance to a continuous target time series to support reliable and interpretable forecasting; and

(4) Extension to mixed-type tabular data analysis, extending beyond temporal settings to enable feature selection by measuring the dependence between continuous features and discrete labels.

Extensive experiments on synthetic and real-world datasets show that the proposed measurement consistently outperforms existing methods and offers interpretable results, establishing it as a foundational building block for heterogeneous temporal data mining. In summary, our main contributions are:

(1) A nonparametric MI estimator that directly quantifies dependence between a time series and a temporal event sequence while accounting for quantization and repeated values.

(2) A clustering strategy that reduces redundancy from event co-occurrence in event sequences.

(3) An empirical evaluation demonstrating effectiveness across core data mining tasks, including cross-modal TDMI estimation, temporal repetition pattern discovery, discrete covariate selection for forecasting, and continuous feature selection for classification.

2 Related Work

2.1 Mutual Information Estimation

Estimating mutual information (MI) for continuous variables is a long-standing challenge [31]. Existing parametric methods rely on strong assumptions about the underlying data distribution (e.g., Gaussianity), which are often violated in real-world temporal data. In contrast, nonparametric methods are more flexible and make fewer assumptions, making them the dominant choice in practice.

Nonparametric MI estimators such as KSG [19] and its extensions [26] are built upon the Kozachenko–Leonenko entropy estimator [18]. These methods assume strictly continuous variables, under which the probability of observing identical samples is zero (i.e., the nearest-neighbor distance is always positive). In real-world time series, however, values are observed with finite precision due to sensor limitations, rounding, or quantization, easily leading to repeated values. When duplicates occur, nearest-neighbor distances collapse to zero, producing degenerate neighborhoods and unstable entropy or MI estimates. This issue is often ignored, heuristically mitigated, or assumed to vanish in high-dimensional data, but it becomes critical in low-dimensional temporal settings (e.g., univariate time series). We address this mismatch between theoretical assumptions and practical data by modeling a time series as a mixture of continuous and discrete components. This formulation treats repeated values as discrete probability mass rather than anomalous cases, thereby resolving a core limitation of existing nonparametric MI estimators without resorting to ad hoc fixes. Recent progress like [3][8] learn an estimator. But, these methods require model training and fail to directly estimate the MI for just one pair data.

2.2 Temporal Data Analysis

Temporal dependence analysis has been extensively studied for homogeneous data types. Classical approaches such as canonical

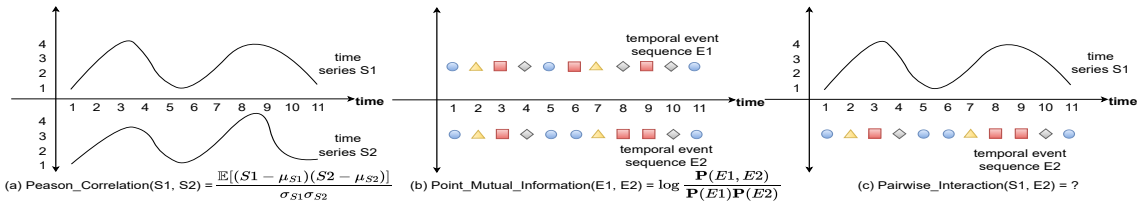


Figure 1: Dependence analysis across temporal data types. (a) Pearson correlation between two time series, where each time step is a continuous value. (b) Point-wise MI between two temporal event sequences, where each time step is a discrete event (shape + color), and marginal and joint probabilities are estimated from event frequencies. (c) Unknown cross-type interaction between a time series and a temporal event sequence.

correlation analysis [1], co-occurrence analysis [10], and sequence similarity measures [25] focus on quantifying dependence within a single temporal data type. In contrast, cross-type temporal analysis between continuous time series and discrete-event sequences has received far less attention. Yet, such heterogeneous temporal relationships are ubiquitous in real-world applications. In this paper, we focus on three representative tasks that highlight the importance of this problem. We review related work for each task below.

Time-Delayed Mutual Information Estimation. Existing TDMI methods assume both variables are continuous time series and are therefore not applicable when one variable is a temporal event sequence. While discrete–continuous MI estimators [14, 29] can in principle be applied, they rely on neighborhood-based density estimation, inherit strict continuity assumptions, and ignore quantization artifacts and event redundancy. Our approach directly models time series–event sequence interactions, enabling robust and interpretable cross-type TDMI estimation.

Temporal Seasonality and Repetition Analysis. Seasonality detection has traditionally relied on analyzing a time series as a continuous signal. Many techniques have been developed for this purpose, including autocorrelation function (ACF) [6], seasonal decomposition method STL [11], and Fourier transform (FT) [7].

ACF detects periodicity by identifying peaks in specific lags. Fourier methods identify dominant frequencies associated with repeating patterns. Despite their effectiveness, these techniques are sensitive to outliers and missing values, as each time step is modeled deterministically. Such irregularities can significantly distort the resulting ACF or FT.

Our method reframes seasonality as a dependence problem between a time series and a temporal event sequence (e.g., calendar structure), enabling a probabilistic treatment that is robust and unifies global and localized repetition patterns.

Covariate Analysis and Feature Selection. Feature and covariate selection methods are two parallel analysis tasks that share common problem structures, especially for continuous feature selection and discrete covariate selection. Both model the interaction between continuous values and discrete symbols, i.e., selecting continuous features for discrete-label prediction and discrete covariates for time-series forecasting. Although model-based methods have been proposed [16][4], due to model bias, model-agnostic methods are more widely used. Information-theory-based methods [13, 23] are representative model-agnostic methods. However, existing methods often rely on discretization when continuous variables are involved. A common strategy is binning [13, 23], which partitions the continuous value range into k intervals and assigns each interval a discrete symbol. Although simple and widely used, binning introduces a strong inductive bias through the choice of bin boundaries and widths. Moreover, discretization assumes a one-to-one mapping between continuous values and discrete symbols. The same continuous value may be associated with multiple discrete events, and forcing a hard partition can distort this relationship and reduce the reliability of feature relevance estimates.

These limitations motivate the need for a measure that can directly quantify relationships between continuous values and discrete events. Our proposed estimator addresses this gap and provides a principled alternative for covariate analysis and feature selection in heterogeneous data.

3 Information Theory based Measurement

Problem definition: Given a time series of continuous values $\mathbf{S} = (s_1, s_2, ..., s_m)$, where m represents the number of time steps, and a temporal event sequence of discrete symbols $\mathbf{E} = (e_1, e_2, ..., e_n)$, where n is the number of observed events, our goal is to quantify their interaction, denoted as $\mathrm{IQ}(\mathbf{S}, \mathbf{E})$, which measures the degree of correlation between temporally aligned elements of $\mathbf{S}$ and $\mathbf{E}$. The time indices encode only relative temporal order; two modalities need not share identical timestamps and may be sampled asynchronously across data sources.

Our proposed measurement builds on mutual information (MI), a principled measure of statistical dependence between two random variables. We first present an MI formulation between a discrete and a continuous variable by unifying Shannon MI for discrete variables with differential MI for continuous variables (Section 3.1). For continuous-valued time series $\mathbf{S}$, we introduce a continuous–discrete duality representation that models the empirical distribution as a mixture of a continuous component and a precision-induced discrete component, and we extend this formulation to estimate MI between a mixture variable and a discrete variable (Section 3.2). For a discrete event sequence $\mathbf{E}$, we introduce an optional clustering strategy that maps event types to a latent event sequence with reduced cardinality, allowing correlated or redundant event types to be modeled jointly (Section 3.3).

3.1 MI of Discrete-Continuous Variables

In Shannon mutual information, we consider two discrete random variables X and Y, where x and y denote their values, and $\mathcal{X}$ and $\mathcal{Y}$ denote their domains, i.e., $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. If both $\mathcal{X}$ and $\mathcal{Y}$ are

subsets of a countable set of categories $\{C_1, C_2, ..., C_n\}$ (i.e., n is the cardinality size of the countable set and we can get the union set of both sets as the countable set), the MI can be defined based on as,

$$\mathrm{I}(X,Y) = \sum_{y\in\mathcal{Y}}\sum_{x\in\mathcal{X}} P_{(X,Y)}(x,y)\log\left(\frac{P(x,y)}{P(x)P(y)}\right) \tag{1}$$

In differential mutual information, for two continuous random variables X and Y, where the range of x and y is $\mathbb{R}$, the MI is,

$$\mathrm{I}(X,Y) = \int_{\mathcal{Y}}\int_{\mathcal{X}} P(x,y)\log\left(\frac{P(x,y)}{P(x)P(y)}\right)\mathrm{d}x\,\mathrm{d}y \tag{2}$$

A natural extension of Eqs. 1 and 2 defines the MI between a discrete variable X and a continuous variable Y. Specifically, we assume that X takes values from a countable set of categories $\mathcal{X} = C_1, C_2, \ldots, C_n$, and Y takes values in the continuous domain $\mathcal{Y} = \mathbb{R}$. The MI between them is,

$$\mathrm{I}(X,Y) = \sum_{x\in\mathcal{X}}\int_{y\in\mathcal{Y}} P(x,y)\log\left(\frac{P(x,y)}{P(x)P(y)}\right)\mathrm{d}\,y \tag{3}$$

Equivalently, the MI can be expressed in terms of the entropy as: $\mathrm{I}(X,Y) = H(X) + H(Y) - H(X,Y)$, We can rewrite Eq. 3 as

$$\begin{aligned}\mathrm{I}(X,Y) = &-\sum_{x\in\mathcal{X}} P(x)\log P(x) - \int_{\mathcal{Y}} P(y)\log P(y)\mathrm{d}y \\ &+ \sum_{x\in\mathcal{X}}\int_{\mathcal{Y}} P(x,y)\log P(x,y)\mathrm{d}y\end{aligned} \tag{4}$$

If we fix a specific value of x and only consider Y as a variable, the joint entropy in Eq. 4 can be written as

$$\begin{aligned}\int_{\mathcal{Y}} P(x,y)\log P(x,y)\,\mathrm{d}y = &\, p(x)\log p(x) \\ &+ p(x)\int_{\mathcal{Y}} P(y\mid x)\log P(y\mid x)\,\mathrm{d}y\end{aligned} \tag{5}$$

where the marginal distributions are $P(x) = p(x)$ and $P(y) = \sum_{x\in\mathcal{X}} p(x)P(y|x)$. The joint probability distribution becomes,

$$P(x,y) = p(x)P(y|x),\ \sum_{x\in\mathcal{X}} p(x) = 1,\ \int_{\mathcal{Y}} P(y|x)\,\mathrm{d}y = 1$$

Based on Eqs. 4 and 5, the $\mathrm{I}(X,Y)$ between X and Y is,

$$\mathrm{I} = \sum_{x\in\mathcal{X}} p(x)\int P(y|x)\log p(y|x)\,\mathrm{d}y - \int P(y)\log P(y)\,\mathrm{d}y \tag{6}$$

The probability mass function of the discrete variable X, denoted by $p(x)$, can be directly estimated from data using empirical frequencies of the corresponding discrete value x. In contrast, estimating the distribution of a continuous variable is more challenging. A common approach is to assume a Gaussian distribution, which is parametric and relies on the strong assumption that the data follow a unimodal normal distribution. This assumption is often unrealistic for temporal data, which frequently exhibit nonstationarity and complex dynamics. To avoid such restrictive assumptions, we adopt a classical nonparametric and asymptotically unbiased entropy estimator introduced by [18], which directly estimates the entropy of a continuous variable from samples. Specifically, given N samples ${x_i}_{i=1}^N$, the entropy is estimated as: $H = ln(\overline{\rho}) + ln(c) + ln(\gamma) + ln(N-1)$, where $\overline{\rho} = \{\prod_{i=1}^{N}\rho_i\}^{1/N}$ (i.e., $\rho_i = min(\{|x_i - x_j|, j \neq i\})$), $c = \pi^{\frac{1}{2}}/\Gamma(\frac{3}{2})$, and $ln(\gamma)$ is the Euler constant (≈ 0.5772).

MI takes values in the range $[0, \infty)$, which makes its magnitude difficult to interpret without knowing max pivot. To obtain a normalized measure analogous to Pearson correlation, which lie in $[0, 1]$, it is common to normalize MI. Among the various normalization schemes summarized in [26], we adopt [21],

$$\mathrm{NMI}(X,Y) = \sqrt{1 - \exp[-2\,\mathrm{I}(X,Y)/(\mathrm{d}X + \mathrm{d}Y)]} \tag{7}$$

As this transformation is for continuous variables and the unit $\mathrm{d}X$ is not defined for discrete symbols, we propose an approximation $\mathrm{d}X$=1, as symbols are usually represented by integer IDs.

Although the entropy of a continuous variable can be estimated using nonparametric entropy estimators, differential entropy is not guaranteed to be nonnegative. By definition, $H(Y) = -\int_{\mathcal{Y}} P_Y(y)\cdot\log P_Y(y)\,\mathrm{d}\,y$, $\int_{\mathcal{Y}} P_Y(y)\,\mathrm{d}\,y = 1$, and $\log P_Y(y) \in (-\inf, \inf)$.

Since our MI estimation is derived from entropy terms of continuous variables (see Eq. 6), negative differential entropy estimates may propagate and yield negative MI. This violates the fundamental non-negativity property of MI and indicates a failure of the estimation. We provide a formal proof of this property in the Appendix.

This issue can be formally addressed by introducing an invariant measure density $m(y)$ to normalize the logarithmic term and convert differential entropy into a relative entropy form, $H(Y) = \int P(y)\log\frac{P(y)}{m(y)}\,\mathrm{d}\,y$, as proposed in [17]. Fortunately, incorporating $m(y)$ does not affect the value of MI, since the relevant terms cancel out when computing MI (see Eq. 4) [26]. Therefore, MI remains unchanged regardless of whether the invariant measure is explicitly introduced, and we omit $m(y)$ in our implementation for simplicity.

Another issue that is largely overlooked in prior analyses [26] is the possibility of obtaining negative estimates due to finite-sample variability. When the true MI is close to zero, small estimation errors in the entropy terms can cause the estimated values to slightly undershoot their true magnitudes, resulting in negative entropy contributions and, consequently, negative MI. Although such outcomes are theoretically impossible, they can arise in practice due to estimation noise. To guarantee the non-negativity of MI, we apply a simple truncation: $I \leftarrow \max(I, 0)$. This operation leaves all valid positive estimates unchanged while removing spurious negative values introduced by estimation noise. It provides a numerically stable, theoretically consistent safeguard to improve the robustness without affecting meaningful dependence measurements.

3.2 Continuous-Discrete Duality

Modeling a time series purely as a continuous variable with a density function (e.g., P_Y) overlooks the fact that real-world measurements often exhibit ordinal characteristics. Ordinal data are categorical in nature but preserve a meaningful ordering among possible values. For example, temperature observations are bounded by physical limits and are recorded with finite precision due to instrument constraints, such as rounding to one decimal place. As a result, the observed values belong to a countable and ordered set rather than a truly continuous domain, a property that cannot be faithfully captured by a purely continuous density model. Such discretization with preserved value ordering makes the observed data analogous to ordinal data. However, temperature itself is a continuous macroscopic variable defined over a thermodynamic system [20]. The discreteness arises not from the underlying physical process, but from measurement and recording constraints. Consequently, an observed temperature time series is jointly shaped by the continuous

nature of the physical phenomenon and the discrete bias introduced by finite-precision instruments. More generally, real-world time series may exhibit purely continuous behavior, purely ordinal behavior, or a combination of both. This motivates a *continuous–discrete duality* in the representation of time series values. That is, a time series variable should simultaneously capture continuous variation and discrete probability mass, and flexibly adjust its relative contributions based on the observed samples.

Formally, we redefine the density function of continuous variable Y as a combination of a continuous distribution $f(y)$ and a discrete distribution $g(y)$, as $p(y) = a\, f(y) + b\, g(y)$, where $\int f(y)dy = 1$, $\sum_y g(y) = 1$, $f(y) \cdot g(y) = 0$, and $a + b = 1$.

This formulation is interpreted as: each observed timestep value is generated from either the continuous distribution $f(y)$ with probability a or the discrete distribution $g(y)$ with probability b.

Given this new formulation of $p(y)$, the entropy of Y can be rewritten with the law of addition for the information entropy applied to both distributions $f(y)$ and $g(y)$ as follows:

$$\begin{aligned} H(p) &= \int a\, f(y) ln(a\, f(y))dy + \sum_y b\, g(y) ln(b\, g(y)) \\ &= a\, ln(a) + a\, H(f(y)) + b\, ln(b) + b\, H(g(y)) \end{aligned} \tag{8}$$

For the joint entropy $H(X, Y)$ between the discrete variable X and the redefined continuous–discrete variable Y, we decompose it into a weighted combination of two components, denoted by $A(X, Y)$ and $B(X, Y)$. Here, $A(X, Y)$ represents the joint entropy contributed by the samples in which Y is generated from its continuous component and their aligned values in X, while $B(X, Y)$ represents the joint entropy contributed by the samples in which Y is generated from its discrete component and their aligned values in X. By the additivity property of entropy over mixture distributions, the joint entropy $H(X, Y)$ can therefore be expressed as a linear combination of these two terms:

$$H(X, Y) = a\, ln(a) + a\, H(A) + b\, ln(b) + b\, H(B) \tag{9}$$

With Eqs. 4, 8, and 9, the mutual information is:

$$I(X, Y) = a\ \mathrm{I}_A + b\ \mathrm{I}_B + a\, ln(a) + b\, ln(b) \tag{10}$$

where $\mathrm{I}_A = \mathrm{I}(X_A, Y_A)$ can be calculated by Eq. 6 and $\mathrm{I}_B = \mathrm{I}(X_B, Y_B)$ is calculated by Eq. 1.

The remaining question is how to partition the time series Y into its continuous and discrete components. We propose to do so based on the presence of repeated values. Specifically, if a value appears only once in the entire series, we treat the corresponding time step as being generated from the continuous component $f(y)$ and include it in Y_A for computing the MI $I(X_A, Y_A)$. The event aligned with this time step is accordingly assigned to X_A. In contrast, if a value appears more than once, all time steps with this value are treated as being generated from the discrete component $g(y)$ and included in Y_B, and their aligned events are included in X_B for the calculation $I(X_B, Y_B)$. This partitioning is theoretically justified by the assumptions underlying classical nonparametric entropy estimators such as [18], which are developed for strictly continuous variables, where the probability of observing identical samples is zero. In such a setting, any repeated value violates the continuity assumption, leading to degenerate neighborhoods.

3.3 Latent Discrete Variable

In practice, temporal event sequences often contain event types that are highly correlated or redundant. In such cases, distinct event symbols may correspond to nearly identical conditional distributions over the aligned time-series values Y, leading to unnecessary fragmentation and unstable dependence estimates. To address this issue, we introduce an optional event-clustering strategy that aggregates similar event types into latent events. The key idea is to replace redundant symbolic distinctions with a more compact representation that better reflects the data generation process.

This procedure consists of three steps: (1) for each event type, collecting all temporally aligned values from the time series Y to form its empirical conditional distribution; (2) performing hierarchical clustering over these distributions using the Wasserstein distance; and (3) replacing the original event type with its corresponding cluster label, yielding a latent event sequence. This transformation reduces redundancy while preserving the discrete nature and temporal structure. Importantly, the clustering does not alter the data modality and is applied only when strong correlations or overlaps among event types are present.

4 Experiments

Evaluating mutual information (MI) estimators is inherently challenging because ground-truth are not available for real-world data. As a result, prior work has primarily relied on synthetic datasets for controlled evaluation [14, 29]. Building on this practice, we first validate our method using carefully designed synthetic data with known ground truth. Beyond synthetic evaluation, we further assess the effectiveness of our measurement through a set of real-world data and analytic tasks in which MI serves as a core component. Improvements in task performance provide complementary and practical evidence of the utility and robustness of the proposed estimator in realistic settings.

4.1 Time-Delayed Mutual Information

This task evaluates the estimation of time-delayed mutual information (TDMI) between a time series S and a temporal event sequence E. Given a time lag τ, TDMI measures the dependence between the event type occurring at time t in E and the value of the time series at time $t + \tau$. A key advantage of the proposed measurement is that it natively supports heterogeneous variables, allowing the time delay to be incorporated directly into the MI formulation without transforming either modality. Specifically, TDMI is computed as

$$\mathrm{TDMI}(E(t), S(t + \tau)) = I(E(t), S(t + \tau)).$$

Dataset. We construct a synthetic dataset to obtain ground-truth MI values. A temporal event sequence is generated by random sampling from three events (IDs are 1, 2, and 3) with equal probability. Each event aligns with a distinct Gaussian distribution. The corresponding distributions are $\mathcal{N}(-0.7, 0.02^2)$, $\mathcal{N}(0, 0.02^2)$, $\mathcal{N}(0.7, 0.02^2)$. When we sample an event, a corresponding continuous value is sampled from the aligned Guassian distribution for the time series, with a fixed time delay of $\tau = 5$. To better reflect real-world measurement properties, we model finite sensor precision by restricting time series values to two decimal places. This introduces quantization effects and repeated values, which are common in practice but rarely considered in prior evaluations. We generate

Table 1: TDMI results across time lags τ. Bold indicates the detected delay. Values in parentheses is mean squared error (MSE, 5 numerical precision) relative to G.T. Series2Seq and Seq2Series do not output discrete–continuous MI, so MSE is not reported.

τ	G.T.	Series2Seq	Seq2Series	Ross	Mixture	Ours
0	0.0376	0.0000	-0.0158	-3.1647 (10.254)	0.0312 (4.0E−05)	0.0353 (1.0E−05)
1	0.0423	0.0000	0.0095	-3.1639 (10.279)	0.0308 (1.3E−04)	0.0352 (5.0E−05)
2	0.0402	0.0000	-0.0071	-3.1633 (10.262)	0.0322 (6.0E−05)	0.0373 (1.0E−05)
3	0.0385	0.0000	0.0024	-3.1627 (10.247)	0.0329 (3.0E−05)	0.0387 (0)
4	0.0374	0.0000	0.0057	-3.1628 (10.241)	0.0314 (4.0E−05)	0.0366 (0)
5	1.0984	0.0000	**-0.4998**	**-3.1468** (18.022)	**1.1078** (9.0E−05)	**1.0988** (0)
6	0.0388	0.0000	-0.0151	-3.1655 (10.267)	0.0277 (1.2E−04)	0.0349 (2.0E−05)
7	0.0397	0.0000	-0.0016	-3.1639 (10.263)	0.0326 (5.0E−05)	0.0376 (0)
8	0.0393	0.0000	0.0006	-3.1657 (10.272)	0.0308 (7.0E−05)	0.0358 (1.0E−05)
9	0.0384	0.0000	-0.0063	-3.1652 (10.263)	0.0309 (6.0E−05)	0.0355 (1.0E−05)

10,000 time steps for both time series and temporal event sequence. The ground-truth MI is computed analytically from the known distributions using a Taylor-series approximation on the samples.

Baselines. We compare the proposed measurement against four representative baselines: (1) **Series2Seq**: discretizing the time series into a symbolic sequence via binning, followed by point-wise MI computation; (2) **Seq2Series**: converting the event sequence into a numeric time series using symbol IDs and measuring dependence via Pearson correlation; (3) **Mixture**: a discrete–continuous MI estimator based on mixture modeling [14]; and (4) **Ross**: a non-parametric estimator for discrete–continuous MI [29]. For a fair comparison of MI estimation accuracy, we do not apply the event clustering strategy in our method for this experiment.

Results. Table 1 reports TDMI estimates across lag steps $\tau = 0, \ldots, 9$. Transformation-based baselines Series2Seq and Seq2Series fail to recover the true dependence structure (i.e., Seq2Series has correlation less than 0.5), highlighting the limitations of forcing heterogeneous data into a homogeneous form. The Ross estimator produces negative values due to the heuristic resolution to handle the repeated values and sampling estimation variance around the 0 value, which violates the non-negativity property of MI (Theorem A.1 is also valid for Ross).

Both the Mixture estimator and our method correctly identify the peak dependence at $\tau = 5$ and near-zero dependence at other lags. However, our method consistently yields lower estimation error across different time lags τ. These results demonstrate that explicitly modeling finite precision and continuous–discrete duality leads to more accurate and stable TDMI estimation. Additional synthetic experiments under varying time series lengths and value precision are reported in the Appendix A. Our method consistently performs better over different data length and digit precision.

4.2 Repeated Temporal Patterns

This experiment evaluates the proposed measurement for identifying *repeated temporal patterns* in time series, including **global seasonality** and **local repeated patterns**. Our central contribution here is not only detecting whether a repeated pattern exists but also providing a *quantitative, interpretable, and scalable* measure of pattern strength between a time series and a temporal event sequence, even when patterns are imperfect, disrupted, or context dependent. Repeated temporal patterns are often evident to humans through visual inspection when the dataset is small. For example,

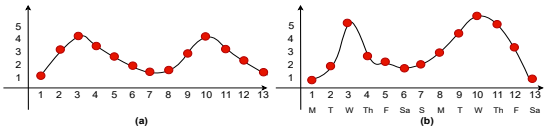


Figure 2: (a) Global seasonality. (b) Local repeated pattern on Wed.

Figure 2(a) shows a clear weekly seasonal pattern, while Figure 2(b) illustrates a local repeated pattern occurring only on Wednesdays. In such limited cases, humans can manually construct an implicit ground truth by identifying whether a pattern exists and which temporal contexts (e.g., day of week) are relevant. However, this form of visual inspection does not scale to large datasets, long time horizons, or noisy real-world measurements, which are common in practice. Instead, practical methods must (i) detect repeated temporal structure with flexibility and (ii) quantify its strength in a way that degrades gracefully under noise, missing values, and disruptions. Our proposed measurement is designed to fill this gap.

Seasonality refers to recurring behaviors that repeat at fixed intervals and follow a periodic structure. By mapping timestamps to calendar attributes, we construct a temporal event sequence whose event types correspond to days of the week. Using our proposed measurement, seasonality is quantified as the MI between the time series S and the calendar event sequence W:

$$\text{Seasonality}(S, W) = \sum_{w \in \{\text{Mon},\ldots,\text{Sun}\}} I(S, W = w).$$

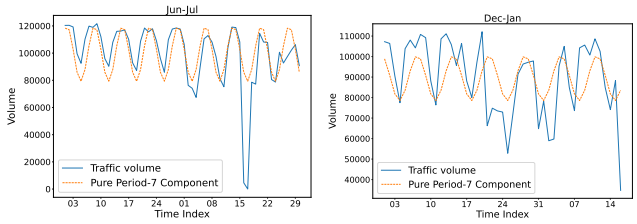
This formulation directly captures the dependence between continuous values and discrete temporal contexts, without assuming strict periodicity or requiring signal transformation. The proposed measurement can also capture **local repeated patterns** that arise only under specific temporal contexts. For example, Figure 2(b) illustrates a pattern that consistently occurs on Wednesdays but not on other days. Using our framework, such a structure is naturally modeled by constructing a binary event sequence (e.g., Wednesday vs. non-Wednesday) and computing the MI between this event sequence and the time series, enabling principled quantification of localized and context-dependent repetition.

4.2.1 Global Seasonality. **Dataset.** We use daily traffic volume data from the Department of Transportation in Minneapolis [15] over four months in 2023 (Figure 3). Visual inspection provides a natural ground truth at this scale: strong weekly seasonality is present in June–July (Figure 3a), while the pattern is substantially weaker in December–January (Figure 3b) due to holiday disruptions.

Baselines. We compare our method with classical seasonality detection approaches: (1) **Auto-correlation function** (ACF) [6]; (2) **ACF+SD**, which applies seasonal decomposition [11] followed by autocorrelation; and (3) **Fourier Transform** (FT) [7].

Results. Table 2 reports seasonality identification results. Both our method and SD+ACF correctly identify weekly seasonality, while ACF and FT fail under noise and irregularities. However, identifying a seasonal period alone is insufficient in practice. A useful method should also reflect *how strongly* seasonality manifests and how it degrades under disruption.

Although SD+ACF detects seasonality, it produces too high values close to 1, considering large offsets from the ideal seasonality



(a) Traffic in Jun. and Jul. (b) Traffic in Dec. and Jan.

Figure 3: Daily traffic volume in Minneapolis (blue) with weekly periodic components identified via harmonic regression (orange), which serve as a manually constructed reference for seasonal structure. Weekly seasonality is clearly presented in June–July, but disrupted in December–January due to holiday effects.

observed in Figure 3a and Figure 3b. In contrast, our proposed measurement yields an MI of 0.6501 for June–July, capturing strong but imperfect seasonality, and a lower MI of 0.5664 for December–January, reflecting stronger disruption. This shows that our method provides both detection and interpretable quantification of the strength of seasonality.

Table 2: Results of seasonality identification and interpretation. For FT, the dominant frequency indicates the seasonal period. For other methods, we report the maximum measure among candidate periods, with values >0.5 suggesting a likely seasonal pattern.

Data	Method	Measure	Seasonality
Jun-Jul	ACF	Autocorr=0.4651	No
	SD+ACF	Autocorr=0.9817	7 days
	FT	Top 1 Freq.=0.13	7-8 days
	Ours	**MI=0.6501**	**7 days**
Dec-Jan	ACF	Autocorr=0.2181	No
	SD+ACF	Autocorr=0.9157	7 days
	FT	Top 1 Freq.=0.08	11-12 days
	Ours	**MI=0.5664**	**7 days**

4.2.2 Local Repeated Patterns. **Dataset.** A *point repeated pattern* refers to repetition that emerges only under specific contextual conditions, rather than as a single global periodic structure. In the temperature setting, fine-grained temporal context (e.g., month and day/night) strongly constrains the temperature distribution, producing narrow, predictable value ranges, whereas coarser context yields weaker repetition and a larger temperature range. Classical wavelet-based methods focus on global periodicity and therefore struggle with such conditional, regime-dependent patterns. Our approach instead quantifies repetition through *context-conditioned value distributions*, enabling a unified treatment of global seasonality, local repetition, and non-periodic structure.

We evaluate this setting using air temperature data collected at 12-hour intervals over one year [15]. The data are throughout 2023 (Figure 4), resulting in 730 timestamps. Each timestamp corresponds to either the maximum daytime temperature (6 am–6 pm) or the minimum nighttime temperature (6 pm–6 am), reflecting the diurnal temperature range. Using the same timestamps, we construct three temporal event sequences with increasing contextual granularity:

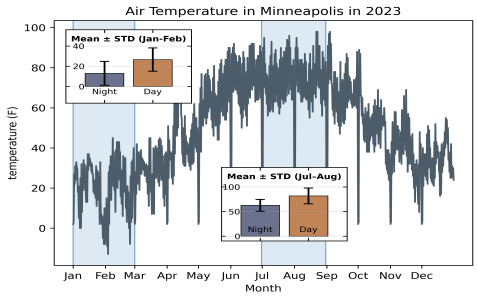


Figure 4: Air temperature in Minneapolis during 2023, sampled at 12-hour intervals (daytime max, nighttime min). Shaded blue regions mark two-month intervals (Jan–Feb and Jul–Aug) within which daytime and nighttime temperatures form stable, narrow ranges (mean, ±STD), revealing strong local repeated patterns. These patterns change across intervals, indicating that repetition is context-dependent rather than globally periodic.

(1) **DN**, labeling each timestep as {Daytime, Night}; (2) **TwoMon**, labeling each timestep by two-month calendar blocks (Jan–Feb, …, Nov–Dec); and (3) **DNTwoMon**, labeling each timestep by the joint context of day/night and two-month block. For each construction, we compute the MI between the temperature series and the corresponding event sequence, measuring how much contextual information reduces uncertainty in the observed values.

Results. Table 3 reports MI estimates between the temperature series and each event sequence. A valid measure of local repeated patterns should exhibit monotonic behavior as contextual information increases. Moving from DN to TwoMon to DNTwoMon introduces progressively richer temporal context that increasingly constrains the temperature distribution (e.g., January nighttime versus July daytime). Our normalized MI increases consistently ($0.6124 \rightarrow 0.8932 \rightarrow 0.9596$), matching both physical intuition and the expected reduction in conditional entropy.

In contrast, the Ross produces negative values that violate the non-negativity of MI, making it unsuitable for quantifying pattern strength in this setting. The Mixture yields positive but unbounded values whose scale grows with event granularity, making it unclear if the contextual event sequence is strong enough without knowing a max pivot. As a result, Mixture does not provide an interpretable reference in different contextual constructions.

Table 3: MI estimates for local repeated patterns under different temporal event constructions. DN distinguishes daytime and nighttime; TwoMon partitions the year into two-month intervals; DNTwoMon combines both. A valid measure should increase as the temporal context better aligns with temperature variation. Our normalized MI provides a bounded, interpretable scale that clearly reflects increasing pattern strength, whereas other methods either produce invalid or unbounded scores that are difficult to interpret.

Method	Range	DN	TwoMon	DNTwoMon
Ross	$[0, \infty]$	-1.8418	-0.7864	-0.2332
Mixture	$[0, \infty]$	0.3749	1.2807	1.5544
Ours	**[0, 1]**	**0.6124**	**0.8932**	**0.9596**

4.3 Discrete Covariate Identification

This task studies discrete covariate identification for time-series forecasting with continuous-valued targets. Given a continuous target time series and a set of candidate discrete covariates, we use the proposed estimator to quantify the dependence between the target and each covariate from historical data. Covariates are then ranked by their estimated MI, and the most informative ones are selected as input to downstream forecasting models. Normalized Discounted Cumulative Gain (NDCG) is used to assess the quality of covariate rankings produced by different MI estimators. We expect that the larger the MI between the covariate and the target time series, the higher the forecast accuracy. It suggests that the covariate with higher MI could provide more useful information. Ideally, the rank of the estimated MI could serve as a proxy to forecast the accuracy rank. This experiment evaluates MI estimators through their *downstream impact* on forecast performance, as ground-truth MI values are unavailable for real-world datasets.

Dataset. We evaluate this task using the **Rossmann Store Sales dataset**[1] and the **M5 Forecasting dataset**[2]. The Rossmann dataset contains store-level daily sales time series with associated discrete covariates such as promotions and holidays (Figure 5). The M5 dataset consists of daily unit sales for Walmart products, along with categorical attributes such as product, department, and calendar-related variables. In both datasets, the goal is to identify informative discrete covariates for forecasting continuous daily sales values. We randomly sample 100 time series from each dataset. Dataset details are provided in the Appendix F.

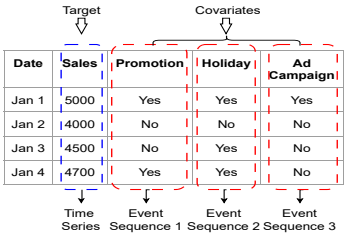


Figure 5: Illustration of discrete covariate identification for forecasting. The sales time series (blue) is the prediction target, and each discrete covariate (red) is modeled as a temporal event sequence.

Baselines & Forecasting Methods. We compare our proposed estimator with **Ross** [29] and **Mixture** [14] for MI estimation and covariate ranking. The selected covariates are then used as inputs to three representative forecasting models: CatBoostRegressor [28], DeepAR [30], FMTimeSeries [12], and Chronos-2 [2]. We also evaluate an ablated variant of our method without event clustering, denoted as **Ours**, and a full variant with clustering, denoted as **Ours-Cluster**. This comparison isolates the effect of grouping correlated discrete values into latent categories during covariate selection.

Results. Table 4 reports forecasting performance measured by mean NDCG over 100 sampled time series and the number of times that each method achieves the highest NDCG. Since higher MI indicates stronger and more reliable dependence between covariates and the target series, more accurate MI estimation is expected to yield better covariate selection and better forecasting performance.

Across both datasets and all models, our proposed method consistently outperforms Ross and Mixture in terms of mean NDCG and the number of winning rounds. This improvement is observed consistently across tree-based (CatBoost), probabilistic (DeepAR), and foundation-model-based (FMTimeSeries and Chronos-2) approaches, indicating that the benefit arises from improved covariate selection rather than from model-specific effects. Moreover, **Ours-Cluster** achieves the highest mean NDCG and the most winning rounds in nearly all settings. This result highlights the importance of modeling correlations and redundancies among discrete covariates by grouping them into latent categories, further improving the stability and effectiveness of MI-based covariate selection.

Table 4: Results of covariate selection for forecasting. Higher NDCG indicates better MI estimation. Values in (parentheses) denote the number of winning rounds (highest NDCG) out of 100 samples.

Model	Benchmark	Ross	Mixure	Ours	**Ours-Cluster**
CatBoostRegressor	Rossmann	0.85 (4)	0.75 (5)	0.94 (29)	**0.95** (62)
	M5	0.75 (17)	0.62 (5)	0.81 (15)	**0.86** (63)
DeepAR	Rossmann	0.78 (2)	0.76 (10)	0.90 (23)	**0.92** (65)
	M5	0.71 (17)	0.54 (4)	0.76 (3)	**0.82** (76)
FMTimeSeries	Rossmann	0.82 (0)	0.92 (28)	0.93 (0)	**0.96** (72)
	M5	0.88 (27)	0.74 (18)	0.89 (8)	**0.92** (47)
Chronos-2	Rossmann	0.72 (0)	0.77 (2)	0.92 (1)	**0.95** (97)
	M5	0.70 (9)	0.59 (7)	0.78 (11)	**0.83** (73)

4.4 Continuous Feature Selection

We further evaluate our estimator on continuous feature selection for mixed-type tabular data, where input features are continuous variables and the target label is discrete. Given a continuous feature, we treat its values across all instances as samples from a continuous random variable, and the corresponding class labels as a discrete random variable. This setting corresponds to the canonical use case of **Ross** [29] and **Mixture** [14].

Our proposed estimator directly computes the MI between a continuous feature and a discrete label without distributional assumptions or ad hoc discretization. Although originally motivated by temporal data, our formulation naturally generalizes to tabular settings by aggregating entropy with permutation-invariant operators (i.e., summation and integration), making feature relevance independent of instance ordering. Conceptually, this corresponds to modeling feature values as samples drawn from an underlying stochastic process, an assumption that is equally valid for both time series and i.i.d. tabular data.

Based on this formulation, we define the feature importance (FI) between a continuous feature variable F and a discrete label variable L as $\mathrm{FI}(L,F) = \sum_{l\in\mathcal{L}} \int_{f\in\mathcal{F}} P(l,f) \log\left(\frac{P(l,f)}{P(l)P(f)}\right) df$.

Dataset. We evaluate feature selection performance using datasets from a widely adopted benchmark [22], which includes multiple classification datasets with continuous features and discrete class labels across diverse domains such as image analysis, biological

[1] https://www.kaggle.com/competitions/rossmann-store-sales
[2] https://www.kaggle.com/competitions/m5-forecasting-accuracy

data, and spoken letter recognition. Additional dataset details are provided in the Appendix G.

Baselines & Classifiers. We compare our estimator with **Ross** [29] and **Mixture** [14] for MI-based feature selection. For all methods, the top-k selected features are used as inputs to three standard classifiers: Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR).

Results. We evaluate feature relevance through downstream classification performance. Detailed classification accuracies using the top k selected features are reported in Detailed classification results for individual datasets and feature budgets are reported in Tables 8 and 10 in the Appendix. Table 5 summarizes performance over a total of 120 evaluation cases, spanning all datasets, classifiers, and feature budgets. In each case, a *win* indicates that features selected by our method yield higher classification accuracy than those selected by a baseline method for the same classifier and dataset; *tie* and *loss* are defined analogously. Across all settings, this aggregated comparison highlights the consistency of our method relative to existing approaches. Our proposed measurement consistently outperforms existing MI-based methods, indicating more accurate quantification of dependence between continuous features and discrete labels. Notably, the optional clustering strategy further improves performance by grouping redundant or highly overlapping feature-value distributions into latent categories, reducing estimation noise and enhancing feature ranking stability.

Table 5: The count of wins, ties, and loses across 120 settings.

Method Comparison	Win	Tie	Lose
Ours VS. Ross	89	14	17
Ours VS. Mixtures	93	15	12
W/ cluster VS. W/O cluster	77	20	23

5 Discussion

We introduced a nonparametric mutual-information–based measurement for directly quantifying interactions between heterogeneous temporal data, specifically continuous time series and discrete event sequences. Unlike prior approaches, the proposed estimator operates natively on mixed data types without requiring discretization, distributional assumptions, learned models, or ad hoc data transformations. This design allows the measurement to remain simple, interpretable, and theoretically grounded while addressing practical issues such as finite precision, repeated values, and correlated events that commonly arise in real-world temporal data. Across a diverse set of fundamental analysis tasks, our experiments demonstrate that the proposed measurement consistently achieves the intended analytical objectives and often outperforms existing discrete–continuous MI estimators and classical correlation-based methods. These results suggest that the estimator can serve as a reliable, task-agnostic measure of dependence between heterogeneous temporal variables, analogous in spirit to Pearson correlation but applicable to mixed data types and nonlinear relationships. Future work includes extending the estimator to better accommodate application-specific noise characteristics and integrating it with learning-based frameworks that can further improve robustness under severe noise or sparsity.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. Specifically, thanks to Junhao Zhu for the insightful discussion. This work has been funded in part by the NIH award R01LM014026.

References

[1] Hirotugu Akaike. 1976. Canonical correlation analysis of time series and the use of an information criterion. In *Mathematics in science and engineering*. Vol. 126. Elsevier, 27–96.
[2] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. 2025. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821* (2025).
[3] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *International conference on machine learning*. PMLR, 531–540.
[4] Miguel AG Belmonte, Gary Koop, and Dimitris Korobilis. 2014. Hierarchical shrinkage in time-varying parameter models. *Journal of Forecasting* 33, 1 (2014), 80–94.
[5] Moïse Blanchard, Adam Quinn Jaffe, and Nikita Zhivotovskiy. 2025. Consistency and Inconsistency in K-Means Clustering. *arXiv preprint arXiv:2507.06226* (2025).
[6] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.
[7] David R Brillinger. 2001. *Time series: data analysis and theory*. SIAM.
[8] Jiawei Chen and Chunhui Zhao. 2024. Addressing spatial-temporal heterogeneity: General mixed time series analysis via latent continuity recovery and alignment. *Advances in Neural Information Processing Systems* 37 (2024), 17910–17946.
[9] Lakshmi P Chitta, Andrei N Zhukov, David Berghmans, Hardi Peter, Susanna Parenti, Sudip Mandal, Regina Aznar Cuadrado, Udo Schühle, Luca Teriaca, Frédéric Auchère, et al. 2023. Picoflare jets power the solar wind emerging from a coronal hole on the Sun. *Science* 381, 6660 (2023), 867–872.
[10] Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics* 16, 1 (1990), 22–29.
[11] Robert B Cleveland, William S Cleveland, Jean E McRae, Irma Terpenning, et al. 1990. STL: A seasonal-trend decomposition. *J. Off. Stat* 6, 1 (1990), 3–73.
[12] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. 2024. A decoder-only foundation model for time-series forecasting. In *Forty-first International Conference on Machine Learning*.
[13] James Dougherty, Ron Kohavi, and Mehran Sahami. 1995. Supervised and unsupervised discretization of continuous features. In *Machine learning proceedings 1995*. Elsevier, 194–202.
[14] Weihao Gao, Sreeram Kannan, Sewoong Oh, and Pramod Viswanath. 2017. Estimating mutual information for discrete-continuous mixtures. *Advances in neural information processing systems* 30 (2017).
[15] Malcolm Grossman, Yuankun Jiao, Haoji Hu, John Hourdos, Yao-Yi Chiang, et al. 2024. *Performance Evaluation of Different Detection Technologies for Signalized Intersections in Minnesota*. Technical Report. Minnesota. Department of Transportation.
[16] Isabelle Guyon and Andre Elisseeff. 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research* (2003), 1157–1182.
[17] Edwin T Jaynes. 1968. Prior probabilities. *IEEE Transactions on systems science and cybernetics* 4, 3 (1968), 227–241.
[18] Lyudmyla F Kozachenko and Nikolai N Leonenko. 1987. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii* 23, 2 (1987), 9–16.
[19] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. 2004. Estimating mutual information. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 69, 6 (2004), 066138.
[20] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. 2013. *Statistical Physics: Volume 5*. Vol. 5. Elsevier.
[21] Oliver F Lange and Helmut Grubmüller. 2006. Generalized correlation for biomolecular dynamics. *Proteins: Structure, Function, and Bioinformatics* 62, 4 (2006), 1053–1061.
[22] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. 2018. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)* 50, 6 (2018), 94.
[23] Huan Liu, Farhad Hussain, Chew Lim Tan, and Manoranjan Dash. 2002. Discretization: An enabling technique. *Data mining and knowledge discovery* 6 (2002), 393–423.
[24] Chen Luo, Jian-Guang Lou, Qingwei Lin, Qiang Fu, Rui Ding, Dongmei Zhang, and Zhe Wang. 2014. Correlating events with time series for incident diagnosis. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1583–1592.
[25] Jessica Magallanes, Tony Stone, Paul D Morris, Suzanne Mason, Steven Wood, and Maria-Cruz Villa-Uriol. 2021. Sequen-c: A multilevel overview of temporal event sequences. *IEEE Transactions on Visualization and Computer Graphics* 28, 1

(2021), 901–911.
[26] Daniel Nagel, Georg Diez, and Gerhard Stock. 2024. Accurate estimation of the normalized mutual information of multidimensional data. *arXiv preprint arXiv:2405.04980* (2024).
[27] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[28] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. CatBoost: unbiased boosting with categorical features. *Advances in neural information processing systems* 31 (2018).
[29] Brian C Ross. 2014. Mutual information between discrete and continuous data sets. *PloS one* 9, 2 (2014), e87357.
[30] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. 2020. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting* 36, 3 (2020), 1181–1191.
[31] Janett Walters-Williams and Yan Li. 2009. Estimation of mutual information: A survey. In *Rough Sets and Knowledge Technology: 4th International Conference, RSKT 2009, Gold Coast, Australia, July 14-16, 2009. Proceedings 4.* Springer, 389–396.

A Proof of the Measurement's Non-negativity

The classical Shannon mutual information (i.e., the mutual information between two discrete variables) is non-negative, and the mutual information between two continuous variables is also proved to be non-negative [26]. We provide the corresponding proof for the mutual information between a continuous variable and a discrete variable. This is the foundation of the forced non-negativity operation in our estimated mutual information.

Theorem A.1. *The mutual information between a continuous variable and a discrete variable is non-negative.*

Proof:

As $H(X) \geq 0$, $\hat{H}(Y) \geq 0$ and $\min(H(X), \hat{H}(Y)) \geq \hat{H}(X,Y) \geq 0$, $H(X) + \hat{H}(Y) - \hat{H}(X,Y) \geq 0$,

$$\begin{aligned}
\mathrm{I}(X,Y) &= H(X) + \hat{H}(Y) - \hat{H}(X,Y) && (11)\\
&= -\sum_{x\in\mathcal{X}} P(x)\log P(x) - \int_{\mathcal{Y}} P(y)\log\frac{P(y)}{m(y)}\,\mathrm{d}\,y && (12)\\
&+ \sum_{x\in\mathcal{X}}\int_{\mathcal{Y}} P(x,y)\log\frac{P(x,y)}{m(y)}\,\mathrm{d}\,y && (13)\\
&= -\sum_{x\in\mathcal{X}} P(x)\log P(x) - \int_{\mathcal{Y}} P(y)\log P(y)\,\mathrm{d}\,y && (14)\\
&+ \int_{\mathcal{Y}} P(y)\log m(y)\,\mathrm{d}\,y + \sum_{x\in\mathcal{X}}\int_{\mathcal{Y}} P(x,y)\log P(x,y)\,\mathrm{d}\,y && (15)\\
&- \sum_{x\in\mathcal{X}}\int_{\mathcal{Y}} P(x,y)\log m(y)\,\mathrm{d}\,y && (16)\\
&= H(X) + H(Y) - H(X,Y) \geq 0 && (17)
\end{aligned}$$

B Theoretical Analysis Discussion

Existing theoretical analysis on the estimator, like consistency analysis and finite-sample bounds, usually relies on the assumption of a large sample size. However, real-world time series have limited length, e.g., tens to thousands. Those bounds, in the worst case, are loose and lack practical guidance. In addition, consistency analysis for clustering is an open challenge [5]. Therefore, we rely on empirical evaluation using real-world data rather than error bound analysis. Therefore, we rely on diverse empirical evaluations from real-world data.

C Mutual Information Estimation Algorithm

Our estimation algorithm could be formalized as Algorithm 1. The algorithm to estimate the continuous entropy is Algorithm 2.

Algorithm 1 Normalized mutual information calculation
Input: A temporal event sequence X and a time series Y.

Get the symbol set X_{set} from X
Assign all values aligning with a specific temporal event e over time by group[e] = [list contains continuous values]
if correlation among events is observed **then**
    Use $hierarchical\_clustering(group, num\_clusters)$ to remap events to latent cluster events and generate X_{new}
    $X = X_{new}$
end if
Partition (X, Y) based on repeated values into (X_A, Y_A) and (X_B, Y_B).
$a = \frac{\|(X_A,Y_A)\|}{\|(X,Y)\|}$, $b = \frac{\|(X_B,Y_B)\|}{\|(X,Y)\|}$
$\mathrm{I}_A = continuous\_entropy(Y_A)$
for each event symbol e in X_{set} **do**
    $\mathrm{I}_A -= \frac{\mathrm{len}(group[e])}{\mathrm{len}(Y)} continuous\_entropy(group_A[e])$
end for
Calculate I_B with Eq. 1.
$\mathrm{I} = a \cdot \mathrm{I}_A + b \cdot \mathrm{I}_B + a \cdot ln(a) + b \cdot ln(b)$
Clip the negative estimation $\mathrm{I} = \max(\mathrm{I}, 0)$
Calculate $\mathrm{NMI}(X, Y)$ (with Eq. 7)
return $\mathrm{NMI}(X, Y)$

Algorithm 2 continuous_entropy
Input: A time series as variable Y

Sort all values in Y and get $sorted\_Y$
Get the difference between all the consecutive pairs in $sorted\_Y$
Initialize an array ρ, each element corresponds to a value v_i in $sorted\_Y$
for each $v_i \in sorted\_Y$ **do**
    Calculate the nearest neighbor distance:
    $\rho_i = \min(v_i - v_{i-1}, v_{i+1} - v_i)$
end for
$sum \leftarrow 0$
for each ρ_i **do**
    $sum = sum + \log(\rho_i)$
end for
$mean\_\rho = \frac{sum}{|\rho|}$
$H(Y) = mean\_\rho + \ln(c) + \ln(\gamma) + \ln(|\rho| - 1)$
return $H(Y)$

D Time-Delayed Mutual Information in Different Settings

The synthetic data distribution is visualized in Fig. 6.

Table 1 shows the results when the decimal place number is 3 and the time series length is 10,000. To better demonstrate the robust performance of the proposed method, we compare the proposed measurement with the Ross and Mixture methods across different digit precision settings (Table 6) and sample numbers (Table 7).

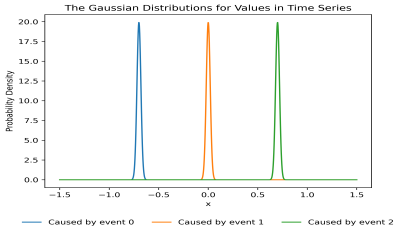


Figure 6: The Gaussian distributions for time series.

Table 6: The comparison with 10,000 time step samples under different decimal places precision. Only report the mean square error (rounded to 5 decimal places) for simplicity.

	Digit Num. = 2			Digit Num. = 3			Digit Num. = 4		
τ	Ro. MSE	Mi. MSE	Pr. MSE	Ro. MSE	Mi. MSE	Pr. MSE	Ro. MSE	Mi. MSE	Pr. MSE
0	29.993	0	0	10.254	4.00E-05	1.00E-05	1.1882	0.1270	0.0468
1	30.000	0	0	10.279	0.00013	5.00E-05	1.1649	0.1374	0.0482
2	30.002	0	0	10.262	6.00E-05	1.00E-05	1.1892	0.1373	0.0474
3	29.990	0	0	10.247	3.00E-05	0	1.1834	0.1373	0.0488
4	29.992	0	0	10.241	4.00E-05	0	1.2047	0.1364	0.0450
5	**43.153**	**0**	**0**	**18.022**	**9.00E-05**	**0**	3.2257	0.0293	0
6	30.003	0	0	10.267	0.00012	2.00E-05	1.1939	0.1443	0.0456
7	29.994	0	0	10.263	5.00E-05	0	1.1689	0.1401	0.0477
8	30.000	0	0	10.272	7.00E-05	1.00E-05	1.2040	0.1362	0.0445
9	29.993	0	0	10.263	6.00E-05	1.00E-05	1.2072	0.1313	0.045

Table 7: The comparison with time series in two decimal places under sample time step numbers. Only report the mean square error (rounded to 5 decimal places) for simplicity.

	Time Step Num. = 100			Time Step Num. = 1000			Time Step Num. = 10000		
	Ro. MSE	Mi. MSE	Pr. MSE	Ro. MSE	Mi. MSE	Pr. MSE	Ro. MSE	Mi. MSE	Pr. MSE
0	1.152	0.1319	0.0153	1.182	0.1741	0.0556	29.993	0	0
1	1.234	0.0844	0.04248	1.156	0.1691	0.0679	30.000	0	0
2	1.265	0.1480	0.00169	1.159	0.1857	0.0756	30.002	0	0
3	1.123	0.1410	0.01668	1.198	0.1628	0.0600	29.990	0	0
4	1.106	0.1871	0.01358	1.146	0.1836	0.0843	29.992	0	0
5	**3.320**	**0.0722**	**0.00047**	**3.317**	**0.0371**	**0**	**43.153**	**0**	**0**
6	1.054	0.1061	0.00472	1.131	0.1768	0.0667	30.003	0	0
7	0.956	0.0622	0.07345	1.108	0.1727	0.0635	29.994	0	0
8	1.179	0.1852	0.02533	1.175	0.1829	0.0553	30.000	0	0
9	0.987	0.2169	0.0116	1.078	0.1692	0.0696	29.993	0	0

Table 6 suggested that the proposed measurement can achieve the best accuracy consistently with different digit numbers. This demonstrates the robustness of our measurement.

Table 7 suggested that the proposed measurement could consistently perform well even with a few samples.

E Cut Threshold in Hierarchical Clustering

The classical agglomerative clustering method [27] has a distance threshold parameter defining the linkage distance at which the clustering process stops merging clusters. We only use the clustering strategy in the covariate selection and feature selection tasks. We fix this threshold to 0.9 for all experiments for simplicity. We also illustrate the sensitivity of this parameter in Figure 7 for the discrete covariate selection task in the M5 dataset.

F Setting for Discrete Covariate Selection

For the Rossmann Store Sales dataset, the goal is to forecast the daily sales number time series from Rossmann drug stores with the following discrete covariates:

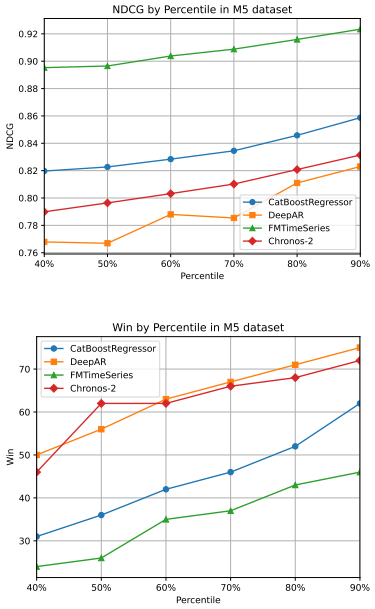


Figure 7: Distance Threshold Sensitivity

- Open - 0 = store is closed, 1 = store is open.
- DayOfWeek - an indicator for the day of a week.
- Promo - indicates if a store is running a promo.
- StateHoliday - indicates a state holiday. Normally, all stores are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None.

For the M5 dataset, the goal is to forecast the daily sales number time series of one retail good with the following discrete covariates:

- Weekday - an indicator for the day of the week.
- Event name 1 - indicates the first event occurs on a specific day. It has various events.
- Event type 1 - indicates the event types of the first event.
- Event name 2 - indicates the second event occurs on a specific day.
- Event type 2 - indicates the event types of the second event.
- Snap CA - indicates whether the stores of CA allow SNAP purchases on the date examined. 1 indicates allowed.
- Snap TX - indicates whether the stores of TX allow SNAP purchases on the date examined. 1 indicates allowed.
- Snap WI - indicates whether the stores of WI allow SNAP purchases on the date examined. 1 indicates allowed.

G Setting and Results for Feature Selection

A toy example of feature selection could be found in Fig. 8.

All the datasets we used are from the widely used benchmark [22], and we selected those with continuous variables as features. The statistics information of the selected datasets is shown in Table 9.

The detailed experimental results compared with Ross using random forest (RF), support vector machine (SVM), and logistic regression (LR) as the base model are shown in Table 8 and 10.

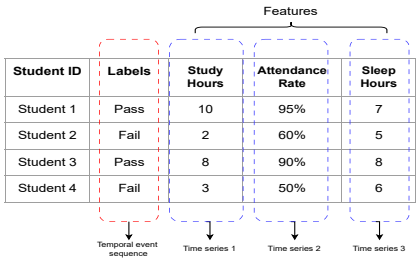


Figure 8: A feature selection example for classifying if a student would fail in the exam. Each feature dimension (blue box) is treated as a time series. The label dimension (red box) is treated as the temporal event sequence. Estimation is between each pair of labels and features (temporal event sequence-time series). The MI can reflect the feature importance.

Table 8: Detailed classification accuracy of random forest (RF), support vector machine (SVM), and logistic regression (LR) with top k selected features based on the existing mutual information method and the proposed method.

Dataset	Method	k=20	k=40	k=60	k=80	k=100
COIL20	RF Ross:	0.9201	0.9514	0.9688	0.9722	0.9757
	RF Mixture:	0.9306	0.9618	0.9722	0.9792	0.9757
	RF NoClu.:	0.941	0.9757	0.9896	0.9931	0.9896
	RF proposed:	0.9965	1.0	0.9965	1.0	1.0
	SVM Ross:	0.7535	0.8576	0.9062	0.9132	0.9271
	SVM Mixture:	0.7326	0.875	0.8958	0.9167	0.9306
	SVM NoClu:	0.8368	0.9306	0.9549	0.9549	0.9722
	SVM proposed:	0.9028	0.9514	0.9722	0.9896	0.9896
	LR Ross:	0.6528	0.7847	0.8125	0.8472	0.8854
	LR Mixture:	0.6562	0.7847	0.8194	0.8368	0.8889
	LR NoClu:	0.7465	0.8403	0.8889	0.8993	0.9062
	LR proposed:	0.8056	0.8889	0.934	0.9479	0.9618
ORL	RF Ross:	0.6375	0.75	0.7875	0.8125	0.875
	RF Mixture:	0.575	0.7	0.775	0.8125	0.8375
	RF NoClu.:	0.75	0.8125	0.85	0.9	0.9
	RF proposed:	0.7125	0.8625	0.9375	0.9	0.9375
	SVM Ross:	0.6875	0.725	0.8125	0.9	0.925
	SVM Mixture:	0.6375	0.8125	0.875	0.8375	0.85
	SVM NoClu:	0.7	0.85	0.8375	0.9	0.9125
	SVM proposed:	0.7375	0.8875	0.925	0.925	0.9375
	LR Ross:	0.5125	0.6625	0.725	0.775	0.8
	LR Mixture:	0.4875	0.6875	0.75	0.7625	0.825
	LR NoClu:	0.625	0.7375	0.8	0.8125	0.8375
	LR proposed:	0.675	0.8375	0.9375	0.9375	0.95
Yale	RF Ross:	0.6061	0.697	0.7273	0.7576	0.7273
	RF Mixture:	0.4242	0.6667	0.7576	0.6667	0.697
	RF NoClu.:	0.5152	0.6667	0.6364	0.697	0.7879
	RF proposed:	0.5152	0.6364	0.7273	0.6364	0.6364
	SVM Ross:	0.4848	0.6667	0.6364	0.6364	0.6061
	SVM Mixture:	0.4848	0.5758	0.6061	0.5758	0.5758
	SVM NoClu:	0.4848	0.6667	0.6364	0.5758	0.6061
	SVM proposed:	0.5758	0.5455	0.6061	0.6364	0.6364
	LR Ross:	0.4848	0.6061	0.6364	0.697	0.6364
	LR Mixture:	0.5152	0.5455	0.6364	0.6364	0.6061
	LR NoClu:	0.4242	0.697	0.6061	0.6061	0.6364
	LR proposed:	0.5152	0.6061	0.6364	0.697	0.7576
warpPIE	RF Ross:	0.8571	1.0	1.0	1.0	1.0
	RF Mixture:	0.9286	1.0	1.0	1.0	1.0
	RF NoClu.:	0.9286	1.0	1.0	1.0	1.0
	RF proposed:	0.9048	1.0	1.0	1.0	1.0
	SVM Ross:	0.9286	0.9524	0.9762	0.9762	0.9762
	SVM Mixture:	0.9762	1.0	1.0	1.0	1.0
	SVM NoClu:	0.9762	1.0	1.0	1.0	1.0
	SVM proposed:	1.0	1.0	1.0	1.0	1.0
	LR Ross:	0.9524	1.0	0.9762	0.9524	0.9762
	LR Mixture:	1.0	1.0	1.0	1.0	1.0
	LR NoClu:	0.9762	1.0	1.0	1.0	1.0
	LR proposed:	0.9762	1.0	1.0	1.0	1.0

Table 9: Statistics information of datasets for feature selection.

Dataset	Num. of Instances	Num. of Features	Num. of Classes
COIL20	1440	1024	20
ORL	400	1024	40
Yale	165	1024	15
warpPIE10P	210	2420	10
Isolet	1560	617	26
TOX_171	171	5748	4
USPS	9298	256	10
CLL_SUB_111	111	11340	3

Table 10: Detailed classification accuracy of random forest (RF), support vector machine (SVM), and logistic regression (LR) with top k selected features based on the existing mutual information method and the proposed method.

Dataset	Method	k=20	k=40	k=60	k=80	k=100
Isolet	RF Ross:	0.6122	0.7404	0.7853	0.7981	0.8109
	RF Mixture:	0.6827	0.7404	0.7917	0.8301	0.8173
	RF NoClu.:	0.4199	0.5096	0.5897	0.7276	0.7821
	RF proposed:	0.5801	0.7436	0.8013	0.7981	0.8237
	SVM Ross:	0.6154	0.7179	0.766	0.7917	0.8141
	SVM Mixture:	0.6538	0.7468	0.8013	0.8333	0.8269
	SVM NoClu:	0.4615	0.4904	0.5641	0.734	0.8269
	SVM proposed:	0.5705	0.7724	0.8301	0.8526	0.8494
	LR Ross:	0.5962	0.7212	0.7724	0.8109	0.8397
	LR Mixture:	0.641	0.7628	0.7949	0.8365	0.8558
	LR NoClu:	0.4231	0.4872	0.5321	0.7372	0.8301
	LR proposed:	0.5513	0.7532	0.8301	0.8718	0.8718
TOX	RF Ross:	0.6857	0.8286	0.8571	0.8	0.6857
	RF Mixture:	0.3714	0.4	0.4857	0.4857	0.5714
	RF NoClu.:	0.7429	0.7429	0.7714	0.8	0.7714
	RF proposed:	0.6286	0.8857	0.8	0.7714	0.8286
	SVM Ross:	0.6	0.7143	0.7143	0.6857	0.7143
	SVM Mixture:	0.4286	0.4	0.6	0.5429	0.6286
	SVM NoClu:	0.7714	0.7143	0.8571	0.8	0.8
	SVM proposed:	0.7143	0.8571	0.8286	0.6571	0.7429
	LR Ross:	0.7143	0.7429	0.7429	0.8286	0.8
	LR Mixture:	0.4857	0.4571	0.6571	0.6571	0.6571
	LR NoClu:	0.7429	0.7429	0.8571	0.7714	0.8571
	LR proposed:	0.6571	0.8857	0.8	0.7429	0.8286
USPS	RF Ross:	0.7129	0.7801	0.821	0.9065	0.9199
	RF Mixture:	0.7145	0.7871	0.872	0.8989	0.9226
	RF NoClu.:	0.728	0.7995	0.8796	0.8995	0.9075
	RF proposed:	0.722	0.8882	0.9	0.922	0.9317
	SVM Ross:	0.5565	0.6419	0.743	0.8946	0.9124
	SVM Mixture:	0.5548	0.6414	0.8199	0.871	0.9108
	SVM NoClu:	0.5715	0.6387	0.7806	0.8188	0.8538
	SVM proposed:	0.6	0.8651	0.8968	0.9151	0.9409
	LR Ross:	0.5215	0.6199	0.7145	0.8849	0.9048
	LR Mixture:	0.5339	0.6156	0.7941	0.8554	0.9
	LR NoClu:	0.5441	0.6199	0.7527	0.7898	0.8124
	LR proposed:	0.5796	0.85	0.8823	0.9102	0.9306
CLL_SUB	RF Ross:	0.6522	0.6522	0.6522	0.5652	0.6087
	RF Mixture:	0.4783	0.4348	0.3478	0.3478	0.4348
	RF NoClu.:	0.6522	0.7391	0.6957	0.6522	0.7391
	RF proposed:	0.6522	0.6957	0.6957	0.6522	0.6522
	SVM Ross:	0.5652	0.6522	0.6087	0.6087	0.6957
	SVM Mixture:	0.4348	0.3478	0.2609	0.3478	0.5217
	SVM NoClu:	0.6087	0.6522	0.6087	0.6087	0.6522
	SVM proposed:	0.4783	0.6957	0.7391	0.6087	0.7391
	LR Ross:	0.6522	0.5652	0.4783	0.6087	0.6087
	LR Mixture:	0.4783	0.3913	0.1739	0.3478	0.4783
	LR NoClu:	0.6957	0.6957	0.5217	0.5652	0.6087
	LR proposed:	0.5652	0.6957	0.6957	0.6087	0.6522