

Article

Causal Discovery and Classification Using Lempel–Ziv Complexity

Dhruthi ¹, Nithin Nagaraj ² and Harikrishnan Nellippallil Balakrishnan ^{1,3,*}¹ Department of Computer Science and Information Systems, BITS Pilani K K Birla Goa Campus, Zuarinagar 403726, Goa, India; f20231019@goa.bits-pilani.ac.in² Complex Systems Programme, National Institute of Advanced Studies, Indian Institute of Science Campus, Bengaluru 560012, Karnataka, India; nithin@nias.res.in³ Consciousness Studies Programme, National Institute of Advanced Studies, Indian Institute of Science Campus, Bengaluru 560012, Karnataka, India

* Correspondence: harikrishnannb@goa.bits-pilani.ac.in

Abstract

Inferring causal relationships in the decision-making processes of machine learning models is essential for advancing explainable artificial intelligence. In this work, we propose a novel causality measure and a distance metric derived from Lempel–Ziv (LZ) complexity. We explore how these measures can be integrated into decision tree classifiers by enabling splits based on features that cause the most changes in the target variable. Specifically, we design (i) a causality-based decision tree, where feature selection is driven by the LZ-based causal score; (ii) a distance-based decision tree, using LZ-based distance measure. We compare these models against traditional decision trees constructed using Gini impurity and Shannon entropy as splitting criteria. While all models show comparable classification performance on standard datasets, the causality-based decision tree significantly outperforms all others on the Coupled Auto Regressive (AR) dataset, which is known to exhibit an underlying causal structure. This result highlights the advantage of incorporating causal information in settings where such a structure exists. Furthermore, based on the features selected in the LZ causality-based tree, we define a causal strength score for each input variable, enabling interpretable insights into the most influential causes of the observed outcomes. This makes our approach a promising step toward interpretable and causally grounded decision-making in AI systems.



Academic Editor: Bumba Mukherjee

Received: 6 August 2025

Revised: 13 September 2025

Accepted: 19 September 2025

Published: 10 October 2025

Citation: Dhruthi; Nagaraj, N.; Nellippallil Balakrishnan, H. Causal Discovery and Classification Using Lempel–Ziv Complexity. *Mathematics* **2025**, *13*, 3244. <https://doi.org/10.3390/math13203244>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: causal discovery; Lempel–Ziv complexity; decision trees; explainable AI; causality; information theory; machine learning

MSC: 68

1. Introduction

The increasing integration of machine learning (ML) into high-stakes domains such as healthcare, finance, and autonomous systems has amplified the demand for models that are not only accurate but also transparent, reliable, and trustworthy [1,2]. A significant limitation of many contemporary ML models is their reliance on statistical correlations, which can be brittle and misleading. These models often fail to distinguish between mere association and true causation, making them vulnerable to spurious conclusions and poor generalization when deployed in new environments [3]. Therefore, moving beyond

correlation-based learning toward causality-informed decision-making is a cornerstone for the next generation of Explainable Artificial Intelligence (XAI) [4].

The central research problem lies in effectively inferring causal relationships from observational data. While randomized controlled trials (RCTs) are the gold standard for establishing causality, they are often impractical, unethical, or prohibitively expensive to conduct in many real-world scenarios, such as studying the link between lifestyle factors and chronic diseases [5]. Consequently, there is a critical need for robust computational methods that can uncover causal structures directly from passively collected observational datasets.

Existing methods for causal discovery from observational data offer various approaches, but each comes with its own set of limitations. Classical techniques like Granger Causality (GC) and Transfer Entropy (TE) are primarily designed for time-series data and rely on Norbert Wiener's principle of predictability, where a cause must improve the prediction of its effect [6–8]. However, these methods can be constrained by assumptions of linearity (in the case of GC) or require substantial data to accurately estimate probability distributions (for TE) [9]. A more recent and powerful paradigm for causal inference is rooted in algorithmic information theory, which posits that the simplest explanation for a phenomenon is likely the correct one. This principle, formalized by Kolmogorov complexity, suggests that if X causes Y , the algorithmic description of Y given X should be simpler than the description of X given Y . Methods like ORIGO, based on the Minimum Description Length (MDL) principle, and ERGO have operationalized this idea using lossless compression and complexity estimates as practical proxies for the uncomputable Kolmogorov complexity [10,11].

Lempel–Ziv (LZ) complexity, a computationally feasible upper bound on Kolmogorov complexity, has emerged as a valuable tool in this domain due to its parameter-free nature and its proven utility in diverse fields from bioinformatics to anomaly detection [12,13]. However, recent attempts to formulate an LZ-based causality measure, such as the one proposed by Pranay and Nagaraj (2021), suffer from two fundamental drawbacks [14]. Firstly, their formulation implicitly assumes a complete temporal precedence of one variable over another, making it unsuitable for systems where variables evolve and interact concurrently. Secondly, their metric can yield a non-zero “self-causation” penalty (i.e., the penalty for a variable causing itself is greater than zero), which is counter-intuitive and problematic for applications like feature selection. These unresolved issues highlight the need for a more robust and theoretically sound causality measure.

Our research is motivated by the goal of bridging this gap. We aim to develop a novel causality measure based on LZ complexity, which not only addresses the limitations of prior work but can also be seamlessly integrated into standard machine learning frameworks to build inherently interpretable, causally informed models. The primary objective is to create a decision tree classifier that builds its structure not on correlation-based metrics like Gini impurity, but on the causal influence of features on the target variable.

The primary novelty of this paper lies in the formulation of the Lempel–Ziv Penalty (LZP) causality measure, which uniquely employs a real-time, parallel parsing of sequences. This approach overcomes the key limitations of previous compression-based methods by naturally handling concurrent data and guaranteeing a zero self-causation penalty. Furthermore, the direct application of this measure as a splitting criterion in a decision tree is a novel approach to imbue a classical ML model with causal reasoning capabilities.

The main contributions of this work are explicitly listed below:

1. We introduce the Lempel–Ziv Penalty (LZP), a novel causality measure based on LZ complexity that is robust for both temporal and non-temporal observational data.

2. We propose a new distance metric derived from the symmetric difference of LZ-generated grammars.
3. We present the design and implementation of two new classifiers: an LZ-Causality-based decision tree that selects features based on their causal influence and an LZ-Distance-based decision tree.
4. We introduce a causal feature importance score derived from our causal decision tree, offering a more interpretable alternative to correlation-based importance metrics.
5. Through extensive experiments, we demonstrate that while our models perform comparably on standard benchmarks, the causal decision tree significantly outperforms traditional methods on datasets with known underlying causal structures, validating its ability to leverage causal information for improved accuracy.

2. Related Works

Compression-based causal inference methods leverage algorithmic information theory to infer causal directions by measuring how well one variable compresses another.

ORIGO applies the Minimum Description Length (MDL) principle to binary and categorical data, using decision trees as compressors to encode one variable given another [10]. The causal direction is inferred by comparing the total description lengths: if encoding Y given X is shorter than the inverse, X is considered the cause of Y . While conceptually powerful, its performance can be sensitive to the parameterization of the chosen compressor, and its scope is primarily limited to discrete data.

To address the challenge of more complex, real-world datasets, ERGO extends this compression-based framework to handle multivariate and mixed-type (continuous and categorical) data [11]. Instead of evaluating variable pairs in isolation, ERGO seeks the causal ordering of an entire set of variables that yields the most compact overall description of the data. The framework was, however, primarily designed for real-valued data, limiting its application to datasets dominated by categorical or temporal components.

Compression Complexity Causality (CCC) offers a robust method specifically for time-series analysis [15]. It estimates causality by measuring changes in the dynamical compression complexity of time series and is notably resilient to common issues like irregular sampling and noise. However, its primary weakness lies in its strict adherence to the Wiener-Granger causality framework, which assumes that causes must temporally precede their effects. This makes it highly effective for detecting lagged dependencies but potentially less sensitive to contemporaneous causal effects where an event and its cause occur in the same time step. The versatility of compression-based metrics is further demonstrated by recent efforts to integrate them with established causal frameworks, such as using compressibility to measure dependence specifically within additive noise models [16].

The work most closely related to our own is that in [14], which also leverages Lempel–Ziv complexity for causal discovery. However, as detailed in Section 3.1.3, their formulation has critical limitations, including an implicit assumption of temporal precedence and a non-zero self-causation penalty. These unresolved issues directly motivate our research and the development of a novel measure designed to overcome these specific drawbacks, thereby enabling a more robust and intuitive integration into machine learning models like decision trees. Nevertheless, its application is hindered by several critical flaws, which are elaborated upon in Section 3.1.3. These unresolved issues directly motivate our research and the development of a novel measure designed to overcome these specific drawbacks.

3. Materials and Methods

This section describes the novel causality and distance measures derived from Lempel–Ziv complexity, their integration into decision trees, and the experimental setups used for validation.

3.1. Proposed Causality and Distance Measures

3.1.1. Foundational Idea: Lempel–Ziv Complexity for Causal Inference

The foundational idea behind using Lempel–Ziv complexity for causal inference lies in the connection between predictability and compressibility [17]. Consider two variables, X and Y . If X causes Y , then the patterns in X should make Y more predictable—and hence more compressible—when compared to the reverse scenario.

Formally, if

$$Y = P_{Y|X} + \text{noise} \quad (1)$$

then $P_{Y|X}$ denotes the shortest program (or description) that maps X to Y , and its length, $K(P_{Y|X})$, reflects the Kolmogorov complexity of this description. Similarly, we can consider

$$X = P_{X|Y} + \text{noise} \quad (2)$$

where $K(P_{X|Y})$ measures the complexity of describing X given Y .

If $K(P_{Y|X}) < K(P_{X|Y})$, we infer that X causes Y , because Y can be more concisely described using X than vice versa.

However, since Kolmogorov complexity of an arbitrary string X is undecidable [18], we approximate it using practical measures such as Lempel–Ziv complexity [19], which serves as a surrogate by quantifying how well a sequence can be compressed. The direction that leads to a lower Lempel–Ziv complexity is considered the likely causal direction.

3.1.2. Lempel–Ziv 1976 Complexity

Lempel–Ziv Complexity (LZC) is a widely adopted measure of complexity that is rooted in dictionary encoding. LZC compresses data by identifying recurring substrings and replacing repeated instances with references, thus efficiently capturing the structural information of the sequence [12]. This mechanism sees widespread use in areas such as file compression schemes, signal analysis, and study of various system dynamics [20–22].

The advantages of LZC include its non-parametric nature and versatility across data types without assuming specific probabilistic models, making it useful in fields such as neuroscience, genomics, and causal inference. However, recent work highlights certain limitations such as sensitivity to signal encoding, low efficiency due to sequential processing, and challenges when extending to multivariate or continuous data, prompting ongoing research to improve its robustness and application scope [23].

Let $X = x_1x_2 \dots x_n \in \{0,1\}^n$ be a binary string. The Lempel–Ziv complexity (LZ76) [12], denoted by $\text{LZ}(X)$, is defined as the number of distinct phrases obtained by parsing X from left to right such that each new phrase is the shortest substring that has not appeared previously as a phrase.

We decompose the string as

$$X = s_1s_2 \dots s_k \quad (3)$$

where

- $s_1 = x_1$,
- each s_i is the shortest substring such that $s_i \notin \{s_1, s_2, \dots, s_{i-1}\} \cup$ their suffix concatenations,
- and $\text{LZ}(X) = k$.

Example 1 (Parsing of $X = 110011$). *Parse the string from left to right:*

Step 1: 1	$\Rightarrow s_1 = 1$
Step 2: 10	$\Rightarrow s_2 = 10$
Step 3: 0	$\Rightarrow s_3 = 0$
Step 4: 11	$\Rightarrow s_4 = 11$

Thus, the parsing is $X = \underbrace{1}_{s_1} \underbrace{10}_{s_2} \underbrace{0}_{s_3} \underbrace{11}_{s_4}$ and hence, $\text{LZ}(110011) = 4$. The set of distinct phrases (also known as the LZ grammar of X) is $G(X) = \{1, 10, 0, 11\}$.

3.1.3. Research Gap and Comparison to Prior Work

The work in [14] utilizes Lempel–Ziv complexity for causal inference. However, our proposed formulation and underlying assumptions differ fundamentally, leading to distinct advantages, particularly in scenarios involving the simultaneous evolution of variables.

The causal metric in [14] is defined as follows:

$$\text{LZ-}P_{X \rightarrow Y} = \text{LZ}(X + Y) - \text{LZ}(X) \quad (4)$$

$$\text{LZ-}P_{Y \rightarrow X} = \text{LZ}(X + Y) - \text{LZ}(Y) \quad (5)$$

This formulation inherently assumes a full temporal precedence of one variable over the other, which is a critical limitation for systems where variables evolve and interact concurrently. Applying a method predicated on sequential occurrence to such data can yield spurious causal inferences. Our method directly tackles this by employing a real-time grammar construction that processes both sequences simultaneously. By incrementally building grammars and tracking the overlap, we avoid the problematic assumption of strict temporal separation.

A second, more fundamental issue with their formulation is that the penalty for a variable “causing” itself is not guaranteed to be zero. For instance, for a sequence $X = “10111”$, the self-causation penalty is non-zero: $\text{LZ-}P_{X \rightarrow X} = \text{LZ}(X, X) - \text{LZ}(X) = 5 - 3 = 2$. This non-zero baseline means it is possible for a completely different sequence Z to have a lower causal penalty with respect to X than X has with itself. This is particularly problematic for applications like feature selection in a decision tree, where a perfectly predictive feature (i.e., one that is identical to the target) could be assigned a higher penalty and thus be deemed less important than a noisy or irrelevant feature. In contrast, our method is designed to guarantee that $\text{LZ-}P_{X \rightarrow X} = 0$ for any sequence X , ensuring a stable and logical baseline by quantifying only the additional complexity required to explain Y using the real-time grammar of X .

3.1.4. Model Assumptions and Properties

The proposed causality measure has the following assumptions:

1. Assumption 1: A cause must precede or occur simultaneously with its effect. Retro-causal scenarios are not considered.
2. Assumption 2: There is no confounding variable that causes X and Y , where X and Y are temporal/ non-temporal data. We modify the measure to deal with possible confounders separately in Section 3.4.

Consider two univariate datasets $X = [a, b, c, a, c, b, a, a \dots]$ and $Y = [b, a, a, a, b, c, \dots]$, which may or may not have temporal structure. We want to develop a measure in order to comment on the direction causality. i.e., $X \rightarrow Y$ or $Y \rightarrow X$. We want the measure to have the following properties:

- If $X \rightarrow Y$, then the grammar of X constructed over real time has patterns that better explain Y . The extra penalty incurred by explaining Y using the generated grammar of X , denoted as $LZP_{X \rightarrow Y}$, is less than the penalty incurred by explaining X using the generated grammar of Y , denoted as $LZP_{Y \rightarrow X}$. For this case, $LZP_{X \rightarrow Y} < LZP_{Y \rightarrow X}$. The inequality will be reversed if $Y \rightarrow X$.
- Explaining X using the real time generated grammar of X denoted as $LZP_{X \rightarrow X}$ should give zero penalty. This follows from assumption 1.

3.1.5. Definition of the LZ Penalty (LZP) Measure

Given two symbolic sequences, $x = x_0x_1x_2 \dots x_{m-1}$ and $y = y_0y_1y_2 \dots y_{n-1}$, where each symbol x_k belongs to an alphabet $\mathcal{A}_x$ and each symbol y_k belongs to an alphabet $\mathcal{A}_y$. We define $LZP_{x \rightarrow y}$ as the penalty incurred by explaining y using the real-time grammar of x . The algorithm for calculating the penalty is described in Algorithm 1.

The real-time grammar of a sequence is constructed incrementally as we process the sequence from left to right. At each step, we identify the shortest substring starting from the current position that has not been encountered previously in the sequence. This new substring is then added to the grammar set of that sequence. For a set G , $|G|$ denotes its cardinality, i.e., the number of elements in the set. Also note that for a symbolic sequence x , $x[i : j]$ denotes the inclusive range of symbols from index i to j .

Example 2 (Calculation of $LZP_{x \rightarrow y}$). Consider the two strings $x = "101110"$ and $y = "110111"$. Then $LZP_{x \rightarrow y}$ is calculated according to the following steps:

- Step 0: We consider the two strings x and y , and construct their respective grammar sets G_x and G_y respectively using a process analogous to the Lempel–Ziv algorithm. We monitor the overlap at each stage, which can be thought to represent the extent to which the current and previous values of x influence y . $x \sim 101110$, $y \sim 110111$, $\text{overlap} = 0$, $G_x = \emptyset$, $G_y = \emptyset$.
- Step 1: Since we are inferring the strength of causation from string x to string y , we start with the selection of the smallest substring of x , starting from the first index, that is not already present in its grammar set. Since the grammar set G_x is empty, the substring '1' is selected and added to G_x . $x \sim \underline{1}01110$, $\text{overlap} = 0$, $G_x = \{'1'\}$, $G_y = \emptyset$.
- Step 2: Similarly, for string y , the smallest substring starting from the first index that is not in G_y is '1'. Since this substring '1' is already present in G_x , the **overlap** is incremented. We can interpret this as the '1' present in x inducing its occurrence in y . The substring '1' is then added to G_y . $y \sim \underline{1}10111$, $\text{overlap} = 1$, $G_x = \{'1'\}$, $G_y = \{'1'\}$.
- Step 3: The process is now repeated for x . The starting index of the subsequent substring must immediately follow the terminal index of the preceding substring to ensure no information is lost. Hence, starting from the second position, the smallest substring not in G_x is '0', which is selected and added to the set G_x . $x \sim 1\underline{0}1110$, $\text{overlap} = 1$, $G_x = \{'1', '0'\}$, $G_y = \{'1'\}$.
- Step 4: As in Step 3, we consider y . Starting from the second position, the smallest substring not in G_y is '1'. Since it is already present in G_y , we extend the substring to '10'. The substring '10' is not present in G_y and is hence added to it. Since '10' is also not present in G_x , the overlap is not incremented. $y \sim 1\underline{1}0111$, $\text{overlap} = 1$, $G_x = \{'1', '0'\}$, $G_y = \{'1', '10'\}$.
- Step 5: For x , starting from the third position, the smallest substring is '1', which is already present in G_x . We extend it to '11', which is not in G_x . Thus, '11' is selected and added to G_x . $x \sim 10\underline{1}110$, $\text{overlap} = 1$, $G_x = \{'1', '0', '11'\}$, $G_y = \{'1', '10'\}$.
- Step 6: For y , starting from the fourth position, the smallest substring is '1', which is in G_y . We extend it to '11', which is not in G_y . Thus, '11' is selected and added to G_y . Since '11' is already present in G_x , the overlap is incremented. $y \sim 110\underline{1}11$, $\text{overlap} = 2$, $G_x = \{'1', '0', '11'\}$, $G_y = \{'1', '10', '11'\}$.

- Step 7: For x , starting from the fifth position, the smallest substring not in G_x is '10', which is added to G_x . $x \sim 1011\mathbf{10}$, $\text{overlap} = 2$, $G_x = \{'1', '0', '11', '10'\}$, $G_y = \{'1', '10', '11'\}$.

Since there are no more substrings left in either string that can be added to the respective grammar sets, the process stops here, and $LZP_{x \rightarrow y}$ is found to be $(|G_y| - \text{overlap})$, which is $3 - 2 = 1$. In the same manner (although not explicitly shown here), $LZP_{y \rightarrow x}$ is also found to be 1. If $LZP_{x \rightarrow y} < LZP_{y \rightarrow x}$, we can say that x causes y . Similarly, if $LZP_{y \rightarrow x} < LZP_{x \rightarrow y}$, we conclude that y causes x . In this case, since the penalties are equal ($LZP_{x \rightarrow y} = LZP_{y \rightarrow x} = 1$), it implies x and y can be independent or bi-directional. In our experiments, we assumed that y and x are independent if $LZP_{y \rightarrow x} = LZP_{x \rightarrow y}$.

Algorithm 1 details the process of calculating the causal penalty, $LZP_{x \rightarrow y}$. It takes two symbolic sequences, x and y , as input. The core of the algorithm involves building the Lempel–Ziv grammars for both sequences, G_x and G_y , in a parallel, step-by-step manner. At each iteration, it finds the next unique phrase in x and adds it to G_x . It then finds the next unique phrase in y . A crucial step is checking if this new phrase from y already exists in the grammar of x (G_x). If it does, an 'overlap' counter is incremented. This 'overlap' quantifies how many of the structural patterns in y had already been discovered while parsing x . The final penalty is the total number of phrases in y 's grammar minus this overlap, representing the "new" information in y that could not be explained by the information in x .

Algorithm 1 Calculation of Penalty of a string y of length n given another string x of length m

```

1: Input: String  $y$  of length  $n$ , String  $x$  of length  $m$ .
2: Output: Penalty of using  $x$  to compress  $y$ 
3: Initialize set  $G_x \leftarrow \emptyset$  {Empty grammar set}
4: Initialize set  $G_y \leftarrow \emptyset$  {Empty grammar set}
5: Initialize  $i_x \leftarrow 1$ 
6: Initialize  $i_y \leftarrow 1$ 
7: Initialize  $\text{overlap} \leftarrow 0$ 
8: while  $i_y \leq n$  do
9:   if  $i_x \leq m$  then
10:    Set  $j_x \leftarrow i_x$ 
11:    while  $x[i_x : j_x] \in G_x$  and  $j_x < m$  do
12:      Increase  $j_x$  by 1
13:    end while
14:    Add substring  $x[i_x : j_x]$  to  $G_x$ 
15:  end if
16:  Set  $j_y \leftarrow i_y$ 
17:  while  $y[i_y : j_y] \in G_y$  and  $j_y < n$  do
18:    Increase  $j_y$  by 1
19:  end while
20:  if  $y[i_y : j_y] \in G_x$  then
21:    Increase  $\text{overlap}$  by 1
22:  end if
23:  Add  $y[i_y : j_y]$  to  $G_y$ 
24:  Set  $i_x \leftarrow j_x + 1$ 
25:  Set  $i_y \leftarrow j_y + 1$ 
26: end while
27: return  $|G_y| - \text{overlap}$ 

```

3.1.6. A Distance Metric Based on Lempel–Ziv Complexity

In this section, we define a distance metric derived from the grammar of two symbolic sequences. The grammar construction is based on the Lempel–Ziv algorithm [24]. Given two symbolic sequences, x and y , their grammars G_x and G_y respectively can be encoded

using the Lempel–Ziv Algorithm, as shown in Algorithm 2. The Lempel–Ziv distance between two sets G_x and G_y is

$$d_{LZ}(G_x, G_y) = |G_x \setminus G_y| + |G_y \setminus G_x|, \quad (6)$$

where $|G|$ represents the cardinality of a set G . The proof of the given measure being a distance metric is given in Appendix A.

Algorithm 2 Calculation of Grammar of a String Using Lempel–Ziv Complexity

```

1: Input: String  $x$  of length  $n$ 
2: Output: Grammar of the string,  $G_x$ 
3: Initialize set  $G_x \leftarrow \emptyset$  {Empty grammar set}
4: Initialize  $i \leftarrow 1$ 
5: while  $i \leq n$  do
6:   Set  $j \leftarrow i$ 
7:   while  $x[i : j] \in G_x$  and  $j < n$  do
8:     Increase  $j$  by 1
9:   end while
10:  Add substring  $x[i : j]$  to  $G_x$ 
11:  Set  $i \leftarrow j + 1$ 
12: end while
13: return  $G_x$ 

```

Algorithm 2 describes the standard Lempel–Ziv 1976 (LZ76) parsing procedure to generate the grammar of a single string x . It iterates through the string from left to right, identifying the shortest substring that has not appeared as a phrase before. Each new, unique phrase is added to the grammar set, G_x . This process continues until the entire string is parsed. The resulting set, G_x is used as the basis for the Lempel–Ziv distance metric defined in Equation (4).

3.2. Applicability in Decision Trees

The decision tree was chosen as the primary framework for this research due to its interpretability. Its hierarchical structure serves as the most direct and transparent vehicle for demonstrating and validating the behavior of a novel splitting criterion.

We utilize the proposed causality and distance measures as splitting criteria in decision trees, yielding (a) an LZ Causal Measure-based decision tree; (b) an LZ distance metric-based decision tree.

3.2.1. Utilization as a Splitting Criterion

To split a binary tree at a given node, a feature and threshold are selected based on minimizing the chosen LZ-based measure. Each feature and target value is transformed into symbolic sequences: elements in the feature column are represented as 0, if below the threshold, and 1, if above. In the target symbolic sequence t , a given label l is represented as 1 and all other labels as 0.

For the causal decision tree, the feature and threshold selection are performed as per the following equation:

$$(f^*, \tau^*) = \arg \min_{f, t, l} (LZP_{f \rightarrow t}) \quad (7)$$

where f^* is the best feature, τ^* is the best threshold, f is a candidate feature, t is the target variable converted to a symbolic sequence based on label l , and $LZP_{f \rightarrow t}$ is the causal penalty.

For the distance-based decision tree, the selection is based on the following:

$$(f^*, \tau^*) = \arg \min_{f, t, l} (d_{LZ}(G_f, G_t)) \quad (8)$$

where $d_{LZ}(G_f, G_t)$ is the Lempel–Ziv-based distance between the grammar of the feature symbolic sequence f and the target symbolic sequence t .

3.2.2. Causal Strength of a Feature on a Target

We propose a method of ranking the strength of causation of different attributes on a particular label based on the causal decision tree. Consider a decision tree with m nodes, and d_i denotes the depth of the i th node. The causal strength s_j for a feature j to the target is given by the following formulation:

$$s_j = \sum_{i=1}^m a_{ij} \cdot 2^{-d_i}, \quad (9)$$

$$\text{where } a_{ij} = \begin{cases} 1 & \text{if } j \text{ is the splitting feature at the } i\text{th node,} \\ 0 & \text{otherwise} \end{cases}.$$

These scores can then be normalized to provide a relative ranking of causal influence.

3.3. Experimental Setup

3.3.1. Simple Causation in AR Processes of Various Orders

Data was generated from a coupled auto regressive process of order $p = 1, 5, 20, 100$. The governing equations for timeseries $X(t)$ and $Y(t)$ are

$$X(t) = aX(t-1) + \eta Y(t-p) + \epsilon_{X,t}, \quad (10)$$

$$Y(t) = bY(t-1) + \epsilon_{Y,t}. \quad (11)$$

The parameters were set to be $a = 0.9$, $b = 0.9$, and the coupling coefficient (η) was varied from 0 to 1 with a step size of 0.1. Noise terms $\epsilon_{X,t} = \nu e_1$ and $\epsilon_{Y,t} = \nu e_2$ where e_1 and e_2 are sampled from the standard normal distribution and noise intensity $\nu = 0.03$. The process was simulated for $t = 2500$ time steps, with the first 500 transients discarded. This procedure was repeated for $n = 1000$ trials for each value of η .

3.3.2. Coupled Logistic Map

The master-slave system of 1D Logistic maps is governed by

$$Y(t) = L_1(Y(t-1)), \quad (12)$$

$$X(t) = (1 - \eta)L_2(X(t-1)) + \eta Y(t-1). \quad (13)$$

The coupling coefficient η is varied from 0 to 0.9. $L_1(t) = A_1 L_1(t-1)(1 - L_1(t-1))$ and $L_2(t) = A_2 L_2(t-1)(1 - L_2(t-1))$, where $A_1 = 4$ and $A_2 = 3.82$. For each η , 1000 data instances of length 2000 were generated, after removing the initial 500 transients.

3.3.3. Deciphering Causation in the Three Variable Case

We apply a modified *LZP* algorithm to three-variable causal structures. To find causation from X to Y conditioned on Z , denoted $LZP(X \rightarrow Y|Z)$, we first build the grammar of the conditioning variable Z . Then, we proceed with the normal *LZP* algorithm for X and Y , with the caveat that a new phrase from X is only added to its grammar if it is not already present in the grammar of Z . The system is modeled by

$$X(t) = aX(t-1) + bY(t-1) + cZ(t-1) + v\epsilon_{X,t}, \quad (14)$$

$$Y(t) = dY(t-1) + eZ(t-1) + fX(t-1) + v\epsilon_{Y,t}, \quad (15)$$

$$Z(t) = gZ(t-1) + hX(t-1) + iY(t-1) + v\epsilon_{Z,t}. \quad (16)$$

By varying parameters $a \dots i$, we simulate different causal structures. Strings of length 2500 were used (first 1000 transients removed), and results were averaged over 200 trials.

3.3.4. Datasets for Classification

We used several datasets from the UCI Machine Learning Repository [25] and scikit-learn [26] for evaluating the decision tree classifiers: Iris [27], Breast Cancer Wisconsin [28], Congressional Voting Records [29], Car Evaluation [30], KRKPA7 [31], Mushroom [32], Heart Disease [33], and Thyroid [34]. Some of the aforementioned datasets, including Breast Cancer Wisconsin, Congressional Voting Records, Car Evaluation, and KRKPA7, were previously used to evaluate the effectiveness of the Causal Decision Tree proposed by [35]. Motivated by this, we employ these and other related datasets in our study.

Additionally, a synthetic AR Dataset was generated with a known causal structure to specifically test the causal decision tree. The selection of the other public datasets was intended to provide a diverse set of benchmark challenges that vary in sample size, number of features, number of classes, data types (numeric vs. categorical), and class balance. This allows for a robust evaluation of the general-purpose classification performance of the proposed models against established methods. Details on the datasets, including train–test splits, are provided in Table 1.

Table 1. Train–test split information for each dataset. An 80%–20% train–test ratio was utilized. The Imbalance Ratio is the ratio of the number of training samples in the majority class to the minority class.

Dataset	Classes	Features	Training Samples/Class	Testing Samples/Class	Imbalance Ratio
Iris	3	4	(39,37,44)	(11,13,6)	1.189
Breast Cancer Wisconsin	2	30	(286,169)	(71,43)	1.692
Congressional Voting Records	2	16	(101, 84)	(23,14)	1.202
Car Evaluation	4	6	(307, 55, 968, 52)	(77, 14, 242, 13)	18.615
KRKPA7	2	35	(1227, 1329)	(300, 340)	1.083
Mushroom	2	22	(2783, 1732)	(705, 424)	1.607
Heart Disease	5	13	(124, 45, 30, 28, 10)	(36, 9, 5, 7, 3)	12.4
Thyroid	3	21	(93, 191, 3488)	(73, 177, 3178)	37.505
AR Dataset	2	1	(113, 127)	(25, 35)	1.123

3.4. Deciphering Causation in the Three Variable Case

We apply the proposed *LZP* algorithm to three different variable causal structures. We find causation from A to B, with C as the conditioned variable; B to C, with A as the conditioned variable; C to A, with B as the conditioned variable. We also find causation in the opposite direction in each case with the same conditioned variable and find the difference in penalties. The sign indicates the direction, and the magnitude indicates the strength of causation. We decide if causation detected is true or not based on its relative strength.

3.4.1. Algorithm for $LZP(X, Y|Z)$

We use the LZP algorithm with a minor modification. Suppose we are finding causation from X to Y by conditioning it against Z . Let G_x , G_y , and G_z be their respective grammar sets. Before we find the shortest possible substring in X not in G_x , we first find the shortest substring in Z not present in G_z and add it to its grammar G_z . Now, we follow the same algorithm as before, with the caveat that a substring from X is added to G_x only if it is not present in G_z , otherwise it is discarded.

3.4.2. Procedure

The system is modeled by the following set of equations:

$$X(t) = aX(t-1) + bY(t-1) + cZ(t-1) + v\epsilon_{X,t}, \quad (17)$$

$$Y(t) = dY(t-1) + eZ(t-1) + fX(t-1) + v\epsilon_{Y,t}, \quad (18)$$

$$Z(t) = gZ(t-1) + hX(t-1) + iY(t-1) + v\epsilon_{Z,t}. \quad (19)$$

By changing the values for the parameters $a...i$, we obtain different causal structures. The errors are randomly sampled from a normal distribution and $v = 0.01$. Each string is taken to be of length 2500, with the first 1000 transients filtered out. The experiment is performed for 200 trials, and the average values of causal strength are found each time. This procedure is repeated for 20 trials to guarantee consistency.

4. Results

This section presents the results from validating the causality measure and the performance of the LZ-based decision trees.

4.1. Causality Measure Validation

The proposed LZP measure was tested on synthetic data with known causal links.

4.1.1. Coupled AR Processes

Figure 1 shows the average $LZP_{X \rightarrow Y}$ and $LZP_{Y \rightarrow X}$ for coupled AR processes of different orders. Across all orders ($p = 1, 5, 20, 100$), as the coupling coefficient η increases, the penalty for the true causal direction ($Y \rightarrow X$) becomes significantly smaller than for the anti-causal direction ($X \rightarrow Y$). This demonstrates that the proposed measure correctly and robustly identifies the direction of causality.

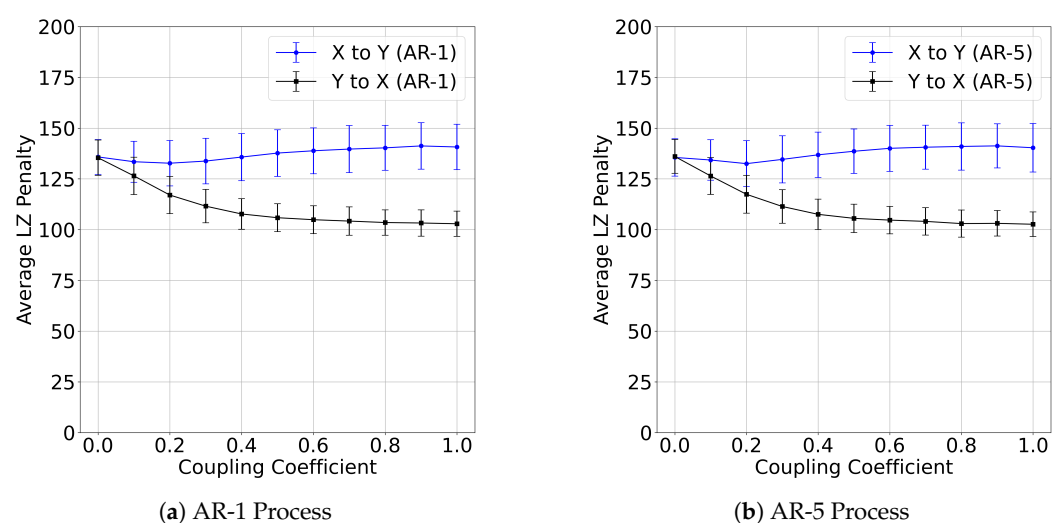


Figure 1. Cont.

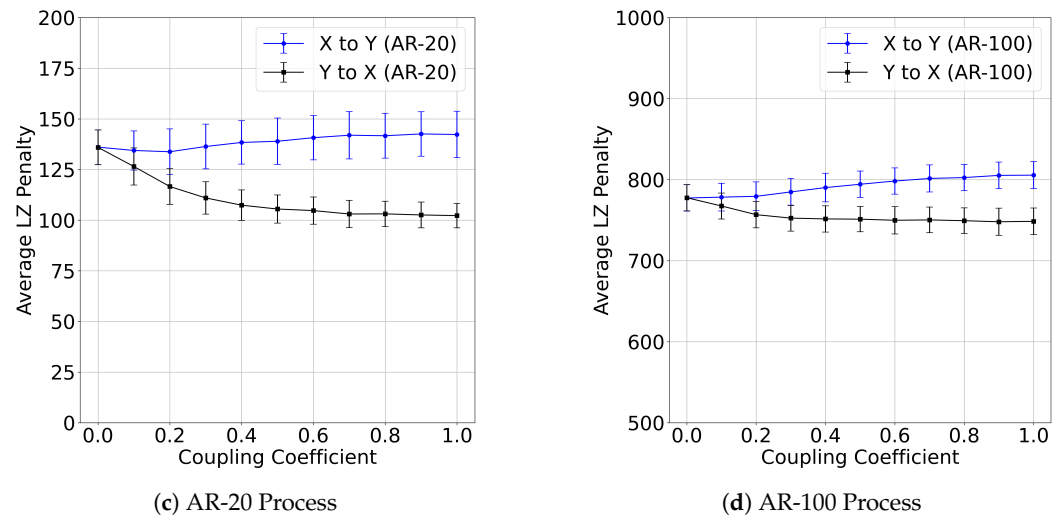


Figure 1. Average LZ Penalty vs. Coupling coefficient for AR processes of order (a) $p = 1$; (b) $p = 5$; (c) $p = 20$; (d) $p = 100$. The coupling coefficient η from Y to X is varied. Results are averaged over 1000 trials. The measure correctly identifies $Y \rightarrow X$ as the causal direction.

4.1.2. Coupled Logistic Map

For the coupled logistic map, the measure correctly identifies the causal direction ($Y \rightarrow X$) for coupling strengths up to $\eta = 0.4$ (Figure 2a). Beyond this point, the two time series become synchronized (Figure 2b), and the causal penalties converge, which is expected as the distinction between cause and effect blurs.

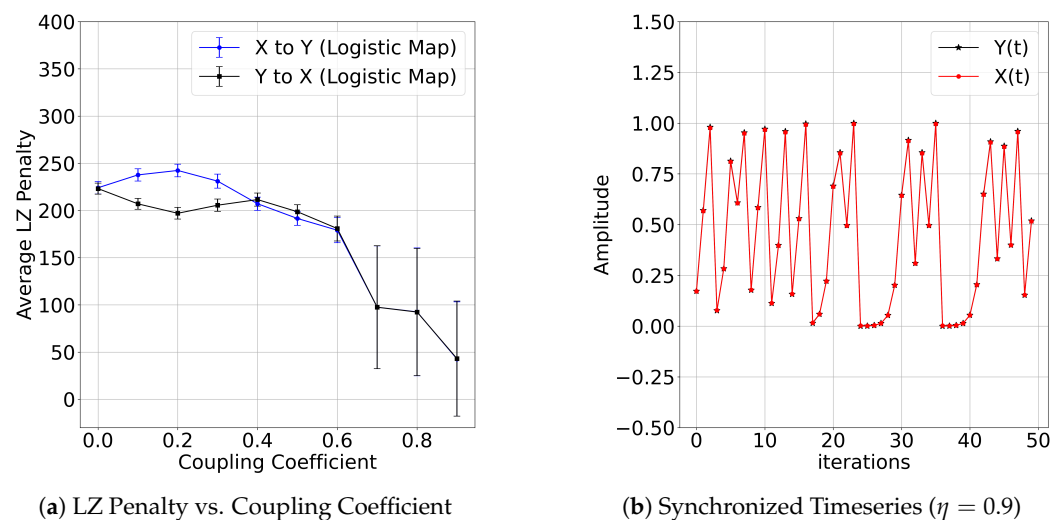


Figure 2. (a) Average LZ Penalty vs. Coupling coefficient for the coupled logistic map. The causal direction $Y \rightarrow X$ is correctly identified for $\eta \leq 0.4$. (b) An instance of the timeseries $X(t)$ and $Y(t)$ for $\eta = 0.9$, showing synchronization.

4.1.3. Three-Variable Causal Structures

Table 2 presents the average causal strengths and variances estimated by the conditional LZP algorithm across several canonical three-variable causal structures. The results validate the algorithm's ability to distinguish between direct, indirect, and spurious influences.

In simple directed cases like $Y \rightarrow Z$, the algorithm assigns strong positive strength to the correct direction (51.05), in comparison to the other cases. In confounder scenarios ($X \rightarrow Y$, $X \rightarrow Z$), both $X \rightarrow Y$ and $X \rightarrow Z$ show high strength (49), while

the spurious $Y \rightarrow Z$ link has a near-zero value (0.40), indicating correct rejection of non-causal association.

For chain structures ($X \rightarrow Y \rightarrow Z$), the algorithm captures both direct ($X \rightarrow Y$: 17.25, $Y \rightarrow Z$: 49.48) and strong indirect influence ($X \rightarrow Z$: 66.02), aligning with expectations. In the collider case ($Y \rightarrow X, Z \rightarrow X$), it reports strong negative values for X 's links to Y and Z , correctly suggesting the reverse direction of influence.

When variables are independent, all strengths remain near zero with higher variance, showing that no causality is detected. Finally, in complex cyclic or converging scenarios (e.g., $X \rightarrow Y, Z \rightarrow Y, Z \rightarrow X$), the algorithm assigns appropriately strong strengths (e.g., -76.99 for $Y \leftrightarrow Z$), capturing the dominant influences. In the cyclic case, the algorithm does not detect any causal structure.

While these results show that conditional LZP can reliably rank the strength and direction of causal links, any categorical decision-making or structure learning would require setting appropriate thresholds on the magnitude of estimated strengths to decide whether a link is present or absent. These thresholds could depend on the application, noise level, or confidence desired, and choosing them remains an important step in translating continuous scores into discrete causal graphs.

Table 2. Strength of Causation and Variance over 20 trials for different three-variable causal structures. The sign indicates direction, and magnitude indicates strength. A positive value for $X \rightarrow Y$ indicates X causes Y . A negative value indicates Y causes X .

Causal Structure	Parameter Values	Average Strength of Causation			Variance		
		X vs. Y	Y vs. Z	X vs. Z	X vs. Y	Y vs. Z	X vs. Z
$Y \rightarrow Z$	$d = 0.5; a = g = i = 0.8$	−34.03	51.05	19.24	0.55	0.68	1.32
$X \rightarrow Y, Y \rightarrow Z, Z \rightarrow X$	$c = f = i = 0.1; a = d = g = 0.5$	−0.34	−0.07	0.66	1.49	0.84	0.92
$X \rightarrow Y, X \rightarrow Z$ (Confounder)	$f = h = 0.8; a = d = g = 0.8$	48.79	0.40	49.17	0.22	0.35	0.42
X, Y, Z Independent	$a = d = g = 0.8$	−0.09	−0.03	−0.37	1.16	1.32	1.28
$X \rightarrow Y, Y \rightarrow Z$ (Chain)	$a = d = 0.5; g = f = i = 0.8$	17.25	49.48	66.02	0.46	0.51	0.33
$Y \rightarrow X, Z \rightarrow X$ (Collider)	$d = g = 0.5; a = b = c = 0.8$	−55.23	−0.19	−55.34	0.59	1.26	0.99
$X \rightarrow Y, Z \rightarrow Y, Z \rightarrow X$	$e = f = c = 0.8; a = d = g = 0.8$	27.44	−76.99	−50.57	0.14	0.42	0.38

4.2. Decision Tree Performance

The LZP Causal and LZ-Distance decision trees were compared against Gini and Entropy-based decision trees, as well as a tree that uses LZ-P ([14]) as the splitting criterion. The comparative performance is summarized in Figure 3.

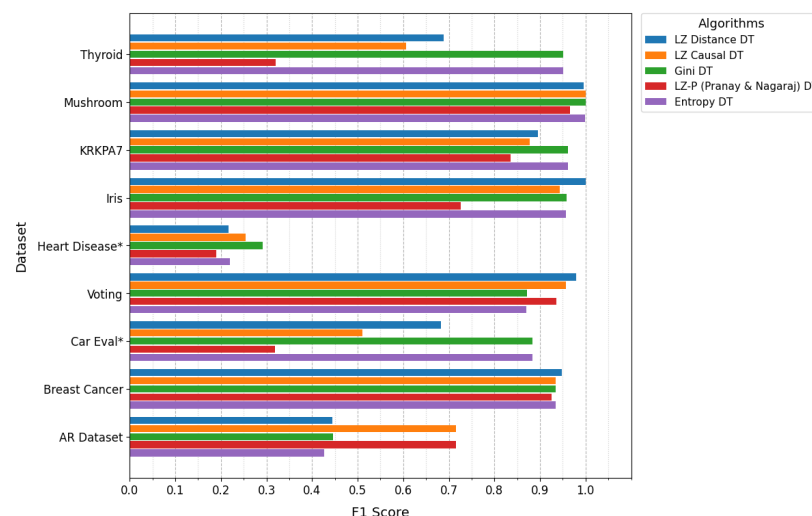


Figure 3. Bar Graph of macro F1 scores of predictions made by LZ Distance Metric, LZP Causal, LZ-P ([14]), Entropy, and Gini impurity-based decision trees for various datasets. Datasets marked with a ‘*’ are highly imbalanced.

The most significant finding is the strong performance of the LZP-Causal Tree on the synthetic AR Dataset, where it achieves a macro F1-score of 0.716. Notably, this performance is matched by the prior LZ-P (Pranay & Nagaraj) DT (0.716), while both methods drastically outperform traditional approaches like the Gini DT (0.446). This result strongly validates the general principle that Lempel–Ziv complexity-based measures are uniquely capable of leveraging underlying causal structures for superior prediction, a task where purely associative metrics fail. The novelty of our LZP, therefore, is affirmed not by a performance gain on this specific dataset, but by its theoretical robustness, such as the guaranteed zero self-causation penalty. On standard, well-balanced datasets such as Mushroom (F1 score of 1.000) and Breast Cancer (0.934), the LZ-Causal tree performs on par with the Gini benchmark, demonstrating its viability as a general-purpose classifier. The results show that the LZ-Decision Tree is also effective in capturing structural similarities between feature and target sequences. Table 3 shows the best and second-best performing algorithms based on Accuracy across all datasets.

However, the results also transparently highlight a key limitation: on highly imbalanced datasets like Thyroid (F1 score of 0.607 vs. Gini’s 0.951), the Gini-based tree maintains superior performance, underscoring the sensitivity of the current LZP formulation to skewed class distributions and reinforcing this as a critical area for future work.

Table 3. Best and second-best performing algorithms based on Accuracy across all datasets.

Dataset	Best Algorithm(s)	Second Best Algorithm(s)
Iris	LZ Distance (1.000)	Gini Impurity (0.967), Entropy (0.967)
Breast Cancer	LZ Distance (0.947)	LZ Causal (0.939)
Voting	LZ Distance (0.979)	LZ Causal (0.957)
Car Evaluation	Gini Impurity (0.945), Entropy (0.945)	LZ Distance (0.829)
KRKPA7 (Chess)	Entropy (0.962)	Gini Impurity (0.961)
Mushroom	LZ Causal (1.000), Gini Impurity (1.000)	Entropy (0.999)
Heart Disease	LZ Causal (0.617)	LZ-P (0.600)
Thyroid	Gini Impurity (0.992), Entropy (0.992)	LZ Distance (0.946)
AR Dataset	LZ Causal (0.717)	LZ-P (0.716)

4.3. Feature Importance and Interpretability

The causal strength metric provides a way to rank features based on their causal influence on the target. Table 4 shows the feature ranking for the Heart Disease dataset for predicting the presence of disease. A similar ranking for the Mushroom dataset is provided in Appendix F.

The causal decision tree for the Heart Disease dataset (Figure 4) provides a clear and interpretable model. The root node splits the data based on serum cholesterol. This feature was chosen because it resulted in the lowest LZP causal penalty, indicating it is the strongest initial causal predictor of heart disease among all features. The features chosen for the top splits (‘chol’, ‘age’, and ‘sex’) directly correspond to the features with the highest causal strength scores in Table 4.

Table 4. Ranking of features of the heart disease dataset based on causal strength to the presence of heart disease.

Feature	Causal Strength
Serum Cholesterol in mg/dL	0.4361
Age	0.2025
Sex	0.1994
ST depression induced by exercise relative to rest	0.0997
Maximum heart rate achieved	0.0545
Number of major vessels (0–3) colored by fluoroscopy	0.0062
Resting blood pressure	0.0016

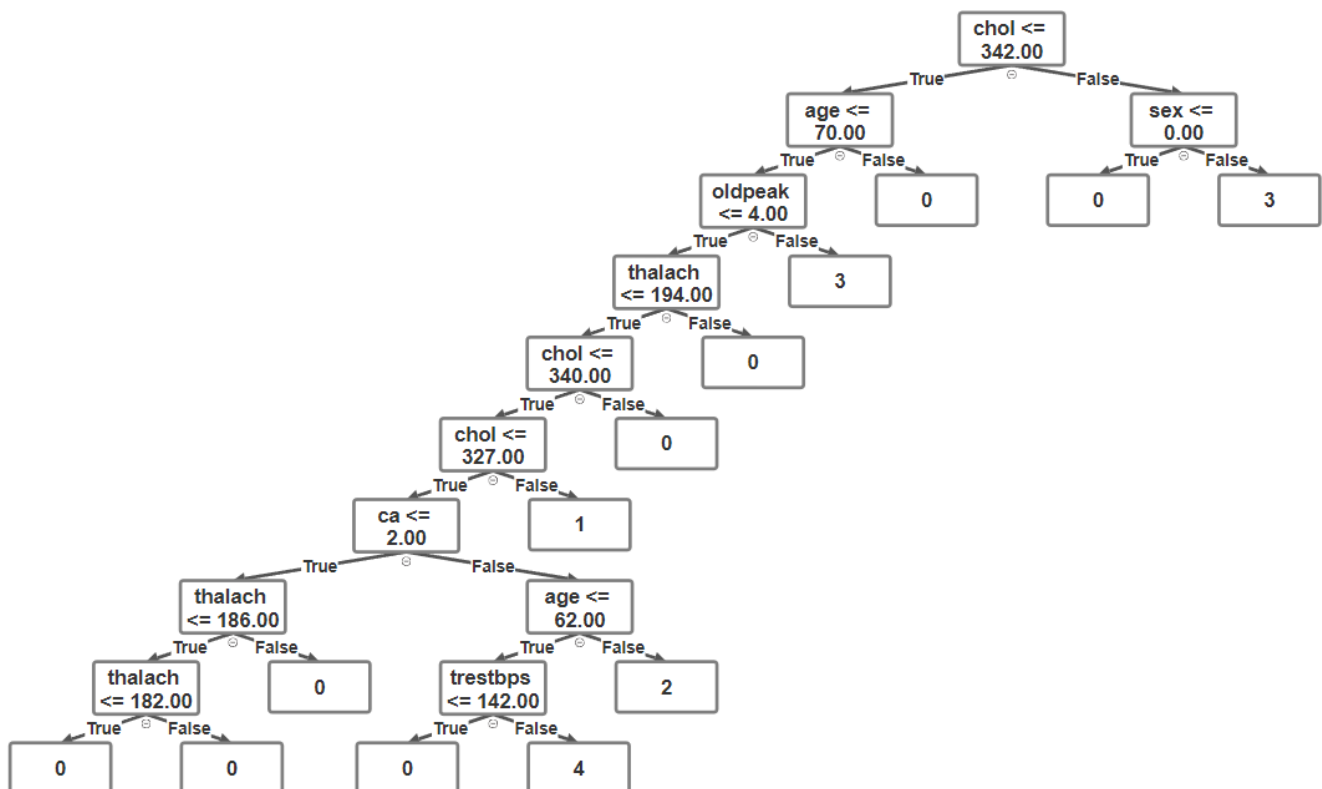


Figure 4. Causal decision tree for Heart Disease dataset. Abbreviations: chol-Serum Cholesterol in mg/dL, oldpeak-ST depression induced by exercise relative to rest, thalach-Maximum heart rate achieved, ca-Number of major vessels (0–3) colored by fluoroscopy, trestbps-Resting blood pressure.

5. Discussion

Our preliminary validation on coupled AR and logistic map systems served to establish the measure’s fundamental soundness. The results also demonstrate that both the proposed LZP Causal and LZ-Distance measures can be effectively integrated into decision tree classifiers. The key finding of this research is the pronounced advantage of the LZP Causal decision tree on datasets with a known underlying causal structure. On the synthetic AR dataset, our causal tree achieved a macro F1-score of 0.716, representing a 60.5% performance improvement over the traditional Gini-based decision tree. This result provides strong evidence for our central hypothesis: by using a splitting criterion based on the LZP causal measure, the model successfully leverages causal information that is inaccessible to purely correlational metrics like Gini impurity. While performance on standard benchmarks was comparable, this specific success demonstrates that our approach is favorable in contexts where causal relationships drive the outcomes.

Extending this, the robust performance of our conditional LZP algorithm on three-variable structures (Table 2) shows its capability to perform in various scenarios. For instance, in the confounder scenario, the algorithm correctly identified the strong causal links from the confounder while assigning a near-zero strength to the spurious correlation. The low scores for independent variables and the inability to find a clear structure in the cyclic case further demonstrate that the measure does not hallucinate causality where none exists. This validation of the underlying measure reinforces the credibility of its application within the decision tree framework.

The feature importance ranking derived from the causal tree provides a transparent, causally grounded explanation for the model’s predictions. For the Heart Disease dataset, the ranking aligns with known medical risk factors, lending credibility to the approach. This moves beyond traditional feature importance measures (like Gini importance or

permutation importance), which are purely correlational and can be misleading in the presence of confounders.

Limitations and Future Work

A key limitation observed is the performance degradation on highly imbalanced datasets. Both LZ-based algorithms work by minimizing a score between the feature's and target's symbolic sequences. In imbalanced scenarios, this minimization can be trivially achieved by collapsing the feature representation into a homogeneous sequence (e.g., all zeros), failing to find a meaningful split. This is because the target sequence for the majority class is long and complex, while the one for the minority class is short and simple. The algorithm may favor splits that do a poor job on the minority class if it leads to a large reduction in penalty/distance for the majority class. Future work should address this by incorporating class weights or using sampling techniques within the splitting criterion.

Another area for future investigation is the sensitivity of the initial data discretization (binning). The conversion of continuous data to symbolic sequences is a critical step, and the choice of binning strategy could influence the results. While simple thresholding was used here, more sophisticated, data-aware binning methods could potentially improve performance.

A fundamental limitation inherited from the standard decision tree architecture is that features are assessed individually at each split. Our LZP metric evaluates the bivariate causal influence of a single feature on the target. This approach cannot capture complex multivariate or interactive causal relationships, where a combination of features jointly causes the outcome, but neither does so individually. Future research could investigate extensions to capture such effects.

6. Conclusions

In this research, we introduced a novel causality measure and a distance metric based on Lempel–Ziv complexity. We successfully demonstrated that our LZP causality measure can identify the correct direction of causation in synthetic datasets, including coupled AR processes and logistic maps, and can handle more complex three-variable structures.

We integrated these measures into a decision tree framework, creating an LZ-Causal and an LZ-Distance classifier. The primary validation of this approach was demonstrated by integrating these measures into a decision tree framework. The key finding of this work is the performance of the LZ-Causal decision tree on the synthetic AR dataset, which was designed with a known causal structure. On this dataset, our model achieved a 60.5% improvement over a traditional Gini-based tree. The LZ-Causal decision tree significantly outperforms traditional methods on data with an inherent causal structure, confirming its ability to exploit more than just statistical correlations. Furthermore, we proposed a method for deriving causal feature importance from the tree structure, offering a more robust form of model interpretability.

While the proposed methods show great promise for building causally aware machine learning models, future work is needed to improve their handling of imbalanced data. We also plan to explore the applicability of the proposed distance metric in bioinformatics, where the order and complexity of sequences like DNA or proteins are of fundamental importance.

Author Contributions: Conceptualization: H.N.B. and N.N.; methodology: D. and H.N.B.; software: D.; validation: D.; formal analysis: D. and H.N.B.; investigation: D. and H.N.B.; writing—original draft preparation: D. and H.N.B.; writing—review and editing, D., H.N.B. and N.N. All authors have read and agreed to the published version of the manuscript.

Funding: Harikrishnan N. B. gratefully acknowledges the financial support from the Prime Minister’s Early Career Research Grant (Project No. ANRF/ECRG/2024/004227/ENS) and the New Faculty Seed Grant (NFSG/GOA/2024/G0906) from BITS Pilani, K. K. Birla Goa Campus.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Python 3.12 code used in this research is available in the following GitHub repository: <https://github.com/i-to-the-power-i/causal-lz-p-decision-tree> (accessed on 18 September 2025). Publicly available datasets were used and are cited in the main text.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AR	Autoregressive
DT	Decision Tree
GC	Granger Causality
LZ	Lempel–Ziv
LZP	Lempel–Ziv Penalty
ML	Machine Learning

Appendix A. Proof of $d_{LZ}(G_x, G_y)$ Being a Distance Metric

We prove that $d_{LZ}(G_x, G_y)$ is a valid distance metric on the metric space of the power set of all symbolic sequences of all finite lengths. Consider two strings x and y . Let G_x and G_y be their grammars encoded by the LZ complexity algorithm.

Definition A1. Let G_x and G_y be two sets. The distance is defined as follows:

$$d_{LZ}(G_x, G_y) = |G_x \setminus G_y| + |G_y \setminus G_x| \quad (\text{A1})$$

where $|A|$ denotes the cardinality of set A .

1. **Non-Negativity:** $d_{LZ}(G_x, G_y) \geq 0$. By definition, the cardinality of any set is non-negative. Hence, $|G_x \setminus G_y| \geq 0$ and $|G_y \setminus G_x| \geq 0$. Their sum must also be non-negative.
2. **Identity of Indiscernibles:** $d_{LZ}(G_x, G_y) = 0 \iff G_x = G_y$. If $G_x = G_y$, then $G_x \setminus G_y = \emptyset$ and $G_y \setminus G_x = \emptyset$. Thus, $d_{LZ}(G_x, G_y) = 0 + 0 = 0$. Conversely, if $d_{LZ}(G_x, G_y) = 0$, then $|G_x \setminus G_y| = 0$ and $|G_y \setminus G_x| = 0$, which implies $G_x \setminus G_y = \emptyset$ and $G_y \setminus G_x = \emptyset$. This is only possible if $G_x = G_y$.
3. **Symmetry:** $d_{LZ}(G_x, G_y) = d_{LZ}(G_y, G_x)$. $d_{LZ}(G_x, G_y) = |G_x \setminus G_y| + |G_y \setminus G_x|$. $d_{LZ}(G_y, G_x) = |G_y \setminus G_x| + |G_x \setminus G_y|$. Since addition is commutative, the property holds.
4. **Triangle Inequality:** $d_{LZ}(G_x, G_z) \leq d_{LZ}(G_x, G_y) + d_{LZ}(G_y, G_z)$ for any set G_z . This is equivalent to showing:

$$|G_x \setminus G_z| + |G_z \setminus G_x| \leq (|G_x \setminus G_y| + |G_y \setminus G_x|) + (|G_y \setminus G_z| + |G_z \setminus G_y|) \quad (\text{A2})$$

Consider an element in $G_x \setminus G_z$. This element must belong to either $G_x \setminus G_y$ or $G_y \setminus G_z$. Thus, we have the general set property $|A \setminus C| \leq |A \setminus B| + |B \setminus C|$. Applying this, we obtain

$$|G_x \setminus G_z| \leq |G_x \setminus G_y| + |G_y \setminus G_z| \quad (\text{A3})$$

and similarly,

$$|G_z \setminus G_x| \leq |G_z \setminus G_y| + |G_y \setminus G_x| \quad (\text{A4})$$

Adding these two inequalities yields the triangle inequality.

Since all four properties are satisfied, $d_{LZ}(G_x, G_y)$ is a valid distance metric.

Appendix B. Dataset Descriptions

A brief description of the datasets used in this study. All are publicly available from the UCI Machine Learning Repository [25] or scikit-learn [26].

- IRIS [27]: 150 instances of iris plants, 3 classes, and 4 numeric features.
- Breast Cancer Wisconsin [28]: 569 instances, 2 classes (malignant/benign), and 30 numeric features derived from cell images.
- Congressional Voting Records [29]: 435 instances, 2 classes (democrat/republican), and 16 categorical features representing votes.
- Car Evaluation [30]: 1728 instances, 4 classes (unacc, acc, good, vgood), and 6 categorical features.
- KRKPA7 [31]: 3196 instances, 2 classes (win/no-win for white), and 35 features describing a chess endgame.
- Mushroom [32]: 8124 instances, 2 classes (edible/poisonous), and 22 categorical features describing physical mushroom characteristics.
- Heart Disease [33]: 303 instances, 5 classes (levels of heart disease), and 13 numeric/categorical features.
- Thyroid [34]: 3772 instances, 3 classes (normal, hyper, hypo), and 21 features. Highly imbalanced.

Appendix C. Hyperparameter Tuning

Hyperparameter tuning for all decision tree models was performed using five-fold cross-validation. For the LZ-Causal and Gini trees dealing with temporal data (AR dataset), a time-series split was used. For all other datasets, a stratified random split was used. The search space for minimum samples per node was [1, 10] and for maximum tree depth was [1, 20]. Table A1 shows the best hyperparameters found and the average training performance.

Table A1. Tuned hyperparameters and training performance metrics for all tested decision tree models.

Dataset	Model	Min Samples	Max Depth	Avg. Train F1	Train F1 Var.
Iris	LZ Distance	2	11	0.940	0.004
	LZ Causal	9	6	0.890	0.016
	Gini Impurity	9	4	0.953	0.002
	LZ-P (Pranay)	6	9	0.761	0.021
	Entropy	9	4	0.962	0.001
Breast Cancer	LZ Distance	9	4	0.890	0.0006
	LZ Causal	2	6	0.910	0.0007
	Gini Impurity	4	8	0.931	0.0008
	LZ-P (Pranay)	9	8	0.890	0.001
	Entropy	3	8	0.923	0.003
Voting	LZ Distance	7	4	0.930	0.002
	LZ Causal	5	9	0.965	0.0001
	Gini Impurity	9	9	0.979	0.0002
	LZ-P (Pranay)	2	7	0.912	0.001
	Entropy	9	9	0.979	0.001

Table A1. Cont.

Dataset	Model	Min Samples	Max Depth	Avg. Train F1	Train F1 Var.
Car Eval.	LZ Distance	2	14	0.660	0.005
	LZ Causal	2	9	0.514	0.0004
	Gini Impurity	2	9	0.832	0.002
	LZ-P (Pranay)	2	9	0.380	0.002
	Entropy	2	9	0.831	0.002
KRKPA7	LZ Distance	2	18	0.900	0.0002
	LZ Causal	4	9	0.862	0.002
	Gini Impurity	3	9	0.968	0.0 *
	LZ-P (Pranay)	4	9	0.760	0.002
	Entropy	4	9	0.965	0.001
Mushroom	LZ Distance	9	9	1.000	0.0 *
	LZ Causal	9	9	0.992	0.0 *
	Gini Impurity	9	7	1.000	0.0 *
	LZ-P (Pranay)	2	9	0.980	0.001
	Entropy	9	5	0.999	0.00076
Heart Disease	LZ Distance	6	19	0.226	0.0002
	LZ Causal	6	9	0.260	0.009
	Gini Impurity	3	9	0.301	0.002
	LZ-P (Pranay)	4	9	0.207	0.003
	Entropy	4	7	0.287	0.006
Thyroid	LZ Distance	9	20	0.664	0.002
	LZ Causal	5	9	0.642	0.0005
	Gini Impurity	2	5	0.966	0.001
	LZ-P (Pranay)	3	8	0.532	0.003
	Entropy	2	7	0.971	0.001
AR Dataset	LZ Distance	2	9	0.503	0.004
	LZ Causal	2	6	0.700	0.008
	Gini Impurity	2	6	0.550	0.004
	LZ-P (Pranay)	9	4	0.716	0.006
	Entropy	9	6	0.544	0.004

* Variance is of the order of 10^{-5} or less, rounded to 0.

Appendix D. Detailed Performance Metrics

Table A2 provides a detailed breakdown of the performance metrics (Accuracy, F1 Score, Precision, and Recall) for all five tested models on the test sets.

Table A2. Performance metrics on test data for all tested decision tree models.

Dataset	Model	Accuracy	F1 Score	Precision	Recall
Iris	LZ Distance	1.000	1.000	1.000	1.000
	LZ Causal	0.933	0.943	0.956	0.939
	Gini Impurity	0.967	0.958	0.978	0.944
	LZ-P (Pranay)	0.767	0.726	0.722	0.731
	Entropy	0.967	0.957	0.976	0.944
Breast Cancer	LZ Distance	0.947	0.948	0.939	0.943
	LZ Causal	0.939	0.934	0.937	0.932
	Gini Impurity	0.938	0.934	0.936	0.932
	LZ-P (Pranay)	0.929	0.926	0.922	0.930
	Entropy	0.938	0.934	0.936	0.932
Voting	LZ Distance	0.979	0.979	0.979	0.979
	LZ Causal	0.957	0.957	0.960	0.958
	Gini Impurity	0.872	0.871	0.896	0.875
	LZ-P (Pranay)	0.936	0.936	0.936	0.936
	Entropy	0.872	0.870	0.896	0.875

Table A2. Cont.

Dataset	Model	Accuracy	F1 Score	Precision	Recall
Car Eval.	LZ Distance	0.829	0.682	0.734	0.664
	LZ Causal	0.760	0.510	0.570	0.496
	Gini Impurity	0.945	0.883	0.914	0.865
	LZ-P (Pranay)	0.685	0.319	0.325	0.319
	Entropy	0.945	0.884	0.914	0.865
KRKPA7	LZ Distance	0.895	0.895	0.895	0.896
	LZ Causal	0.878	0.877	0.879	0.876
	Gini Impurity	0.961	0.961	0.960	0.962
	LZ-P (Pranay)	0.836	0.835	0.836	0.837
	Entropy	0.962	0.962	0.962	0.963
Mushroom	LZ Distance	0.996	0.996	0.995	0.997
	LZ Causal	1.000	1.000	1.000	1.000
	Gini Impurity	1.000	1.000	1.000	1.000
	LZ-P (Pranay)	0.969	0.966	0.970	0.963
	Entropy	0.999	0.999	0.999	0.998
Heart Disease	LZ Distance	0.567	0.217	0.210	0.235
	LZ Causal	0.617	0.254	0.325	0.263
	Gini Impurity	0.550	0.292	0.293	0.308
	LZ-P (Pranay)	0.600	0.190	0.220	0.220
	Entropy	0.550	0.220	0.213	0.234
Thyroid	LZ Distance	0.946	0.688	0.827	0.640
	LZ Causal	0.940	0.607	0.818	0.556
	Gini Impurity	0.992	0.951	0.946	0.957
	LZ-P (Pranay)	0.926	0.320	0.310	0.332
	Entropy	0.992	0.951	0.946	0.957
AR Dataset	LZ Distance	0.467	0.444	0.444	0.445
	LZ Causal	0.717	0.716	0.719	0.718
	Gini Impurity	0.450	0.446	0.450	0.448
	LZ-P (Pranay)	0.716	0.716	0.718	0.718
	Entropy	0.433	0.427	0.430	0.428

Appendix E. Additional Decision Tree Visualizations

Figures A1 and A2 show the decision trees generated by the LZ-Distance and Gini impurity models for the Heart Disease dataset, respectively, for comparison with the causal tree in the main text.

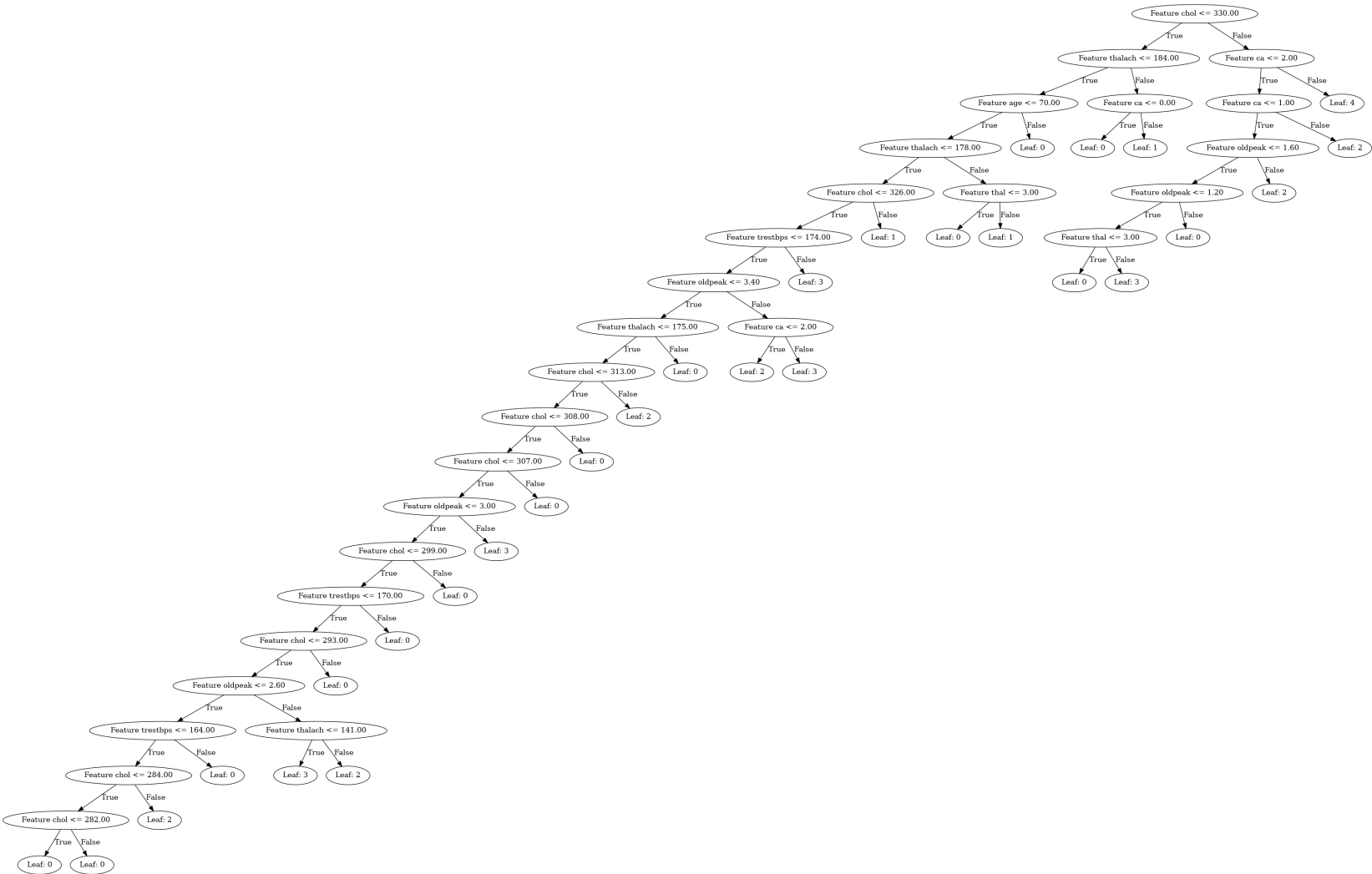


Figure A1. LZ Distance metric-based decision tree for the Heart Disease dataset.

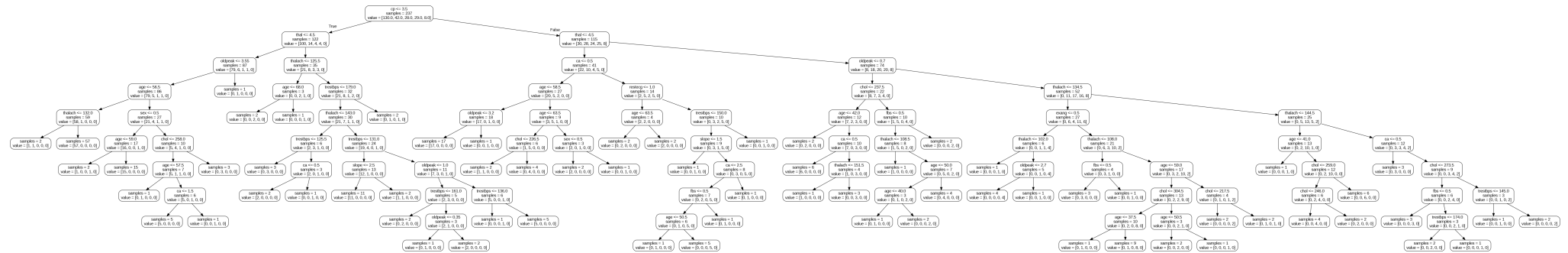


Figure A2. Gini impurity-based decision tree for the Heart Disease dataset.

Appendix F. Feature Ranking for Mushroom Dataset

Table A3 provides the causal strength ranking for the features of the Mushroom dataset, predicting whether a mushroom is poisonous. ‘Odor’ is identified as the most causally influential feature.

Table A3. Ranking of features of the Mushroom dataset based on causal strength as to whether the mushroom is poisonous.

Feature	Causal Strength
Odor	0.3181
Cap Surface	0.3041
Spore print color	0.2444
Ring Number	0.0374
Gill color	0.0374
Habitat	0.0187
Stalk color above ring	0.0187
Stalk surface above ring	0.0094
Stalk shape	0.0094
Cap color	0.0023

References

- Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [\[CrossRef\]](#)
- Longo, L.; Brcic, M.; Cabitza, F.; Choi, J.; Confalonieri, R.; Del Ser, J.; Guidotti, R.; Hayashi, Y.; Herrera, F.; Holzinger, A.; et al. Explainable Artificial Intelligence (XAI) 2.0: A manifesto of open challenges and interdisciplinary research directions. *Inf. Fusion* **2024**, *106*, 102301. [\[CrossRef\]](#)
- Jiao, L.; Yu, J.; Wang, Q.; Xu, J.; Liu, Y.; Hu, X. Causal Inference Meets Deep Learning: A Comprehensive Survey. *Research* **2024**, *7*, 0467. [\[CrossRef\]](#) [\[PubMed\]](#)
- Holzinger, A.; Langs, G.; Denk, H.; Zatloukal, K.; Müller, H. Causability and explainability of artificial intelligence in medicine. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2019**, *9*, e1312. [\[CrossRef\]](#) [\[PubMed\]](#)
- Duong, B.; Nguyen, T.; Huynh, T. Utilising causal inference methods to estimate effects and heterogeneity in observational health data. *PLoS ONE* **2024**, *19*, e0314761. [\[CrossRef\]](#)
- Granger, C.W.J. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica* **1969**, *37*, 424–438. [\[CrossRef\]](#)
- Schreiber, T. Measuring information transfer. *Phys. Rev. Lett.* **2000**, *85*, 461–464. [\[CrossRef\]](#)
- Wiener, N. The theory of prediction. In *Modern Mathematics for Engineers*; Beckenbach, E.F., Ed.; McGraw-Hill: New York, NY, USA, 1956.
- Hu, W.; Lin, A.; Mi, Y. Measuring complex causality and high-order interactions: An extended transfer entropy spectrum approach. *Chaos Solitons Fractals* **2025**, *160*, 112382. [\[CrossRef\]](#)
- Budhathoki, K.; Vreeken, J. Origo: Causal inference by compression. *Knowl. Inf. Syst.* **2018**, *56*, 285–307. [\[CrossRef\]](#)
- Vreeken, J. Causal inference by direction of information. In Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, Canada, 30 April–2 May 2015; pp. 909–917.
- Lempel, A.; Ziv, J. On the complexity of finite sequences. *IEEE Trans. Inf. Theory* **1976**, *22*, 75–81. [\[CrossRef\]](#)
- Borowska, M. Multiscale permutation Lempel–Ziv complexity measure for biomedical signal analysis: Interpretation and application to focal EEG signals. *Entropy* **2021**, *23*, 832. [\[CrossRef\]](#)
- Pranay, S.Y.; Nagaraj, N. Causal discovery using compression-complexity measures. *J. Biomed. Inform.* **2021**, *117*, 103724.
- Kathpalia, A.; Nagaraj, N. Data-based intervention approach for Complexity-Causality measure. *PeerJ Comput. Sci.* **2019**, *5*, e196. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ong, H.J.J.; Lim, B.G.S.; Tiu, B.R.C.; Tan, R.R.P.; Ikeda, K. A Compression-Based Dependence Measure for Causal Discovery by Additive Noise Models. *Commun. Comput. Inf. Sci.* **2025**, *2282*, 5.
- Vitányi, P.; Li, M. On prediction by data compression. In *Machine Learning: ECML-97*; van Someren, M., Widmer, G., Eds.; Springer: Berlin/Heidelberg, Germany, 1997; pp. 14–30.
- Cover, T.M. *Elements of Information Theory*; John Wiley & Sons: New York, NY, USA, 1999.
- Faloutsos, C.; Megalooikonomou, V. On data mining, compression, and Kolmogorov complexity. *Data Min. Knowl. Discov.* **2007**, *15*, 3–20. [\[CrossRef\]](#)

20. Agber, S.; Esin, J.O.; Agbo, M.O. Efficiency Evaluation of Huffman, Lempel-Ziv, And Run-Length Algorithms in Lossless Image Compression for Optimizing Storage and Transmission Efficiency. *Int. J. Comput. Appl.* **2024**, *975*, 8887. [CrossRef]
21. Li, Y.; Geng, B.; Jiao, S. Dispersion entropy-based Lempel-Ziv complexity: A new metric for signal analysis. *Chaos Solitons Fractals* **2022**, *161*, 112400. [CrossRef]
22. Wang, S.; Feng, Z.; Yan, R.; Zhao, Z.; Wang, W. Multivariate multiscale dispersion Lempel–Ziv complexity for fault diagnosis of machinery with multiple channels. *Inf. Fusion* **2024**, *104*, 102152. [CrossRef]
23. Yin, J.; Sui, W.; Zhuang, X.; Sheng, Y.; Li, Y. The Application of Lempel-Ziv Complexity in Medicine Science, Nature Science, Social Science, and Engineering: A Review and Prospect. *IEEE Access* **2024**, *12*, 179330–179352. [CrossRef]
24. Sayood, K. *Introduction to Data Compression*, 5th ed.; Morgan Kaufmann: Burlington, MA, USA, 2017.
25. Kelly, M.; Longjohn, R.; Nottingham, K. The UCI Machine Learning Repository. Available online: <https://archive.ics.uci.edu> (accessed on 1 October 2024).
26. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
27. Fisher, R. Iris Dataset. UCI Machine Learning Repository. 1936. Available online: <https://archive.ics.uci.edu/dataset/53/iris> (accessed on 1 October 2024).
28. Street, W.N.; Wolberg, W.H.; Mangasarian, O.L. Nuclear feature extraction for breast tumor diagnosis. In Proceedings of the Biomedical Image Processing and Biomedical Visualization, San Jose, CA, USA, 1–4 February 1993; Volume 1905, pp. 861–870. Available online: <https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic> (accessed on 1 October 2024).
29. Congressional Voting Records Dataset. UCI Machine Learning Repository. 1987. Available online: <https://archive.ics.uci.edu/dataset/105/congressional+voting+records> (accessed on 1 October 2024).
30. Bohanec, M. Car Evaluation Dataset. UCI Machine Learning Repository, 1988. Available online: <https://archive.ics.uci.edu/dataset/19/car+evaluation> (accessed on 1 October 2024).
31. Shapiro, A. Chess (King-Rook vs. King-Pawn) Dataset. UCI Machine Learning Repository. 1983. Available online: <https://archive.ics.uci.edu/dataset/22/chess+king+rook+vs+king+pawn> (accessed on 1 October 2024).
32. Mushroom Dataset. UCI Machine Learning Repository. 1981. Available online: <https://archive.ics.uci.edu/dataset/73/mushroom> (accessed on 1 October 2024).
33. Janosi, A.; Steinbrunn, W.; Pfisterer, M.; Detrano, R. Heart Disease Dataset. UCI Machine Learning Repository. 1989. Available online: <https://archive.ics.uci.edu/dataset/45/heart+disease> (accessed on 1 October 2024).
34. Quinlan, R. Thyroid Disease Dataset. UCI Machine Learning Repository. 1986. Available online: <https://archive.ics.uci.edu/dataset/102/thyroid+disease> (accessed on 1 October 2024).
35. Li, J.; Ma, S.; Le, T.; Liu, L.; Liu, J. Causal decision trees. *IEEE Trans. Knowl. Data Eng.* **2016**, *29*, 257–271. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.