Chapter 6

Interpretable Conditional Mutual Information Estimation with Adaptive Histograms

This chapter has been published as Marx, A, Yang, L, and van Leeuwen, M **Estimating Conditional Mutual Information for Discrete-Continuous Mixtures using Multi-Dimensional Adaptive Histograms.** In: *Proceedings of the SIAM Conference on Data Mining 2021 (SDM'21), SIAM, 2021.*

Chapter Abstract

Estimating conditional mutual information (CMI) is an essential yet challenging step in many machine learning and data mining tasks. Estimating CMI from data that contains both discrete and continuous variables, or even discrete-continuous mixture variables, is a particularly hard problem. In this chapter, we show that CMI for such mixture variables, defined based on the Radon-Nikodym derivative, can be written as a sum of entropies, just like CMI for purely discrete or continuous data. Further, we show that CMI can be consistently estimated for discrete-continuous mixture variables by learning an adaptive histogram model. In practice, we estimate such a model by iteratively discretizing the continuous data points in the mixture variables. To evaluate the performance of our estimator, we benchmark it against state-of-the-art CMI estimators as well as evaluate it in a causal discovery setting.

6.1 Introduction

In many research areas, such as classification (Lee and Kim 2013), feature selection (Vinh et al. 2014), and causal discovery (Spirtes et al. 2000), estimating the strength of a dependence plays a key role. A theoretically appealing way to measure dependencies is through mutual information (MI) since it has several important properties, such as the chain rule, the data processing inequality, and—last but not least—it is zero if (and only if) two random variables are independent of each other (Cover and Thomas 2012). For structure identification, such as causal discovery, conditional mutual information (CMI) is even more interesting since it can help to distinguish between different graph structures. For instance, in a simple Markov chain $X \to Z \to Y$, X and Y may be dependent, but are rendered independent given Z. Vice versa, a collider structure such as $X \to Z \leftarrow Y$ may introduce a dependence between two marginally independent variables X and Y when conditioned on Z.

While estimating (conditional) mutual information for purely discrete or continuous data is a well-studied problem (Cover and Thomas 2012; Darbellay and Vajda 1999; Gao et al. 2016; Han et al. 2015; Paninski and Yajima 2008), many real-world settings concern a mix of discrete and continuous random variables, such as age (in years) and height, or even random variables that can individually consist of a *mixture* of discrete and continuous components. Although several discretization-based approaches that can estimate MI for a mix of discrete and continuous random variables have recently emerged (Cabeli et al. 2020; Mandros et al. 2020; Suzuki 2016), so far only methods based on k-nearest neighbour (kNN) estimation were shown to work on *mixed variables*, which may consist of discrete-continuous mixture variables (Gao et al. 2017; Mesner and Shalizi 2020; Rahimzamani et al. 2018).

Regardless of the success of kNN-based estimators, discretization-based approaches have attractive properties, e.g., with regard to global interpretation. That is, a natural and understandable way to discretize a continuous random variable is via creating a histogram model, where we cut the sample space of the continuous variable in multiple non-overlapping parts called bins (Scott 2015), or (hyper)rectangles for multi-dimensional variables. Within a bin, we consider the distribution to be constant, which allows us to estimate the density function via

Riemann integration by making the bins smaller and smaller (Cover and Thomas 2012). This definition, however, is less straightforward when mixed variables are involved.

In this chapter, we approach the problem as follows: we first extend the definition of entropy for a univariate discrete-continuous mixture variable given by Politis (Politis 1991) to multivariate variables. Using this definition, we show that CMI for mixed random variables can be written as a sum of entropies that are well-defined through the Radon-Nikodym derivative (see Section 6.2). Exploiting this property, we propose a consistent CMI estimator for such data that is based on adaptive histogram models in Section 6.3. To efficiently learn adaptive histograms from data, in Section 6.4 we define a model selection criterion based on the minimum description length (MDL) principle (Grünwald 2007). Subsequently, we propose an iterative greedy algorithm that aims to obtain the histogram model that minimizes the proposed MDL score in Section 6.5. We discuss related work in Section 6.6 and in Section 6.7, we empirically show that our method performs favourably to state-of-the-art estimators for mixed data and can be used in a causal discovery setting.

6.2 Entropy for Mixed Random Variables

We consider multi-dimensional *mixed random variables*, of which any individual dimension can be discrete, continuous, or a discrete-continuous mixture. Further, we call a vector of such mixed random variables a *mixed random vector*. For a mixed random vector (X, Y), where X and Y are possibly multivariate, we need to adopt the most general definition of mutual information (MI), i.e., the measure-theoretic definition:

$$I(X;Y) = \int_{\mathcal{X}\times\mathcal{Y}} \log \frac{dP_{XY}}{dP_X P_Y} dP_{XY} \ ,$$

where $dP_{XY}/(dP_X P_Y)$ is the Radon-Nikodym derivative, dP_{XY} the joint measure, and $P_X P_Y$ the product measure. It has been proven that $P_X P_Y$ is *absolutely continuous* with respect to P_{XY} (Gao et al. 2017), i.e., $P_{XY} = 0$ whenever $P_X P_Y = 0$; and therefore, such a Radon-Nikodym derivative always exists and $I(X, Y)$ is well-defined. This measure-theoretic definition can be extended to CMI using the chain rule: $I(X;Y|Z) = I(X;\{Y,Z\}) - I(X;Z)$.

It is common knowledge that for purely discrete or continuous random variables, CMI can be written as a sum of entropies, i.e., $I(X;Y|Z) = H(X,Z) + H(Y,Z) - H(X,Y,Z) - H(Z)$. What is not clear, however, is if this formula also holds when (X,Y,Z) contains discrete-continuous mixture random variables. We investigate this problem in two steps. We first define the measure-theoretic entropy for a (possibly multi-dimensional) discrete-continuous mixture random variable and prove it to be well-defined, though previous work claimed the opposite (Gao et al. 2017). Second, using this definition, we prove that (conditional) MI for a mixed random vector can be written as the sum of measure-theoretic entropies, just like purely continuous or discrete random vectors.

6.2.1 A Generalized Definition of Entropy

The measure-theoretic entropy is defined only for one-dimensional random variables (Politis 1991). Building upon this definition, we give an explicit proof that such a one-dimensional measure-theoretic entropy is well-defined, and then extend this definition to the multi-dimensional case, which we prove is also well-defined.

Generalized One-Dimensional Entropy

We start off by reviewing the existing definition for the one-dimensional case (Politis 1991). Given a one-dimensional random variable X, entropy H is defined as

$$H(X) = \int_{\mathbb{R}} \frac{dP_X(x)}{dv(x)} \log \frac{dP_X(x)}{dv(x)} dv(x), \tag{6.1}$$

where $v(\cdot)$ is a measure defined on all one-dimensional Borel sets (Politis 1991). If $v(\cdot)$ is the Lebesgue measure, which we denote as $u(\cdot)$, $H(X)$ becomes the differential entropy. Alternatively, if $v(\cdot)$ is a counting measure, $H(X)$ becomes the common (discrete) entropy.

If, however, X is a discrete-continuous mixture variable, v is defined as follows. We split $\mathbb{R}$ into three disjoint subsets s.t. $\mathbb{R} = S_d \cup S_c \cup S_o$. First, S_o is the subset of $\mathbb{R}$ on which X has zero probability measure, i.e., $P_X(S_o) = 0$. Second, the set S_d contains all discrete points, i.e., S_d is countable and $\forall x \in S_d, P_X(x) > 0$. Third, S_c covers the continuous points, hence $P_X(S_c) + P_X(S_d) = 1$ and for any

Borel set $A \subseteq S_c$ satisfying $u(A) = 0$, we have $P_X(A) = 0$. Based on these three subsets S_d, S_c, and S_o, we can define v as

$$v(A) = u(A \cap S_c) + |A \cap S_d| \ , \tag{6.2}$$

where $|A \cap S_d|$ is the cardinality of this intersection.

To show that the generalized one-dimensional entropy is well-defined, we need to prove that the Radon-Nikodym derivative dP_X/dv always exists. This we show in the following lemma.

Lemma 1. *Given a one-dimensional discrete-continuous random variable X with probability measure P_X, P_X is absolutely continuous w.r.t. v, i.e., $P_X = 0$ whenever $v = 0$, and hence dP_X/dv always exists.*

We provide the proof of Lemma 1, as well as for Lemmas 2 and 3 in Supplementary Material 6.9.1.

Generalized Multi-Dimensional Entropy

In the following, we extend the measure-theoretic entropy definition to a mixed k-dimensional random vector $W = (W_1, \ldots, W_k)$. For each W_i, we define S_d^i, S_c^i, S_o^i and measure v^i as above, and also define the *product measure* v for the k-dimensional random vector as $v = v^1 \times \ldots \times v^k$. Then, define the entropy for W as

$$H(W) = \int_{\mathbb{R}^k} \frac{dP_W(w)}{dv(w)} \log \frac{dP_W(w)}{dv(w)} dv(w). \tag{6.3}$$

To prove that such entropy is well-defined, we show that dP_W/dv always exists.

Lemma 2. *Given a mixed k-dimensional random vector $W = (W_1, \ldots, W_k)$ with probability measure P_W, dP_W/dv always exists.*

Last, based on Lemma 1 and 2, we can prove that just like for a purely continuous or discrete random vector, conditional mutual information for a mixed random vector can be written as a sum of entropies.

Lemma 3. *Given a mixed random vector (X, Y, Z) with joint probability measure P_{XYZ}, we can write $I(X;Y|Z) = H(X,Z) + H(Y,Z) - H(Z) - H(X,Y,Z)$, where each entropy can be defined as in Equation* (6.3)*.*

As a direct implication of the above proof, it follows that mutual information can also be written as the sum of entropies, since it is a special case of CMI with $Z = \emptyset$. With this generalized definition, we can now show how to estimate CMI using adaptive histogram models.

6.3 Adaptive Histogram Models

Adaptive histogram models have been thoroughly studied for continuous random variables (Scott 2015); however, to the best of our knowledge, there exists no rigorous definition of histograms for mixed random variables. Thus, to use histogram models as a foundation to estimate the measure-theoretic (conditional) MI, we need to rigorously define histograms for mixed random variables. We start with the one-dimensional case.

6.3.1 One-Dimensional Histogram Models

A histogram model is typically defined based on a set of consecutive intervals called *bins* (Scott 2015). However, to deal with discrete-continuous mixture random variables, we define the set of bins, denoted as B, such that each bin is either an interval or a set containing only a single point. That is, $B = B' \cup B''$, where B' and B'' are sets of subsets of $\mathbb{R}$, with B' consisting of countable consecutive intervals and B'' consisting of countable single point sets. Last, we define the "width" of a bin using the measure v as defined in Equation 6.2, i.e., for a bin $B_j \in B$ we have

$$v(B_j) = u(B_j \cap B') + |B_j \cap B''| \,. \tag{6.4}$$

As any $B_j \in B''$ contains only a single discrete point, $v(B_j) = 1$ for all $B_j \in B''$.

Further, we define a histogram model M as a set of bins equipped with a parameter vector of length K, where $K = |B|$ is the number of bins. That is, a histogram model M is a family of probability distributions $P_{X,\theta}$, parametrized by the vector $\theta = (\theta_1, \ldots, \theta_K)$. Each element of θ represents the Radon-Nikodym derivative (or density) of each bin. Note that this definition generalizes to purely continuous random variables when $B'' = \emptyset$ and also to discrete random variables if $B' = \ \emptyset$. For the latter case, the histogram model degenerates to a multinomial model.

6.3.2 Multi-Dimensional Histograms

First, we define the set of multi-dimensional bins. For a mixed k-dimensional random vector $W = (W_1, \dots, W_k)$, we define the set of *bins* for each W_i as in Section 6.3.1, denoted as B^i. Consequently, we can define a set of k-dimensional *bins*, denoted B, by the Cartesian product $B = B^1 \times \dots \times B^k$.

Since each B^i is countable, B is also countable, and we can hence assume B is indexed by j. Then, we split B in a similar way as in the one-dimensional case, i.e., $B = B' \cup B''$, where B'' contains *only discrete values*. That is, for any k-dimensional bin $B_j \in B''$, each dimension of B_j is a set that contains a single one-dimensional point. Note that, however, for any $B_j \in B'$, each dimension of B_j can either be a one-dimensional interval or a one-dimensional single-point set. Further, we define the volume of a multi-dimensional bin $B_j \in B$ using the product measure $v(B_j)$ (see Section 6.2.1).

Similar to one-dimensional histograms, a multi-dimensional histogram model M can be described by a probability distribution $P_{W,\theta}$ parametrized by the vector $\theta = (\theta_1, \dots, \theta_K)$, where K is the number of bins and θ_i is the Radon-Nikodym derivative for each bin.

6.3.3 Maximum Likelihood Estimator

Given a possibly multi-dimensional histogram with K bins, we denote the Radon-Nikodym derivative $dP_{W,\theta}/dv$ as f_θ^h and its maximum likelihood estimator as $f_{\hat{\theta}}^h$. Observe that for any parameter $\theta_j \in \theta$, the product $\theta_j v(B_j)$ follows a multinomial distribution. Thus, given a dataset $D = \{D_i\}_{i=1,\dots,n}$, with D_i representing a row, the maximum log-likelihood is denoted as and equal to

$$l_M(D) = \log f^h_{\hat{\theta}(D)}(D) = \log \prod_{j=1}^{K} \left(\frac{c_j}{n \cdot v(B_j)} \right)^{c_j}, \tag{6.5}$$

where c_j and $v(B_j)$ are respectively the number of data points and the bin volumes of bin $j \in \{1 \dots K\}$. Notice that this maximum likelihood generalizes to the purely discrete case (i.e., multinomial distribution) where all $v(B_j) = 1$, and to the purely continuous case (Scott 2015) where v becomes the Lebesgue measure.

6.3.4 Conditional Mutual Information Estimator

Combining all previous theoretical discussions, we can now estimate conditional mutual information for three (possibly multivariate) random variables X, Y and Z by

$$I^h(X;Y|Z) = H^h(X,Z){+}H^h(Y,Z){-}H^h(X,Y,Z){-}H^h(Z)\ .$$

The corresponding measure-theoretic entropies are estimated from k-dimensional data over (X, Y, Z), where k_X, k_Y and k_Z are the corresponding number of dimensions of X, Y and Z. We estimate the entropies as

$$\begin{aligned} H^h(X,Y,Z) &= -\int_{\mathbb{R}^k} f^h_{\hat{\theta}}(x,y,z)\log(f^h_{\hat{\theta}}(x,y,z))dv \\ H^h(X,Z) &= -\int_{\mathbb{R}^{k_X+k_Z}} f^h_{\hat{\theta}}(x,z)\log(f^h_{\hat{\theta}}(x,z))dv \\ H^h(Y,Z) &= -\int_{\mathbb{R}^{k_Y+k_Z}} f^h_{\hat{\theta}}(y,z)\log(f^h_{\hat{\theta}}(y,z))dv \\ H^h(Z) &= -\int_{\mathbb{R}^{k_Z}} f^h_{\hat{\theta}}(z)\log(f^h_{\hat{\theta}}(z))dv \end{aligned} \tag{6.6}$$

in which $f^h_{\hat{\theta}}(x,y,z)$ is the maximum likelihood estimator given the data, while we obtain $f^h_{\hat{\theta}}(x,z)$, $f^h_{\hat{\theta}}(y,z)$, and $f^h_{\hat{\theta}}(z)$ via marginalization from $f^h_{\hat{\theta}}(x,y,z)$. Next, we will prove that I^h is a strongly consistent estimator for conditional mutual information on mixed data.

Theorem 1. Given a mixed random vector (X, Y, Z) with probability measure P_{XYZ},

$$\lim_{v'\to 0}\lim_{n\to\infty} I^h(X;Y|Z) = I(X;Y|Z)$$

almost surely, where n refers to the sample size and v' refers to the maximum of the histogram volumes for bins in B' (defined in Section 6.3.2).

The proof is provided in Supplementary Material 6.9.1. Informally, our proof is based on the following key aspects: 1) All volume-related terms in I^h cancel out, 2) discrete empirical entropy converges to the true entropy almost surely (Antos and Kontoyiannis 2001), and 3) in the limit, differential entropy can be obtained by discretizing a continuous random variable into "infinitely" small bins Cover and Thomas 2012, Theorem 8.3.1. Notably, the order of the double limit in Theorem 1 inherently indicates that n should grow faster than the number of bins (Rudin et al. 1964), which is also required for histograms on purely continuous data to converge (Scott 2015).

6.4 Learning Adaptive Histograms from Data

To efficiently estimate a histogram model that inherits the consistency guarantees from Theorem 1 we need to consider the following requirements. First of all, we need to ensure that we learn a joint histogram model over (X, Y, Z). This is due to the fact that we obtain the lower-dimensional entropies such as $H^h(X, Z)$ by marginalization over the likelihood estimator $f^h_{\hat{\theta}}(x, y, z)$. If we would not learn a joint model, the volume-related terms in $H^h(X, Y, Z), H^h(X, Z), H^h(Y, Z)$, and $H^h(Z)$ would not cancel out. In addition, we need to make sure that the number of bins is in $o(n)$ and increases if we were to increase the number of samples n, while at the same time the size of the bins decreases.

One way to achieve those properties would be to fix the bin width or the number of bins depending on the number of samples. However, such an approach is not very flexible and does not allow for variable bin widths. To allow for a more flexible model, we formally consider the problem of constructing an adaptive multi-dimensional histogram as a model selection problem and employ a selection criterion based on the minimum description length (MDL) principle (Rissanen 1978). MDL-based model selection has been successfully used for learning one-dimensional (Kontkanen and Myllymäki 2007b) and two-dimensional histograms (Kameya 2011; Yang et al. 2023), demonstrating adaptivity to both local density changes and sample size.

We now briefly introduce the MDL principle and define the MDL-optimal histogram model. Specifically, while previous work (Kameya 2011; Kontkanen and Myllymäki 2007b; Yang et al. 2023) only considers purely continuous data (or more precisely, data with arbitrarily small precision), we apply the MDL principle to mixed-type data, based on our rigorous definition of histogram models for mixed random variables. On top of that, we empirically show that our score fulfils the desired properties—i.e. the number of bins grows as $o(n)$.

6.4.1 MDL and Stochastic Complexity

The minimum description length principle is arguably one of the best off-the-shelf model selection criteria (Grünwald 2007), which has been successfully applied to many machine learning and data mining tasks. The general idea is to assign a code length to data D compressed by a model M, e.g., a histogram model. Given a collection of candidate models, denoted as $\mathcal{M}$, MDL selects the model M^* that minimizes the joint code length of the model and the data. Formally, our goal is to find

$$M^* = \arg\min_{M \in \mathcal{M}} L(D|M) + L(M), \tag{6.7}$$

where $L(D|M)$ denotes the code length[1] of the data given the model, while $L(M)$ denotes the code length needed to encode the model.

The optimal way of encoding data D given M, in the sense that it will result in minimax *regret*, is to use the *normalized maximum likelihood (NML)* code (Grünwald 2007). Accordingly, the code length of the data is called *stochastic complexity (SC)*, which is defined as the sum of the negative log-likelihood $-l_M(D)$, defined in Equation 6.5, and the *parametric complexity* (also called *regret*) $\log R(n, K)$ (Grünwald 2007). The parametric complexity of a histogram model with K bins is given by (Kontkanen and Myllymäki 2007b; Yang et al. 2023)

$$R(n,K) = \sum_{c_1+\cdots+c_K=n} \frac{n!}{c_1! \cdots c_K!} \prod_{i=1}^{K} \left(\frac{c_i}{n}\right)^{c_i} ,$$

and can be computed in sub-linear time (Mononen and Myllymäki 2008).

6.4.2 Code Length of the Model

Given a dataset D with n rows and k individual columns D^j, we now define the model class $\mathcal{M}$. First, we create fixed bins according to B'' (as defined in Section 6.3.2) per discrete value that occurs in D_j. Next, we enumerate all possible bins for B' with fixed precision ϵ. To this end, denote the remaining non-discrete data points in D_j as D_j^c. If D_j^c is empty D_j corresponds to a discrete variable and we can stop here. Otherwise, we create all possible cut points for D_j^c as $C_j^0 = \{\min(D_j^c), \min(D_j^c){+}\epsilon, \ldots, \max(D_j^c)\}$. By selecting a subset of cut points $C_j \subseteq C_j^0$, we get a valid solution for B'. We can enumerate all possible segmentations by enumerating each $C_j \subseteq C_j^0$.

By repeating this process for each dimension, we obtain our model class $\mathcal{M}$.

[1] The code length L denotes the number of bits needed to describe the given object. Hence, all logarithms are to base 2 and $0 \log 0 = 0$.

Further, we get the code length for a model $M \in \mathcal{M}$ by encoding all combinations of cut points for each dimension (Kontkanen and Myllymäki 2007b), i.e.,

$$L(M) = \sum_{j \in \{1,\ldots,k\}} L(C_j) = \sum_{j \in \{1,\ldots,k\}} \log \binom{|C_j^0|}{|C_j|} . \tag{6.8}$$

This completes the definition of our final optimization score $L(D|M) + L(M)$.

To proof consistency for this score, we need to show that the number of selected bins grows at rate $o(n)$. Since the theoretical analysis is rather difficult, we instead empirically demonstrate this property for Gaussian distributed data in Supplementary Material 6.9.3. In the next section, we present an iterative greedy algorithm that optimizes our MDL score.

6.5 Implementation

In this section, we describe our algorithm to estimate the joint entropy $H(X_1, \ldots, X_k)$ for a k-dimensional discrete-continuous mixture random vector.

6.5.1 Algorithm

To discretize a one-dimensional random variable X, we first create bins for the discrete values of X and then discretize the continuous values. We detect discrete data points by checking if a single value x in the domain $\mathcal{X}$ of X occurs multiple times. If a user-defined threshold t, e.g., 5 is reached, we create a bin for this point. To discretize the remaining continuous values, we start by splitting $\mathcal{X}$ into K_{init} equi-width bins, which we can safely choose from the complexity class $o(\sqrt{n})$ (see Supplementary Material 6.9.3). Using dynamic programming, we compute the variable-width histogram model M that minimizes $L(D, M)$ in quadratic time w.r.t. K_{init} (Kontkanen and Myllymäki 2007b).

Since the runtime complexity to compute the optimal variable-width histogram over a multi-dimensional random variable would grow exponentially w.r.t. k, we opt for an iterative greedy algorithm (we provide the pseudocode in Supplementary Material 6.9.2). We start by initializing the optimization: for every dimension, we fix bins for the discrete values and put the remaining continuous values into a single bin. Then, in each iteration, we compute a candidate discretization for each dimension and keep the discretization of that dimension that provides the highest gain in compression. To compute a candidate discretization for a dimension X_j, we extend the one-dimensional algorithm described above. That is, we determine those cut points for X_j that provide the highest gain in $L(D, M)$, while keeping the bins for the remaining dimensions fixed. We repeat this until the maximum number of iterations $i_{\max}$ is reached or we cannot further decrease $L(D, M)$.

6.5.2 Complexity

The complexity of discretizing a univariate random variable is in $\mathcal{O}(K_{\max} \cdot (K_{\text{init}})^2)$ and depends on the number of initial bins K_{init} and the maximum number of bins $K_{\max}$, which we typically chose as a fraction of K_{init} (both in $o(\sqrt{n})$). In a multi-dimensional setting we have to multiply this complexity by the current domain size of the remaining variables, since we have to update each bin conditioned on those. In the worst case, this number is equal to $(K_{\max})^{k-1}$. Overall, we apply this procedure—if all variables are continuous—$i_{\max} \cdot k$ times.

6.6 Related Work

We discuss related methods for adaptive histograms and (conditional) mutual information estimation.

Both theoretical properties and practical issues of density estimation using histograms have been studied for decades (Scott 2015). Various algorithms have been proposed for the challenging task of constructing an adaptive one-dimensional histogram, among which the MDL-based histogram (Kontkanen and Myllymäki 2007b) is considered to be the state-of-the-art, as it is self-adaptive to both local density structure and sample size and does not have any hyperparameters. Learning adaptive multivariate histograms is even harder due to the combinatorial explosion of the search space. One approach is to resort to the dyadic CART algorithm (Klemelä 2009); various methods designed for specific tasks also exist (Kameya 2011; Weiler and Eggert 2007). Our algorithm is similar to that of Kameya (Kameya 2011), but they only consider the two-dimensional case.

For discrete data, (conditional) mutual information estimation is a well-studied problem (Cover and Thomas 2012; Han et al. 2015; Marx and Vreeken 2019; Paninski 2003; Valiant and Valiant 2011) and it has been shown that mutual information can be estimated using the $3H$ principle (Han et al. 2015). An important observation is that for discrete data, the empirical estimator for entropy is sub-optimal (Paninski 2003), which encouraged the design of more efficient entropy estimators with sub-linear sample complexity (Han et al. 2015; Valiant and Valiant 2011).

For estimating (conditional) mutual information on continuous data or a mix of discrete and continuous data, three classes of approaches exist. The first class concerns kernel density estimation (KDE) methods (Gao et al. 2016; Paninski and Yajima 2008), which perform well on continuous data; however, no KDE-based MI and CMI estimation methods exist that are designed for discrete-continuous mixture random variables. Moreover, bandwidth tuning for KDE can be computationally expensive, which becomes even worse for mixed data, as different bandwidths may be needed for discrete random variables. The second class of methods relies on k-nearest neighbour (kNN) estimates (Frenzel and Pompe 2007; Kozachenko and Leonenko 1987; Kraskov et al. 2004), which have been established as the state of the art (Gao et al. 2017; Rahimzamani et al. 2018). kNN approaches can be applied not only to a mix of discrete and continuous variables, but can also be used as consistent MI (Gao et al. 2017) and CMI (Mesner and Shalizi 2020; Rahimzamani et al. 2018) estimators for discrete-continuous mixtures. The third class of methods first discretizes the continuous random variables and then calculates mutual information from the discretized variables (Cover and Thomas 2012; Darbellay and Vajda 1999; Suzuki 2016). Two recent approaches based on adaptive partitioning for mixed random variables have been proposed (Cabeli et al. 2020; Mandros et al. 2020). While Mandros et al. (2020) focus on mutual information and its application to functional dependency discovery, Cabeli et al. (2020), similar to us, build upon an MDL-based score to estimate MI and CMI, to which we compare in Section 6.7. The key difference is that Cabeli et al. (2020) compute $I(X;Y|Z)$ as $(I(X;\{Y,Z\}) - I(X;Z) + I(Y;\{X,Z\}) - I(Y;Z))/2$ and maximize each of the four terms (with penalty terms) directly, while we first learn a joint histogram.

To the best of knowledge, we are the first to propose a CMI estimator for discrete-continuous mixture variables based on discretization or histogram density estimation. Our method can consistently estimate CMI on mixed random variables containing discrete-continuous mixtures. We focus on histogram-based models instead of kNN estimation, since histograms are more interpretable (Scott 2015) and do not require tuning of the parameter k, which can have a large impact on the outcome.

6.7 Experiments

In this section, we empirically evaluate the performance of our approach. First, we will benchmark our estimator against state-of-the-art CMI estimators on different data types. After that, we evaluate how well our estimator is suited to test for conditional independence in a causal discovery setup. For reproducibility, we make our code available online.[2]

6.7.1 Mutual Information Estimation

On the mutual information estimation task, we compare our approach to the state-of-the-art MI estimators. In particular, we compare against FP (Frenzel and Pompe 2007), RAVK (Rahimzamani et al. 2018) and MS (Mesner and Shalizi 2020), which all rely on kNN estimates, and MIIC (Cabeli et al. 2020), which is a discretization-based method. All of those can be applied to our setup, but only the authors of RAVK and MS specifically consider discrete-continuous mixture variables. We apply MIIC using the default parameters and use $k = 10$ for all kNN-based approaches.[3] For our algorithm, we set the maximum number of iterations and the threshold to detect discrete points in a mixture variable to 5, set $K_{\text{init}} = 20 \log n$ and $K_{\max} = 5 \log n$. To comply with the literature, we compute all entropies in this section using the *natural logarithm.*

Experiment I-IV

As a sanity check, we start with an experiment on purely continuous data. That is, for **Experiment I**, let X and Y be Gaussian distributed random variables

[2] https://github.com/ylincen/CMI-adaptive-hist.git

[3] We evaluated all approaches with $k = 5, 10, 20$. Since $k = 10$ had the best trade-off and is close to $k = 7$ as used by Mesner and Shalizi (Mesner and Shalizi 2020), we report those results.

with mean 0, variance 1, and covariance 0.6. Consequently, the correlation ρ between X and Y is 0.6 and true MI can be calculated as $I(X;Y) = -\frac{1}{2}\log(1-\rho^2)$. In **Experiment II**, X is discrete and drawn from $\text{Unif}(0, m-1)$, with $m = 5$ and Y is continuous with $Y \sim \text{Unif}(x, x+2)$ for $X = x$. Therefore, $I(X;Y) = \log(m) - \frac{(m-1)\log 2}{m}$ (Gao et al. 2017). Next, for **Experiment III**, X is exponentially distributed with rate 1 and Y is a zero-inflated Poissonization of X—i.e., $Y = 0$ with probability $p = 0.15$ and $Y \sim \text{Pois}(x)$ for $X = x$ with probability $1 - p$. The ground truth is $I(X;Y) = (1-p)(2\log 2 - \gamma - \sum_{k=1}^{\infty} \log k \cdot 2^{-k}) \approx (1-p)0.3012$, where γ is the Euler-Mascheroni constant (Gao et al. 2017). Last, in **Experiment IV**, we generate the data according to the Markov chain $X \to Z \to Y$ (see Mesner and Shalizi (Mesner and Shalizi 2020)). In particular, X is exponentially distributed with rate $\frac{1}{2}$, $Z \sim \text{Pois}(x)$ for $X = x$ and Y is binomial distributed with size $n = z$ for $Z = z$ and probability $p = \frac{1}{2}$. Due to the Markov chain structure, the ground truth $I(X;Y \mid Z) = 0$.

For each of the above experiments, we sample data with sample size $n \in \{100, 200, \dots, 1\,000\}$ and generate 100 data sets per sample size. We run each of the estimators on the generated data and show the mean squared error (MSE) of each estimator in Figure 6.1. Overall, our estimator performs best or very close to the best throughout the experiments and reaches an MSE lower than 0.001 with at most 1 000 samples. The best competitors are MS and MIIC; however, both are biased when we consider discrete-continuous mixture variables, as we show in Experiment V.

Experiment V

Next, we generate data according to a discrete-continuous mixture (Gao et al. 2017). Half of the data points are continuous, with X and Y being standard Gaussian with correlation $\rho = 0.8$, while the other half follows a discrete distribution with $P(1,1) = P(-1,-1) = 0.4$ and $P(1,-1) = P(-1,1) = 0.1$. In addition, we generate Z independently with $Z \sim \text{Binomial}(3,\ 0.2)$. Hence the ground truth is equal to $I(X;Y) = I(X;Y \mid Z) = 0.4 \cdot \log \frac{0.4}{0.5^2} + 0.1 \cdot \log \frac{0.1}{0.5^2} - \frac{1}{4}\log(1-0.8^2) \approx 0.352$.

In Figure 6.2 (top) we show the mean and MSE for this experiment. We see that our estimator starts by overestimating the true value, but its average quickly converges to the true value, while the competing estimators seem to have a slightly positive or negative bias. Especially FP and MIIC, which were not designed for

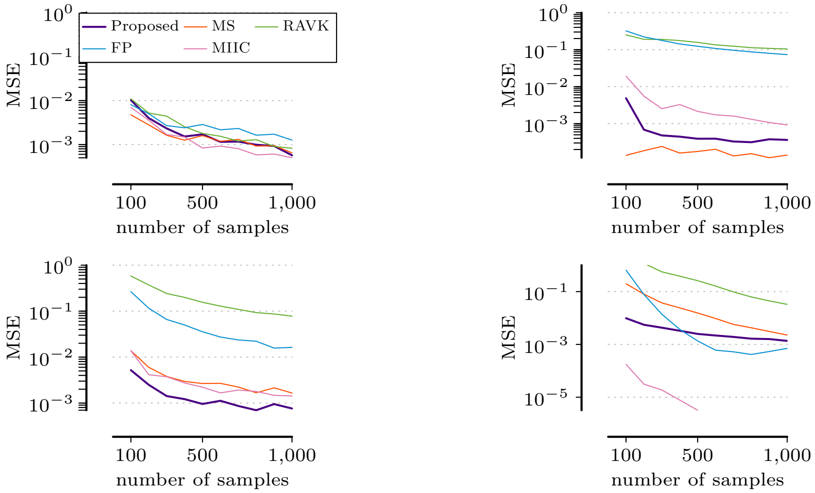

Figure 6.1: Synthetic data with known ground truth. Ordered from top-left to bottom-right, we show the MSE for Experiments I-IV, for our estimator and competing algorithms MS, RAVK, FP and MIIC.

this setup, have a clear bias even for 1 000 data points. The same trend can be observed for MSE.

Experiment VI

Last, we test how sensitive our method is to dimensionality. We generate X and Y as in Experiment II, but fix n to 2 000 and add k independent random variables, $Z_k \sim \text{Binomial}(3, 0.5)$.

Figure 6.2 (bottom) shows the mean and MSE. Our estimator recovers the true CMI up to a negligible error up to $k = 2$. After that, it starts to slowly underestimate the true CMI. This can be explained by the fact that the model costs increase linearly with the domain size and hence, we will fit fewer bins to the continuous variable for large k. We validated this conjecture by repeating the experiment for $n = 10\,000$. On this larger sample size, the MSE for our estimator remained below 0.001 even for $k = 4$. While MIIC is slightly more stable for $k \geq 3$, the competing kNN-based estimators deviate quite a bit from the true estimate for higher dimensions.

Overall, we are on par with or outperform the best competitor throughout Experiments I–VI. Especially on mixture data, which is our main focus, our method

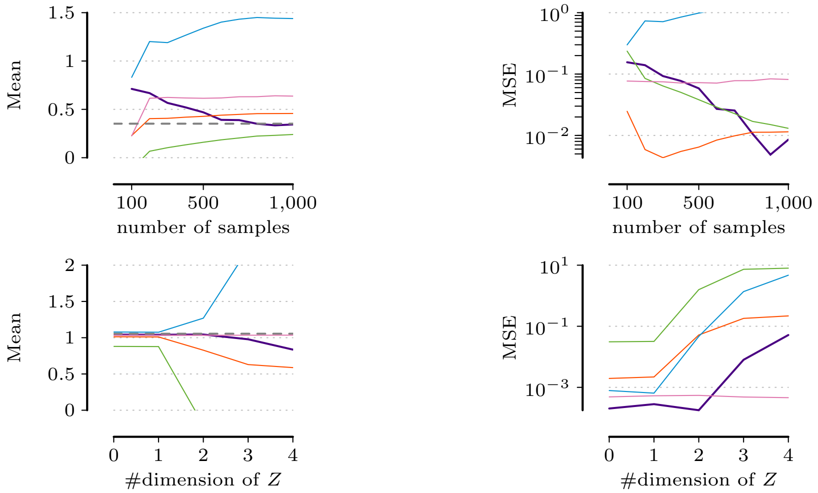

Figure 6.2: Top row: Experiment V, where we show the mean of the estimators (left) with the true CMI as a dashed gray line and the MSE (right). Bottom row: Experiment VI, where the sample size is constant at 2 000 and the x-axis refers to the number of dimensions of Z. We show the mean (left) and MSE (right). The color coding is chosen as in Figure 6.1.

is the only one that converges to the true estimate.

6.7.2 Independence Testing

In theory, two random variables X and Y are conditionally independent given a set of random variables Z, denoted as $X \perp\!\!\!\perp Y \mid Z$, if $I(X;Y \mid Z) = 0$. Vice versa, X and Y are dependent given Z, if $I(X;Y \mid Z) > 0$. In practice, we cannot simply rely on our estimator to conclude independence: due to the monotonicity of mutual information, i.e., $I(X;Y) \leq I(X;Y \cup Z)$, estimates will rarely be *exactly zero* in the limited sample regime, but only *close to zero* (Marx and Vreeken 2019; Vinh et al. 2014). To address this problem, we use our algorithm to discretize X, Y and Z, and compute $I_{\mathcal{C}}(X;Y|Z) := \max\{0, I_n(X_d;Y_d|Z_d) + \mathcal{C}_n(X_d;Y_d|Z_d)\}$, where $\mathcal{C}_n$ is a correction term calculated from the discretized variables, which is negative. In the following, we evaluate our estimator with two different correction criteria. The first one is a correction for mutual information based on the Chi-squared distribution, with $\mathcal{C}_n$ equal to $-\mathcal{X}_{\alpha,l}/2n$ (Vinh et al. 2014), where $\mathcal{X}_{\alpha,l}$ refers to the critical value of the Chi-squared distribution with significance level α and degrees of freedom l. We can compute the degrees of freedom l from the domain

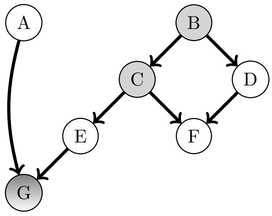

Figure 6.3: Synthetic network with continuous (white), discrete (gray) and mixed (shaded) random variables consisting of different causal structures, such as colliders, a chain ($C \to E \to G$), and a fork ($C \leftarrow B \to D$).

sizes of the discretized variables for the conditional case as $l = (|\mathcal{X}| - 1)(|\mathcal{Y}| - 1)|\mathcal{Z}|$ (Suzuki 2019), and for the unconditional case as $l = (|\mathcal{X}| - 1)(|\mathcal{Y}| - 1)$. For the second correction, we replace each empirical entropy in $I_n(X_d; Y_d|Z_d)$ with its corresponding stochastic complexity term as defined in Section 6.4.1. If we subtract the regret terms for $H_n(X_d, Y_d, Z_d)$ and $H_n(Z_d)$ from those for $H_n(X_d, Z_d)$ and $H_n(Y_d, Z_d)$, we are guaranteed to get a negative value, thus a valid regret term (Marx and Vreeken 2019). In the following, we refer to the test using the Chi-squared correction as $I_{\mathcal{X}^2}$ and to the one based on stochastic complexity as I_{SC}.

To test how well $I_{\mathcal{X}^2}$ and I_{SC} perform on mixed-type and continuous data, we benchmark both against state-of-the-art kernel-based tests RCIT and RCoT (Strobl et al. 2019), as well as JIC (Suzuki 2016), and MIIC (Cabeli et al. 2020), which are both discretization-based methods using a correction based on stochastic complexity.[4] To apply RCIT and RCoT on mixed data, we treat the discrete data points as integers. In the following, we evaluate the performance of each test in a causal discovery setup. In addition, we provide a more detailed description of the data generation and experiments on individual collider and non-collider structures in Supplementary Material 6.9.3.

Causal Discovery

To evaluate our test in a causal discovery setting, we generate data according to a small synthetic network—shown in Figure 6.3—that consists of a mixture of generating mechanisms that we used in experiments I-IV and includes continuous

[4] Note that MIIC calculates stochastic complexity based on factorized NML and JIC uses an asymptotic approximation of stochastic complexity, while we use quotient NML for I_{SC} (Marx and Vreeken 2019; Silander et al. 2018).

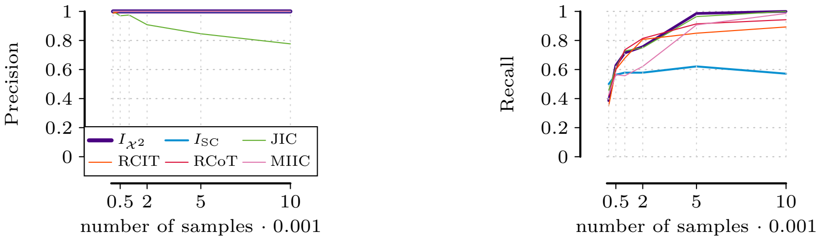

Figure 6.4: Precision (left) and recall (right) on undirected graphs inferred using the PC-stable algorithm equipped with the corresponding independence test. The data is generated from the graph shown in Figure 6.3.

and discrete (ordinal) random variables and one mixture variable, which is partially Gaussian and partially Poisson distributed (for details see Supplementary Material 6.9.3). To evaluate how well the ground truth graph can be recovered, we apply the PC-stable algorithm (Colombo and Maathuis 2012; Spirtes et al. 2000) equipped with the different independence tests, where we use $\alpha = 0.01$ for $I_{\mathcal{X}^2}$, RCIT and RCoT.

Fig 6.4 shows recovery precision and recall for the undirected graph, averaged over 20 draws per sample size $n \in \{100, 500, 1\,000,\ 2\,000, 5\,000, 10\,000\}$.

We see that overall $I_{\mathcal{X}^2}$ performs best and is the only method that reaches both a perfect accuracy and recall. While JIC also reaches a perfect recall, it finds too many edges leading to a precision of only 80%. Although also MIIC, RCIT and RCoT have a perfect precision, their recall is worse than for $I_{\mathcal{X}^2}$. Neither of the kernel-based tests manages to recall all the edges even for 10 000 samples. After a closer inspection, this is due to the edge $E \to G$ that involves the discrete-continuous variable G. If we compare $I_{\mathcal{X}^2}$ to I_{SC}, we clearly see that the latter is too conservative, which leads to a bad recall.

6.8 Conclusion

We proposed a novel approach for the estimation of conditional mutual information from data that may contain discrete, continuous, and mixture variables. To be able to deal with discrete-continuous mixture variables, we defined a class of generalized adaptive histogram models. Based on our observation that CMI for mixture-variables can be written as a sum of entropies, we presented a CMI estimator based on such histograms, for which we proved that it is consistent.

Further, we used the minimum description length principle to formally define optimal histograms, and proposed a greedy algorithm to practically learn good histograms from data. Finally, we demonstrated that our algorithm outperforms state-of-the-art (conditional) mutual information estimation methods, and that it can be successfully used as a conditional independence test in causal graph structure learning. Notably, for both setups, we observe that our approach performs especially well when mixture variables are present.

6.9 Supplementary Material

The supplementary material is structured as follows. First, we provide proofs for all lemmas and theorems. After that, we provide the pseudocode for our algorithm. Last, we provide additional experiments and details for the data generation for the causal discovery experiment.

6.9.1 Proofs

Proof of Lemma 1

Proof. Given a Borel set $A \subseteq \mathbb{R}$ such that $v(A) = u(A \cap S_c) + |A \cap S_d| = 0$, we have $u(A \cap S_c) = 0$ due to non-negativity of any measure, as well as $|A \cap S_d| = 0$. Since $A \cap S_c \subseteq S_c$, by the definition of S_c we have $P(A \cap S_c) = 0$. It remains to show that $A \cap S_d = \emptyset$, which we do by contradiction. Assume that $A \cap S_d \neq \emptyset$, then there exists $x \in A \cap S_d$ s.t. for a set containing only x, $|\{x\}| = 1$. Then $|A \cap S_d| \geq |\{x\}| = 1$, which contradicts $|A \cap S_d| = 0$. Thus, we must have $A \cap S_d = \emptyset$ and then $P_X(A) = 0$. □

Proof of Lemma 2

Proof. Given a k-dimensional Borel set A, there exist one-dimensional Borel sets $A_1, \dots, A_k$ such that $A = A_1 \times \ldots \times A_k$. If $v(A) = 0$, then there exists at least one $v^i, i \in \{1, \dots, k\}$, such that $v^i(A_i) = 0$. Thus, by Lemma 1, $P_{W_i}(A_i) = 0 \Rightarrow P_W(\mathbb{R} \times \ldots \times \mathbb{R} \times A_i \times \mathbb{R} \times \ldots \times \mathbb{R}) = 0 \Rightarrow P_W(A) = 0$, as $A = A_1 \times \ldots \times A_k \subseteq \mathbb{R} \times \ldots \times \mathbb{R} \times A_i \times \mathbb{R} \times \ldots \times \mathbb{R}$. □

Proof of Lemma 3

Proof. We first proof the statement for $Z \neq \emptyset$, for which we can write $I(X;Y|Z) = I(X;\{Y,Z\}) - I(X;Z)$ by the chain rule for mutual information. Thus, it suffices to prove that $I(X;Z) = H(X) + H(Z) - H(X,Z)$ and $I(X;\{Y,Z\}) = H(X) + H(Y,Z) - H(X,Y,Z)$. Next, denote v as the product measure defined based on (X,Z), where $v = v^1 \times \ldots \times v^{k_{XZ}}$, and k_{XZ} is the number of dimensions of X plus that of Z; then by Lemma 2, we also have $P_{XZ} \ll v$. Then, we show that $P_X P_Z \ll v$. For some k_{XZ}-dimensional Borel set $A = A_1 \times \ldots \times A_{k_{XZ}}$ satisfying $v(A) = 0$ there exists $v^i \in \{v^1, \ldots, v^{k_{XZ}}\}$ such that $v^i(A_i) = 0$. Hence, $P_X P_Z(A) = 0$ because $0 \leq P_X P_Z(A) = P_X P_Z(A_1 \times \ldots \times A_{k_{XZ}}) \leq P_X P_Z(\mathbb{R} \times \ldots \mathbb{R} \times A_i \times \mathbb{R} \ldots \times \mathbb{R}) = P_i(A_i) = 0$, where P_i is the marginalization of the product measure $P_X P_Z$ to the ith dimension and $P_i(A_i) = 0$ is because $v^i(A_i) = 0$ by the definition of v.

Finally, by the chain rule of the Radon-Nikodym derivative we have that

$$I(X;Z) = \int \log \frac{dP_{XZ}}{dP_X P_Z} dP_{XZ} \tag{6.9}$$

$$= \int \log \frac{dP_{XZ}/dv}{dP_X P_Z/dv} (dP_{XZ}/dv) dv \tag{6.10}$$

$$= H(X) + H(Z) - H(X,Z) \; . \tag{6.11}$$

The proof for $I(X;\{Y,Z\})$ is equivalent. If $Z = \emptyset$, CMI reduces to $I(X;Y)$, for which we can prove the statement in the same manner. □

Proof of Theorem 1

To proof Theorem 1 we need several intermediate results. Lemma 6 shows that a histogram results in a valid discretization as all terms corresponding to volumes in I^h cancel out, and hence I^h can be written as a sum of plug-in estimators of *discrete entropies.* Then, Lemma 4 shows a classic result that the plug-in estimator of discrete entropies will converge to the true entropy almost surely. Further, we show in Lemma 5 that as the volumes of histogram bins containing continuous values go to 0, the true entropies of discretized variables (which are discretized by the histogram) converges to the true entropy of original variables.

Definition 1. *Given discrete random variables X_d, Y_d, Z_d (possibly multi-dimensional),* *with support S_d^X, S_d^Y, S_d^Z, and given dataset $D = (x_i, y_i, z_i)_{i\in\{1,\ldots,n\}}$ with sample size n, the plug-in estimator of discrete entropy H is denoted and defined as*

$$H_n(X_d, Y_d, Z_d) = -\sum_{j\in S_d^X \times S_d^Y \times S_d^Z} \hat{p}(j) \log \hat{p}(j)$$

with probability estimates

$$\hat{p}(j) = \frac{|\{(x_i, y_i, z_i)_{i\in\{1,\ldots,n\}} : (x_i, y_i, z_i) = q_j\}|}{n} ,$$

where $|\cdot|$ represents the cardinality of a set, and q_j is the jth element in $S_d^X \times S_d^Y \times S_d^Z$.

Lemma 4. *Given a discrete random vector (X_d, Y_d, Z_d), $lim_{n\to\infty} H_n(X_d, Y_d, Z_d) = H(X_d, Y_d, Z_d)$ almost surely (Antos and Kontoyiannis 2001).*

Lemma 5. *Given a random vector (X, Y, Z) that contains discrete-continuous mixture random variables, with bins $B = B' \cup B''$ and the resulting discretized random vector (X_d, Y_d, Z_d), where B'' contains discrete data points (of which every dimension has a discrete value) and $B' = B \setminus B''$, we have*

$$\lim_{v'\to 0} H(X_d, Y_d, Z_d) = H(X, Y, Z) ,$$

where $v' = \max_{B_j\in B'}(v(B_j))$.

Proof. Firstly, it is well-known that this result holds if (X, Y, Z) is a continuous random vector (Cover and Thomas 2012); then, if (X, Y, Z) contains mixture variables,

$$H(X, Y, Z) = \lim_{v'\to 0} \sum_{B_j\in B'} \frac{P_{X_d Y_d Z_d}}{v(B_j)} \log \frac{P_{X_d Y_d Z_d}}{v(B_j)} \tag{6.12}$$

$$+ \sum_{B_j\in B''} \frac{P_{X_d Y_d Z_d}}{v(B_j)} \log \frac{P_{X_d Y_d Z_d}}{v(B_j)} \tag{6.13}$$

$$= \lim_{v'\to 0} H(X_d, Y_d, Z_d) , \tag{6.14}$$

which concludes the proof. □

Definition 2. *Given a random vector (X, Y, Z) that contains mixture variables, and an adaptive grid B, we define the discretized random variable X_d, Y_d, Z_d, with* *probability measure (probability mass function)*

$$P_{X_d,Y_d,Z_d}((j_1,j_2,j_3)) = \int_{B_j} \frac{d_{XYZ}}{dv} dv ,$$

where B_j denotes the jth bin of B.

Lemma 6. *Given a k-dimensional random vector (X,Y,Z) that contains mixture variables with an unknown probability measure P_{XYZ}, a dataset $D = (x_i, y_i, z_i)_{i\in\{1,\dots,n\}}$ generated by P_{XYZ}, a histogram model M, and corresponding discretized random vector (X_d, Y_d, Z_d), we have $I^h(X,Y|Z)$ is equal to*

$$H_n(X_d, Z_d) + H_n(Y_d, Z_d) - H_n(X_d, Y_d, Z_d) - H_n(Z_d) .$$

That is, the terms corresponding to volumes in I^h cancel out and our histogram model results a valid discretization.

Proof. Denote the adaptive grid of histogram model M as B^{XYZ}, which is the Cartesian product of bins defined on X,Y,Z—i.e. $B^{XYZ} = B^X \times B^Y \times B^Z$, and denote the corresponding MLE of histogram density function as $f^h_{\hat{\theta}_{XYZ}}$. Further, define a function v_X, such that for each x_i in D, $v_X(x_i) = v(B^X_j)$ if $x_i \in B^X_j$, where B^X_j is a bin of B^X and v is defined based on the random variable X. Then, define $v_Y, v_Z, v_{XZ}, v_{YZ}, v_{XYZ}$ similarly.

By the definition $I^h(X,Y|Z)$ is equal to

$$H^h(X,Z) + H^h(Y,Z) - H^h(X,Y,Z) - H^h(Z) .$$

First consider $H^h(X,Z)$. We write $B^{XZ} = B^X \times B^Z$, with marginal density function $f^h_{\hat{\theta}_{XZ}}$. W.l.o.g. suppose that B_{XZ} consists of K bins, denoted as $B^{XZ}_j, j \in$

$\{1, \ldots, K\}$. Then,

$$\begin{aligned} H^h(X,Z) &= -\int_{\mathbb{R}^{k_X+k_Z}} f^h_{\hat{\theta}_{XZ}} \log f^h_{\hat{\theta}_{XZ}} dv \\ &= -\sum_{j=1}^{K} \int_{B_j^{XZ}} f^h_{\hat{\theta}_{XZ}} \log f^h_{\hat{\theta}_{XZ}} dv \\ &= -\sum_{j=1}^{K} c_j \log \left(\frac{c_j}{nv(B_j)} \right) \\ &= -\sum_{j=1}^{K} c_j \log \left(\frac{c_j}{n} \right) + \sum_{i=1}^{n} \log(v_{XZ}(x_i, z_i)) \\ &= H_n(X_d, Z_d) + \sum_{i=1}^{n} \log(v_{XZ}(x_i, z_i)) \ , \end{aligned} \tag{6.15}$$

where c_j is the number of data points in B_j and $v_{XZ}(x_i, z_i) = v_X(x_i)v_Z(z_i)$. The remaining entropies can be calculated similarly. Hence, $I^h(X, Y|Z) = H_n(X_d, Z_d) + H_n(Y_d, Z_d) - H_n(X_d, Y_d, Z_d) - H_n(Z_d)$, as the sum of the volume related terms

$$\sum_{i=1}^{n} \log(v_{XZ}(x_i, z_i)) + \sum_{i=1}^{n} \log(v_{YZ}(y_i, z_i)) \tag{6.16}$$

$$-\sum_{i=1}^{n} \log(v_{XYZ}(x_i, y_i, z_i)) - \sum_{i=1}^{n} \log(v_Z(z_i)) \tag{6.17}$$

is equal to zero. □

To proof Theorem 1, we link the above results:

$$\begin{aligned} &\lim_{v' \to 0} \lim_{n \to \infty} I^h(X; Y \mid Z) \\ =& \lim_{v' \to 0} \lim_{n \to \infty} (H^h(X,Z) + H^h(Y,Z) - H^h(X,Y,Z) - H^h(Z)) \\ =& \lim_{v' \to 0} \lim_{n \to \infty} (H_n(X_d, Z_d) + H_n(Y_d, Z_d) - \\ & H_n(X_d, Y_d, Z_d) - H_n(Z_d)) \\ =& \lim_{v' \to 0} (H(X_d, Z_d) + H(Y_d, Z_d) - H(X_d, Y_d, Z_d) - H(Z_d)) \\ =& I(X; Y \mid Z) \ . \end{aligned} \tag{6.18}$$

6.9.2 Implementation Details

As discussed in Section 6.5, our goal is to find that discretization that minimizes the joint entropy over a set of k random variables via an iterative greedy algorithm. We provide the pseudocode in Algorithm 6. As input, we are given a dataset $D = \{D^1, \ldots, D^k\}$ consisting of n rows and k columns, representing a sample of size n from a k-dimensional random vector X, and a user-specified parameter $i_{\max}$ specifying the maximum number of iterations. First, we initialize the discretization X_d (line 1) by creating single bin histograms for the continuous points in D_j and a bin with bin-width 1 per discrete point. To detect the latter, we check if there exist $|\{x \in \mathcal{X}_j \mid D_j = x\}| \geq t$, where t is a user-defined threshold. After that, we iteratively update the discretization for that X_j providing the highest gain in stochastic complexity, until either the score cannot be improved or the maximum number of iterations has been reached (lines 3 – 13). To update the discretization of a variable X_j we call the function refine (line 6), which receives as input the data D_j and the discretization after iteration i. It then re-discretizes X_j using an extension of the dynamic programming algorithm by Kontkanen and Myllymäki (2007b). In essence, instead of simply discretizing X_j independently of the remaining variables, we keep the discretizations for all $X_i \neq X_j$ fixed and find the optimal histogram model M^* over X_j s.t. the overall score $L(D, M)$ is minimized.

6.9.3 Data Generation and Additional Experiments

In the following, we first provide an empirical analysis on how the number of bins depends on the number of samples, then we give the details of the data generation for the experiments carried out on the synthetic causal network and last we provide additional experiments to evaluate $I_{\mathcal{X}^2}$ and I_{SC}.

Sample Size and Number of Bins

As discussed in Section 6.3, an important requirement to ensure consistency is that the number of bins grows as a sub-linear function w.r.t. the number of samples. We demonstrate that MDL-optimal histograms have this desirable property when learned on one-dimensional Gaussian distributions in Figure 6.5: the number of bins K grows with n, but slower than $\sqrt{n}$. In addition, for multi-

Algorithm 6: Discretization of a Mixture Model

```
Input: Data $D = \{D_1, \dots, D_k\}$ representing a sample from a
       $k$-dimensional random vector $X$, maximum number of iterations
       $i_{\max}$
Output: Discretization $X_d$
$X_d \leftarrow$ init($D$)                    ▷ Initialize the discretization;
$i \leftarrow 1$;
while $X_d$ changes $\wedge\ i \leq i_{max}$ do
    $X_d^i \leftarrow X_d$;
    for $j \in \{1, \dots, k\}$ do
        $X_d^{ij} \leftarrow$ refine($D_j \mid X_d$)          ▷ Refine discretization;
        if $Score(X_d^{ij}) < Score(X_d^i)$ then
            $X_d^i \leftarrow X_d^{ij}$;
    $X_d \leftarrow X_d^i$;
    $i \leftarrow i + 1$;
return $X_d$                    ▷ Return final discretization;
```

dimensional data, for which we can only approximate the histogram model that minimizes $L(D, M)$, we observe that if the number of dimensions increases, the average number of bins per dimension decreases if we keep n fixed.

Synthetic Network

Here, we describe the data generation for the synthetic network shown in Figure 6.3. The source nodes of the network are A and B. A is generated as $A \sim \text{Exp}(1)$ and $B \sim \text{Unif}(0, 4)$ (discrete). To get $B \rightarrow C$ we generate C as $C \sim \text{Binom}(b, 0.5)$ for $B = b$, for $B \rightarrow D$ we sample D as $D \sim N(b-2, 1)$ for $B = b$ and E is sampled is exponentially distributed with rate $\frac{1}{c+1}$ for $C = c$. F is generated as a function of C and D. First, we generate C' by rounding the values of C and then we write F as $F = D^{\frac{C'}{2}} + N(0, 1)$. Last, we generate G as the zero inflated Poissonization of A. Let $E' = \frac{\text{sign}(E-1)+1}{2}$, which ensures that E' is either zero or one dependent on the value of E. Then $G \sim N(a, 1)$ if $E' = 0$ and $A = a$, and $G \sim \text{Poisson}(a)$ for $A = a$ if $G = 1$.

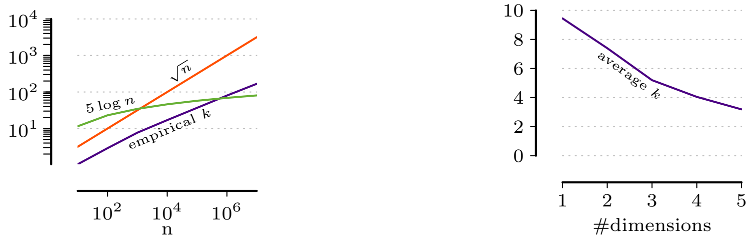

Figure 6.5: Left: Average number of bins k to discretize $X \sim N(0,1)$ for increasing sample sizes (20 repetitions). Right: Per dimension of a multivariate Gaussian distribution with $X_i \perp\!\!\!\perp X_j$ and $X_i \sim N(0,1)$, we show the average number of bins ($n = 2\,000$, 20 repetitions).

Detecting Collider and Non-Collider Structures

To evaluate how well $I_{\mathcal{X}^2}$ and I_{SC} can identify conditional (in)dependencies, we evaluate both variants on various generating mechanisms that involve collider and non-collider structures. Those structures are at the core of causal discovery, since collider structures can be inferred by detecting conditional dependencies, while non-collider structures impose conditional independencies. As in the causal discovery experiment, we set $\alpha = 0.01$ for $I_{\mathcal{X}^2}$, RCIT and RCoT.

Collider Structures We generate data according to a collider structure, which can be represented by a directed acyclic graph as, e.g., $X \to Z \leftarrow Y$. According to this structure, we model X and Y by some distribution and write Z as a non-deterministic function of X and Y. We generate data for different generating mechanisms, including two continuous and four mixed settings.

1. $X \perp\!\!\!\perp Y$ and X, Y are either drawn from $N(0,1)$ or Uniform$(-2,2)$. Z is an additive function of polynomials up to degree three or the tangent function plus additive noise $N \sim N(0, 0.1)$—e.g. $Z = X^3 + \tan(Y) + N$. We pick the type of the distribution of X, Y, as well as the function type, uniform at random.

2. X, Y are drawn from a standard Gaussian distribution, with $X \perp\!\!\!\perp Y$ and $Z = \text{sign}(X \cdot Y) \cdot \text{Exp}(1/\sqrt{2})$.

3. $X, Y \sim N(0,1)$ with $X \perp\!\!\!\perp Y$ and $Z = \text{sign}(X \cdot Y)$, where we randomly assign a $z \in \mathcal{Z}$ to 10% of the values in Z to make the function non-deterministic.

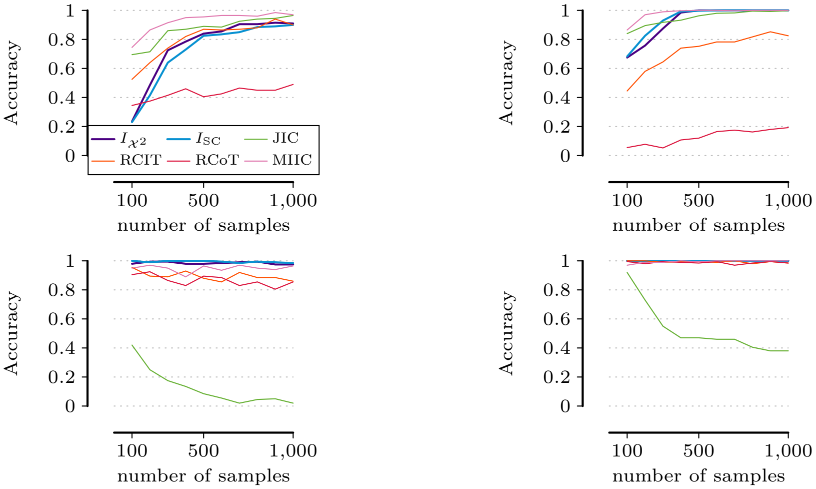

Figure 6.6: Accuracy for detecting continuous (left) and mixed-type (right) dependencies in collider structures (top) and independencies in non-collider structures (bottom) for different sample sizes.

4. $X \sim N(0, 1)$, $Y \sim \text{Poisson}(\lambda)$, with parameter λ selected uniformly at random from $\{1, 2, 3\}$. We generate Z as X modulo Y and assign 10% of the data points randomly.

5. X, Y are unbiased coins. $Z' = X \oplus Y$, where $\oplus$ denotes the xor operator. From Z' we calculate Z as $N(0, 0.1)$ if $Z' = 0$ and $\text{Poisson}(5) \cdot N(0, 0.1)$ under the condition that $Z' = 1$.

6. We generate X, Y and Z' as above, but this time we generate Z as $\text{Poisson}(5)+$ $N(0, 0.1)$ if $Z' = 1$ and as $N(0, 0.1)$ if $Z' = 0$.

For each generating mechanism, including two purely continuous and four mixed mechanisms, we generate 100 data sets and report the averaged results, separately for the continuous and mixed data, in Figure 6.6 (top). On the continuous data, both of our approaches perform on par with RCIT and JIC for more than 400 data points, whereas MIIC has a slightly better performance and RCoT is not able to detect the dependence for the sign function and hence has an accuracy of about 50%. Since the functions for mixed data include an xor and the modulo operator, it is difficult to treat all discrete variables as ordinal and hence RCIT only reaches up to 80% accuracy—which is mostly due to an xor determining the scaling of a Gaussian distributed variable. On the other hand, both of our tests perform very well and only need 400 samples to obtain an accuracy close to 100%. JIC and MIIC perform on par with our tests.

Non-Collider Structures Similar to collider structures, there also exist non-collider structures of the form $X \to Z \to Y$ or $X \leftarrow Z \to Y$. In both cases, the ground truth is that $X \perp\!\!\!\perp Y \mid Z$. To simulate data according to these graphs, we consider two continuous mechanisms based on polynomial functions and two mixed generating mechanisms.

1. $X \sim N(0, 1)$, Z is an additive noise function of X and Y is an additive noise function of Z. The functions can be polynomials up to degree three or the tangent function.
2. $Z \sim N(0, 1)$, X and Y are independent additive noise functions of Z, as defined above.
3. X, Y and Z are generated as in Experiment IV.
4. X and Y are generated according to Experiment II and $Z \sim N(\mu, x)$ for $X = x$ and $\mu \in [-4, 4]$.

In essence, Figure 6.6 (bottom) shows that both our tests obtain almost perfect accuracies for the continuous and mixed data, whereas RCIT and RCoT fail to detect up to 20% of the independencies for continuous data, MIIC does not detect up to 11% and JIC seems to generally overestimate dependencies for those test cases. If we consider these results in comparison to the results for detecting dependencies for the collider setting, we suspect that both MIIC and JIC have a larger tendency to falsely detect dependencies, while our approach is more conservative and hence needs more samples to detect true dependencies.