Discovering Functional Dependencies from Mixed-Type Data

Panagiotis Mandros
pmandros@mpi-inf.mpg.de
Max Planck Institute for Informatics
Saarbrücken, Germany

David Kaltenpoth
david.kaltenpoth@cispa.saarland
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany

Mario Boley
mario.boley@monash.edu
Monash University
Melbourne, Australia

Jilles Vreeken
jv@cispa.saarland
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany

ABSTRACT

Given complex data collections, practitioners can perform non-parametric functional dependency discovery (FDD) to uncover relationships between variables that were previously unknown. However, known FDD methods are applicable to nominal data, and in practice non-nominal variables are discretized, e.g., in a pre-processing step. This is problematic because, as soon as a mix of discrete and continuous variables is involved, the interaction of discretization with the various dependency measures from the literature is poorly understood. In particular, it is unclear whether a given discretization method even leads to a consistent dependency estimate. In this paper, we analyze these fundamental questions and derive formal criteria as to when a discretization process applied to a mixed set of random variables leads to consistent estimates of mutual information. With these insights, we derive an estimator framework applicable to any task that involves estimating mutual information from multivariate and mixed-type data. Last, we extend with this framework a previously proposed FDD approach for reliable dependencies. Experimental evaluation shows that the derived reliable estimator is both computationally and statistically efficient, and leads to effective FDD algorithms for mixed-type data.

CCS CONCEPTS

• **Information systems → Data mining**.

KEYWORDS

mutual information, functional dependency discovery, mixed data

ACM Reference Format:
Panagiotis Mandros, David Kaltenpoth, Mario Boley, and Jilles Vreeken. 2020. Discovering Functional Dependencies from Mixed-Type Data. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3394486.3403193

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403193

1 INTRODUCTION

Scientific discovery and similar applications [10, 17] constantly produce high-dimensional data collections of mixed variable types (i.e., nominal, ordinal, continuous). To uncover previously unknown relationships in such complex data collections, practitioners perform **exploratory data analysis (EDA)** [21]. Non-parametric **functional dependency discovery (FDD)** is a well-suited EDA approach, as there are no a priori assumptions about both the type of variables involved and the form of the relationship (e.g., XOR, non-linear).

Formally, given data $\mathbf{D}$ sampled according to the joint distribution $p(\mathcal{I}, \mathcal{Y})$ of explanatory variables $\mathcal{I} = \{X_1, \ldots, X_d\}$ and target variables $\mathcal{Y}$ that can be of any type, we are interested in identifying those variable sets $\mathcal{X} \subseteq \mathcal{I}$ with which $\mathcal{Y}$ can be described most accurately via some unknown function f, i.e., $\mathcal{Y} \approx f(\mathcal{X})$. To effectively solve the FDD problem, we require a dependency measure $D(\mathcal{X}; \mathcal{Y})$ acting as a proxy for the strength of a potential relationship $\mathcal{Y} \approx f(\mathcal{X})$, and a search algorithm that is guaranteed to (approximately) find those variable sets that maximize $D(\mathcal{X}; \mathcal{Y})$. The dependency measure D should capture any type of relationship, and it should do so for high-dimensional $\mathcal{X}$ comprising of any variable type. The **mutual information** $I(\mathcal{X}; \mathcal{Y})$ satisfies this requirement: measuring the divergence between $p(\mathcal{X}, \mathcal{Y})$ and $p(\mathcal{X})p(\mathcal{Y})$, it captures any type of relationship, while it naturally accounts for multivariate and mixed random variable sets. For FDD in particular, where the target $\mathcal{Y}$ is fixed, normalizing with the **entropy** $H(\mathcal{Y})$ gives rise to the **fraction of information** $F(\mathcal{X}; \mathcal{Y}) = I(\mathcal{X}; \mathcal{Y})/H(\mathcal{Y})$, an interpretable score in $[0, 1]$ quantifying the proportional reduction of uncertainty of $\mathcal{Y}$ by knowing $\mathcal{X}$.

Since we lack access to the distribution $p(\mathcal{I}, \mathcal{Y})$, in practice we estimate mutual information from the data $\mathbf{D}$. This creates a two-fold estimation problem. First, instead of directly considering the underlying continuous variables, we have to resort to their approximations from either data-based discretization or density estimation. Secondly, even for discrete data, we cannot measure the underlying mutual information but instead rely again on the data to obtain an estimate $\hat{I}(\mathcal{X}, \mathcal{Y})$ with some estimator $\hat{I}$. These two are more profound for FDD, where we have to efficiently identify the strongest and most reliable dependencies by comparing $\hat{I}(\mathcal{X}; \mathcal{Y})$ for all possible $\mathcal{X} \subseteq \mathcal{I}$. While efficiently discovering reliable dependencies has been principally addressed [15, 16], it remains unclear with what quantization methods it can be combined such that the search consistently identifies the strongest dependencies.

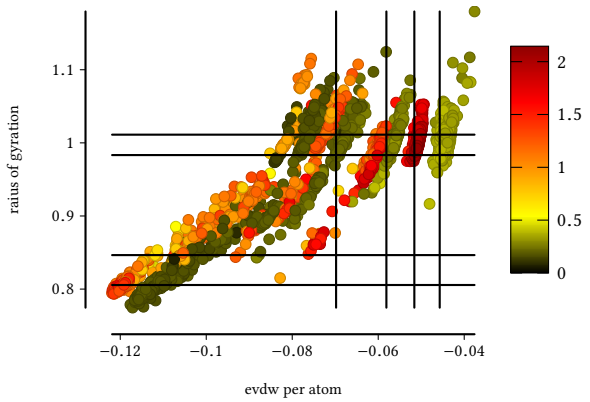


Figure 1: Top dependency discovered for a case study on nano-clusters [9]. The target variable HOMO-LUMO gap determines the electro-chemical properties of a cluster. Our proposed FDD method uncovers that structural feature "radius of gyration" and non-local dispersion energies "evdw per atom" approximately determine the target with $\hat{F}_0$ score 0.43. Black lines represent the resulting partition in $\mathbb{R}^2$ with a budget of up to 5 bins per axis (cOP, $l = 5, c = 2$).

Our main contributions are the following. First, to arrive at a consistent mixed estimator $\hat{I}_{\text{MX}}$, we recall that mutual information for two continuous random variables can be attained as a limit along a refining quantization sequence [3, Sec. 8.3]. We extend this result for mixed sets of variables, as well as identify the class of quantizations applicable that includes known techniques such as equal-frequency. We then translate this process for an empirical sample, and identify the requirements for consistency (Sec. 3). Second, based on the theory developed we propose a framework for mixed mutual information estimation and demonstrate how it can be applied in practice for FDD (Sec. 4). Third, we combine the mixed estimator with the framework of Mandros et al. [15, 16] for reliable FDD. In particular, we show that the reliable mutual information estimator is well-suited for the mixed estimator framework, and provide effective algorithms for exact, approximate, and heuristic search (Sec. 5, see Fig. 1 for a demo). Lastly, we perform extensive evaluation on a wide range of real and synthetic data (Sec. 6). We start with preliminaries in Sec. 2, and end with related work, discussion, and conclusions in Sec. 7, 8, and 9, respectively.

2 PRELIMINARIES

We represent random variables with capital letters, and sets with curly capital letters, e.g., $\mathcal{X} = \{X_1, \ldots, X_m\}$. We denote the domain of random variables with $V(.)$, e.g., $V(\mathcal{X})$ and $V(X)$, and values with $x \in V(X)$ and $\mathbf{x} \in V(\mathcal{X})$. Whenever clear from the context we use $x \in X$ and $\mathbf{x} \in \mathcal{X}$ instead. We often use $\mathcal{D}, \mathcal{G}$, to indicate sets of discrete variables, and $\mathcal{C}$ for sets of continuous random variables.

We consider **quantization strategies** for continuous random variables which we denote with Q. Given $k \in \mathbb{Z}^+$ and a continuous random variable C, Q produces a partition $Q_k = \{S_1, \ldots, S_k\}$ of $V(C) \subseteq \mathbb{R}$ in k consecutive intervals with $\cup_{i=1}^{k} S_i = V(C)$ (upper-bound exclusive). With C_{Q_k} we represent the quantized C according to Q and k. As an example, **equal-frequency** denoted as Q^{EF}, partitions C with $Q_k^{\text{EF}} = \{S_1, \ldots, S_k\}$ such that $\int_{S_i} f_C(c)dc = 1/k$ for all $i \in [k]$, where $f_C(c)$ is the density function of C. Given Q and k, we use δ_i for the corresponding length of the sub-interval S_i. In this paper, we are interested in the class of quantization strategies for which $\max_{i\in[k]} \delta_i \to 0$ as $k \to \infty$, which we refer to as **converging** strategies. These notions extend to the multivariate case $\mathcal{C} = \{C_1, \ldots, C_m\}$, with $Q_{k^m} = \{S_1, \ldots, S_{k^m}\}$ being a partition of $V(\mathcal{C}) \subseteq \mathbb{R}^m$, produced by partitioning each $C \in \mathcal{C}$ in k bins. We use Q_k whenever clear from the context. For a Q, the set $\Pi_l(Q) = \{Q_1, \ldots, Q_l\}$ corresponds to all partitions by Q in up to l bins, and $\Pi_l^m(Q)$ to the set of all partitions for domains in $\mathbb{R}^m$.

We define the following relation for two partitions: Q'_v is a **refinement** of Q_u, denoted as $Q_u \preceq Q'_v$, if $v \geq u$ and there exists a map $r : [u] \to 2^{[v]}$, such that for every $i \in [u]$, we have $S_i = \cup_{j\in r(i)} S'_j$. For example, we have that $Q_2^{\text{EF}} = \{S_1, S_2\} \preceq Q_4^{\text{EF}} = \{S'_1, S'_2, S'_3, S'_4\}$, since $S_1 = S'_1 \cup S'_2$ and $S_2 = S'_3 \cup S'_4$.

We identify the n i.i.d samples of a random variable set $\mathcal{X}$ with the map $\mathcal{X} : [n] \to V(\mathcal{X})$, or simply $\mathcal{X}$ whenever clear from the context. Given samples, a quantization strategy Q translates to a **discretization strategy**, denoted as $\hat{Q}$, that corresponds to the same strategy to partition the n sample points X_s in k bins, where X_s is X sorted in ascending order. For example, let us consider random variable $X \sim \mathbf{U}(-1, 1)$, and a sorted sample $X = [-0.5, -0.3, 0, 0.6, 0.9, 1]$. For $k = 3$ and equal-frequency, $\hat{\pi} = \hat{Q}^{\text{EF}}$ can be seen as a map $\hat{\pi} : \mathbb{R} \to [k]$ that splits the data sample in three bins of two points each, to create discrete variable $X_{\hat{\pi}} = [1, 1, 2, 2, 3, 3]$ with domain $V(X_{\hat{\pi}}) = \{1, 2, 3\}$. With $\Pi_{l,n}$, we denote the set of all possible partitions of n data points in up to $l \leq n$ bins, and for a $\hat{Q}$, we have $\Pi_{l,n}(\hat{Q}) = \{\hat{Q}_1, \ldots, \hat{Q}_l\}$. Note that we also consider X_π for $\pi = Q_k^{\text{EF}}$, meaning that X is discetized according to the equal-frequency quantization of the population domain $V(X) = [-1, 1]$, that is, for $\pi = Q_3^{\text{EF}} = \{[-1, -1/3), [-1/3, 1/3), [1/3, 1]\}$, $X_\pi = [1, 2, 2, 3, 3, 3]$.

For n samples of a discrete random variable set $\mathcal{G}$, we define $n_{\mathcal{G}} : V(\mathcal{G}) \to \mathbb{Z}$ to be the **empirical counts** of $\mathcal{G}$, i.e., $n_{\mathcal{G}}(\mathbf{g}) = |\{i \in [n] : \mathcal{G}(i) = \mathbf{g}\}|$. However, we use $n_{\mathbf{g}}$ whenever clear from the context. We further denote with $\hat{p}_{\mathcal{G}} : V(\mathcal{G}) \to [0, 1]$, where $\hat{p}_{\mathcal{G}}(\mathbf{g}) = n_{\mathcal{G}}(\mathbf{g})/n$ is the **empirical probability distribution** of $\mathcal{G}$.

Finally, recall the notion of **dominated convergence**: let a_{mn} be a sequence such that for all m the limit $a_m^* = \lim_{n\to\infty} a_{mn}$ exists. Further, let $p_m \geq 0$ be another sequence and let $u_m \geq |a_{mn}|$ for all m, n such that $\sum_m p_m u_m < \infty$. Then the limit $\lim_{n\to\infty} \sum_m p_m a_{mn}$ exists and is equal to $\sum_m p_m a_m^*$.

3 CONSISTENCY OF MIXED MUTUAL INFORMATION ESTIMATION

In this section we introduce the information-theoretic notions of multivariate entropy and mutual information for mixtures of discrete (both nominal and ordinal) and continuous random variables. We demonstrate how the sequence of finer-grain quantizations of continuous random variables leads to the actual (i.e., unquantized) mutual information. Finally, we show how this process translates to empirical samples, enabling estimation from mixed-type data.

Given sets $\mathcal{D}$ and $\mathcal{C}$ of discrete and continuous random variables, respectively, the **entropy** of $\mathcal{D} \cup \mathcal{C}$ with joint probability

distribution $f(\mathbf{d},\mathbf{c}) = f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})p(\mathbf{d})$, is defined as

$$\begin{aligned}H(\mathcal{D},C) &= -\sum_{\mathbf{d}\in\mathcal{D}}\int_C f(\mathbf{d},\mathbf{c})\log f(\mathbf{d},\mathbf{c})d\mathbf{c}\\ &= -\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\int_C f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})\log f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})d\mathbf{c}\\ &\quad -\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\log p(\mathbf{d})\int_C f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})d\mathbf{c}\\ &= H(C\mid\mathcal{D}) + H(\mathcal{D})\ .\end{aligned}$$

Let us consider a converging Q and $Q_k = \{S_1,\ldots,S_{k^m}\}$ an m-dimensional partition of the domain $V(C)\subseteq\mathbb{R}^m$. Let us assume that $f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})$ is continuous within each hypercube for all $\mathbf{d}\in\mathcal{D}$. Then, using the mean value theorem for integrals, there exists a value $\mathbf{c}_i$ within each hypercube i such that $f_{C\mid\mathbf{d}}(\mathbf{c}_i\mid\mathbf{d})\boldsymbol{\delta}_i = \int_{S_i} f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})d\mathbf{c}$. The quantized C is defined as $C_{Q_k} = \mathbf{c}_i$ for $C\in S_i$, and has conditional probability $p_{i\mid\mathbf{d}} = \boldsymbol{\delta}_i f_{C\mid\mathbf{d}}(\mathbf{c}_i\mid\mathbf{d})$ that $C_{Q_k} = \mathbf{c}_i$ when $\mathcal{D} = \mathbf{d}$. The following Lemma demonstrates how $H(\mathcal{D},C_{Q_k})$ converges to $H(\mathcal{D},C)$.

Lemma 3.1. *Given random variables $\mathcal{D}$ of finite domain $V(\mathcal{D})$, random variables C, and converging Q, if the conditional density $f_{C\mid\boldsymbol{d}}(\mathbf{c}\mid\boldsymbol{d})$ is Riemann integrable for all $\boldsymbol{d}\in\mathcal{D}$, then*

$$\lim_{k\to\infty} H(\mathcal{D},C_{Q_k}) + \beta_{Q_k}(\mathcal{D}) = H(\mathcal{D},C)\ ,$$

where $\beta_{Q_k}(\mathcal{D}) = \sum_{\boldsymbol{d}\in\mathcal{D}} p(\boldsymbol{d})\sum_i \boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\boldsymbol{d})\log\boldsymbol{\delta}_i$. Further, if for all $\boldsymbol{d}\in\mathcal{D}$, k, we have $h_k(\boldsymbol{d}) = \left|\sum_{i=1}^k \boldsymbol{\delta}_i f(\mathbf{c}_i\mid\boldsymbol{d})\log f(\mathbf{c}_i\mid\boldsymbol{d})\right| \le a(\boldsymbol{d})$ such that $\sum_{\boldsymbol{d}} p(\boldsymbol{d})a(\boldsymbol{d}) < \infty$, the result also holds for infinite $V(\mathcal{D})$.

Proof. We write $f_C(\mathbf{c}_i\mid\mathbf{d})$ instead of $f_{C\mid\mathbf{d}}(\mathbf{c}_i\mid\mathbf{d})$. We have

$$\begin{aligned}H(\mathcal{D},C_{Q_k}) &= -\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_{i=1}^k \boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\mathbf{d})\log\left(\boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\mathbf{d})\right) + H(\mathcal{D})\\ &= -\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_{i=1}^k \boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\mathbf{d})\log f_C(\mathbf{c}_i\mid\mathbf{d})\\ &\quad \underbrace{-\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_{i=1}^k \boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\mathbf{d})\log\boldsymbol{\delta}_i}_{\beta_{Q_k}(\mathcal{D})} + H(\mathcal{D})\ .\end{aligned}$$

Since $f_{C\mid\mathbf{d}}(\mathbf{c}\mid\mathbf{d})$ is Riemann integrable, the inner sum of the first term converges to its integral as $k\to\infty$. For finite $V(\mathcal{D})$, the first sum then converges to $H(C\mid\mathcal{D})$. For infinite $V(\mathcal{D})$, the sum also converges to $H(C\mid\mathcal{D})$ as $\sum_{\mathbf{d}} p(\mathbf{d})a(\mathbf{d}) < \infty$ is the assumption required for dominated convergence. □

Lemma 3.1 states that for convergence a sequence of finer-grained quantizations and a correction by β are required. In addition, $h_k(\mathbf{d})$ have to be bounded for convergence with infinite $V(\mathcal{D})$. An example in the Appendix shows how it can fail otherwise. Note that the correction $\beta_{Q_k}(\mathcal{D})$ is necessary due to the **infinite quantization error** as $k\to\infty$. That is, as the partitions get finer, $H(\mathcal{D},C_{Q_k})$ diverges. We also note that the entropy $H(\mathcal{D},C)$, unlike the discrete case $H(\mathcal{D})$, can be negative, e.g., for $C\sim\mathbf{U}(0,a)$, $a<1$ [3, Sec. 8.1]. These, however, do not extend to mutual information.

The **mutual information** for $\mathcal{X} = \{\mathcal{D},C\}$ and $\mathcal{Y} = \{\mathcal{D}',C'\}$, is defined as $I(\mathcal{X};\mathcal{Y}) = H(\mathcal{D},C) + H(\mathcal{D}',C') - H(\mathcal{D},C,\mathcal{D}',C')$, and it holds that $I(\mathcal{X};\mathcal{Y})\ge 0$. We proceed with the following Theorem about the convergence of $I(\mathcal{X};\mathcal{Y})$ w.r.t. the quantization process.

Theorem 3.2. *Given random variables $\mathcal{X} = \{\mathcal{D},C\}$, $\mathcal{Y} = \{\mathcal{D}',C'\}$, with Riemann integrable conditional density $f_{C,C'\mid\boldsymbol{d},\boldsymbol{d}'}(\boldsymbol{c},\boldsymbol{c}'\mid\boldsymbol{d},\boldsymbol{d}')$ for all $\boldsymbol{d}\in\mathcal{D}$, $\boldsymbol{d}'\in\mathcal{D}'$, as well as converging Q,Q', then*

$$I(\mathcal{X};\mathcal{Y}) = \lim_{k\to\infty} I(\mathcal{D},C_{Q_k};\mathcal{D}',C'_{Q'_k})\ .$$

Proof. For readability, we drop k, as well as use $f_C(\mathbf{c}_i\mid\mathbf{d})$ instead of $f_{C\mid\mathbf{d}}(\mathbf{c}_i\mid\mathbf{d})$, whenever clear from the context. We have:

$$\begin{aligned}&I(\mathcal{D},C_Q;\mathcal{D}',C'_{Q'}) = H(\mathcal{D},C_Q)+H(\mathcal{D}',C'_{Q'})-H(\mathcal{D},C_Q,\mathcal{D}',C'_{Q'})\\ &= -\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_i \boldsymbol{\delta}_i f_C(\mathbf{c}_i\mid\mathbf{d})\log f_C(\mathbf{c}_i\mid\mathbf{d}) + \beta_Q(\mathcal{D}) + H(\mathcal{D})\\ &\quad -\sum_{\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d}')\sum_j \boldsymbol{\delta}'_j f_{C'}(\mathbf{c}_j\mid\mathbf{d}')\log f_{C'}(\mathbf{c}_j\mid\mathbf{d}') + \beta_{Q'}(\mathcal{D}') + H(\mathcal{D}')\\ &\quad +\sum_{\mathbf{d}\in\mathcal{D},\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d},\mathbf{d}')\sum_{i,j}\boldsymbol{\delta}_i\boldsymbol{\delta}'_j f_{C,C'}(\mathbf{c}_i,\mathbf{c}_j\mid\mathbf{d},\mathbf{d}')\log f_{C,C'}(\mathbf{c}_i,\mathbf{c}_j\mid\mathbf{d},\mathbf{d}')\\ &\quad -\beta_{Q,Q'}(\mathcal{D},\mathcal{D}') - H(\mathcal{D},\mathcal{D}')\ .\end{aligned}$$

We know from Lemma 3.1 that the sums converge to $H(C\mid\mathcal{D})$, $H(C'\mid\mathcal{D}')$, and $H(C,C'\mid\mathcal{D},\mathcal{D}')$. It remains to show that $\beta_{Q,Q'}(\mathcal{D},\mathcal{D}') = \beta_Q(\mathcal{D}) + \beta_{Q'}(\mathcal{D}')$, which we postpone for the Appendix. □

Theorem 3.2 states that the unquantized $I(\mathcal{X};\mathcal{Y})$ is attained for converging Q. We now proceed to translate this quantization process for samples of $p(\mathcal{X},\mathcal{Y})$, enabling the estimation from mixed data in practice. For this, we use consistent discrete estimators $\hat{H}$ for entropy H and their corresponding sampling complexities $S_{\hat{H}}$.

Recall that an estimator $\hat{H}$ is called **consistent** if $\hat{H}\xrightarrow{p} H$ as $n\to\infty$. For entropy, the **sample complexity**, i.e., the minimum sample size that achieves a certain concentration (ϵ-δ-PAC guarantee), is usually expressed as a function of the domain size. For example, the **plug-in estimator** $\hat{H}_{\text{PL}}$ defined for the empirical $\hat{p}$, i.e., $\hat{H}_{\text{PL}}(\mathcal{G}) = -\sum_{\mathbf{g}\in\mathcal{G}}\hat{p}(\mathbf{g})\log(\hat{p}(\mathbf{g}))$, has sampling complexity $S_{\hat{H}_{\text{PL}}}(k)\in O(k)$ where $k = |V(\mathcal{G})|$. The main idea of the following Theorem is to use consistent estimators for H and upper-bound the number of partitions per n w.r.t. their sample complexity.

Theorem 3.3. *Let $\mathcal{X} = \{\mathcal{D},C\}$, $\mathcal{Y} = \{\mathcal{D}',C'\}$ be i.i.d. samples from $p(\mathcal{X},\mathcal{Y})$, with finite $V(\mathcal{D}),V(\mathcal{D}')$ and Riemann integrable conditional densities $f(\boldsymbol{c},\boldsymbol{c}'\mid\boldsymbol{d},\boldsymbol{d}')$. Further, let Q,Q' be two converging strategies, $\hat{H}$ a consistent estimator for discrete entropy, and $g(n)$ a strictly increasing function such that $g(n)\le S^{-1}_{\hat{H}}(n)$. Then*

$$\lim_{n\to\infty}\hat{I}(\mathcal{D},C_{Q_{g(n)}};\mathcal{D}',C'_{Q'_{g(n)}}) = I(\mathcal{X},\mathcal{Y})\ .$$

Further, if $\hat{p}(\boldsymbol{d},\boldsymbol{d}')\xrightarrow{L^1} p(\boldsymbol{d},\boldsymbol{d}')$ and $\hat{H}(C_{Q_k}\mid\boldsymbol{d},\boldsymbol{d}') + \hat{H}(C'_{Q'_k}\mid\boldsymbol{d},\boldsymbol{d}')\le\alpha$ uniformly for all $\boldsymbol{d}\in V(\mathcal{D})$, $\boldsymbol{d}'\in V(\mathcal{D}')$ and $k\in\mathbb{Z}^+$, the result also holds for countably infinite $V(\mathcal{D}),V(\mathcal{D}')$.

Proof. We drop subscripts from Q,Q' for readability. Note that the latter two assumptions are implied for finite $V(\mathcal{D}),V(\mathcal{D}')$, and hence, we prove the more general statement. We have

$$\hat{I}(\mathcal{X};\mathcal{Y}) = \hat{H}(\mathcal{D},C_Q) + \hat{H}(\mathcal{D}',C'_{Q'}) - \hat{H}(\mathcal{D},C_Q,\mathcal{D}',C'_{Q'})\ .$$

Now, let us focus on the first term, i.e., $\hat{H}(\mathcal{D}, C_Q) = \hat{H}(C_Q \mid \mathcal{D}) + \hat{H}(\mathcal{D})$. For $\hat{H}(\mathcal{D})$, we know it converges due to the consistency of $\hat{H}$. For $\hat{H}(C_Q \mid \mathcal{D})$, we have

$$\begin{aligned}\hat{H}(C_Q \mid \mathcal{D}) &= \sum_{\mathbf{d}\in\mathcal{D}} \hat{p}(\mathbf{d})\hat{H}(C_Q \mid \mathcal{D} = \mathbf{d}) \\ &= \sum_{\mathbf{d}\in\mathcal{D}} \big(\hat{p}(\mathbf{d}) - p(\mathbf{d})\big)\hat{H}(C_Q \mid \mathcal{D} = \mathbf{d}) \\ &\quad + \sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\hat{H}(C_Q \mid \mathcal{D} = \mathbf{d}) \ .\end{aligned}$$

Since all $\hat{H}(C_{\hat{Q}} \mid \mathcal{D} = \mathbf{d}) \leq \alpha$ are bounded and $\hat{p} \xrightarrow{L^1} p$, the first sum converges to zero. For the second sum, we have that $\lim_{n\to\infty} \hat{H}(C_{\hat{Q}} \mid \mathcal{D} = \mathbf{d}) = H(C \mid \mathcal{D} = \mathbf{d})$ due to the additional assumption for $\hat{H}$. Hence, the complete sum converges to $H(C \mid \mathcal{D})$ as the conditions for dominated convergence apply. Analogous arguments for the remaining entropy terms establish the result. □

Theorem 3.3 states the two requirements for convergence to $I(\mathcal{X}; \mathcal{Y})$ given i.i.d. samples: converging quantization strategies and consistent discrete estimators for entropy. To end this section, we make the following **two remarks**. In exploratory scenarios with no access to p, a Q that partitions the variable domain is not directly applicable. Instead, we use the empirical $\hat{Q}$. Note, however, that for EF we have $\hat{Q}_k \xrightarrow{n\to\infty} Q_k$. In the remainder of this paper we remove hat symbols for $\hat{Q}$. For the second remark, while the necessary and sufficient requirements for consistent entropy estimation is $S_{\hat{H}}(k) \in \Omega(k/\log(k))$ (see [11] for an excellent review on the topic), in this paper we study "slower" estimators of the form $\hat{I}_{\text{PL}} + b(n)$, with $b(n) \xrightarrow{n\to\infty} 0$. The reason is that these estimators are more flexible w.r.t. the FDD task, e.g., $b(n)$ directly penalizes data sparsity and optimization algorithms have been provided. In the next section, we derive a mutual information estimator for mixed variables.

4 MIXED DATA ESTIMATOR FOR FDD

We start by noting that attaining the population value $I(\mathcal{X}; \mathcal{Y})$ via quantization, can be equivalently formulated as a supremum over all finite partitions of the domains $V(C), V(C')$, regardless of Q and k. Translating this to a sample, is an estimator of the form

$$\max_{\boldsymbol{\pi}\in\Pi_{l,n}^{|C|}, \boldsymbol{\pi}'\in\Pi_{l,n}^{|C'|}} \hat{I}(\mathcal{D}, C_{\boldsymbol{\pi}}; \mathcal{D}', C'_{\boldsymbol{\pi}'}) \ ,$$

i.e., the optimization problem of maximizing a discrete consistent estimator $\hat{I}$ over the set of all possible partitions $\Pi_{l,n}^{|C|}$ and $\Pi_{l,n}^{|C'|}$, with $l \in \mathbb{Z}^+$ being the maximum number of bins. For our FDD purposes, we consider the mutual information $I(\mathcal{X}; Y)$ between $\mathcal{X} = \{\mathcal{D}, C\}$ and a univariate discrete target Y, i.e., the case

$$\hat{I}_{\text{MX}}(\mathcal{X}, Y) = \max_{\boldsymbol{\pi}\in\Pi_{l,n}^{|C|}} \hat{I}(\mathcal{D}, C_{\boldsymbol{\pi}}; Y) \ .$$

This optimization problem, however, is infeasible in practice: the search space is prohibitively large with $|C| \sum_{i=0}^{l} \binom{n-1}{i}$ possible $|C|$-dimensional partitions $\boldsymbol{\pi}$ in up to l bins. Moreover, while estimators $\hat{I}$ are consistent, they can be statistically inefficient for limited data samples and almost trivially produce arbitrary partitions and estimates due to data sparsity in the $|\mathcal{X}|$-dimensional space [15, 22]. We present solutions for both problems, starting with the former.

4.1 Optimization

First, let us assume $|C| = m$, and reformulate the problem. Instead of directly searching for high-dimensional partitions $\boldsymbol{\pi} \in \Pi_{l,n}^{m}$, we can equivalently search for m univariate partitions, i.e.,

$$\max_{\pi_1,\ldots,\pi_m\in\Pi_{l,n}} \hat{I}(\mathcal{D}, \{C_{1\pi_1}, \ldots, C_{m\pi_m}\}; Y) \ .$$

This approach allows us to consider the abundant research on partitioning the real line $\mathbb{R}$. Here, we provide two solutions from prior work on dependency estimation. The first has been used for an exact solution, while the second for an approximate. Let us focus for now on the univariate continuous case, i.e., $\mathcal{X} = \{C\}$.

For an exact solution, note that a naive algorithm would perform exhaustive search through all $\sum_{i=0}^{l} \binom{n-1}{i}$ partitions for C. However, Reshef et al. in seminal work on dependency estimation for pairs of continuous variables [21], give a polynomial time algorithm for the plug-in $\hat{I}_{\text{PL}}$, exploiting the optimal substructure of $\max_{\pi\in\Pi_{l,n}} \hat{I}_{\text{PL}}(C_\pi; Y)$: the best partition in up to l bins is comprised of the best partition in up to $l-1$ bins. The dynamic programming (**DP**) algorithm has complexity $O(ln^2)$. For efficiency, the authors propose a relaxation where C is partitioned in l equal-frequency bins, and DP finds the best partition from $\{\pi : \pi \preceq Q_l^{\text{EF}}\}$. For more candidate partitions, a parameter $c \in \mathbb{Z}^+$ controls the number of initial bins via cl (see [21, Sup. material, Sec. 3.2.2]). The complexity now is $O(c^2l^3)$, and we refer to this partitioning scheme as **constrained optimal-partition** (**cOP**), with $\Pi_{l,n}(Q^{\text{cOP}}) = \{\pi : \pi \preceq Q_{cl}^{\text{EF}}, |\pi| \leq l\}$ for parameter c. For $cl = n$, cOP becomes optimal. The approximate technique is based on **equal-frequency**. To find an appropriate partition for estimating mutual information from pairs of discrete/continuous random variables, Suzuki suggests to pick the equal-frequency partition that maximizes mutual information in up to $l = 0.5\log_2(n)$ bins, i.e., $\max_{k\in[l]} \hat{I}(C_{Q_k^{\text{EF}}}; Y)$ [26]. Sugiyama and Borgwardt perform the same process in order to estimate the information dimension of a continuous variable, with $l = \log_2(n)$ [25]. For Q^{EF}, we have $\Pi_{l,n}(Q^{\text{EF}}) = \{\pi : \pi = Q_k^{\text{EF}}, k \in [l]\}$. Regarding the two techniques, cOP has a clear advantage: a larger space of candidate partitions controlled by parameters based on the availability of resources. However, EF has the negligible complexity of $O(l)$. In addition, EF is applicable to any estimator $\hat{I}$, while cOP requires optimal substructure for the polynomial DP algorithm.

Now given set $\mathcal{X} = \{\mathcal{D}, C\}$, in order to perform a multidimensional discretization in practice, we adopt a greedy approach of **iteratively discretizing** one $C \in C$ at a time. Note that while this approach is greedy in nature, the choice for a partition is done jointly with all the already discrete and discretized variables. In addition, the consistency is not violated for $k, n \to \infty$. Since the result now depends on the order, we first sort the variables in $X \in \mathcal{X}$ in decreasing order of **marginal mutual information** $\hat{I}(X; Y)$; variables $C \in C$ are discretized according to Q and l. That way, we let the most informative continuous variables discretize first, jointly with the already discrete. The details of our **proposed mixed estimator framework** are shown in Alg. 1. Given set of mixed random variables $\mathcal{X} = \{\mathcal{D}, C\}$, discrete target Y, partitioning strategy Q,

Algorithm 1 $\hat{I}_{\text{MX}}$: Given set of mixed random variables $\mathcal{X} = \{\mathcal{D}, C\}$, discrete target Y, partitioning strategy Q, consistent discrete estimator $\hat{I}$, and maximum number of bins l, the algorithm returns an estimate of $I(\mathcal{X}; Y)$

```
function Î_MX(X, Y, Q, Î, l)
    X' = sortMarginally(X, Y, Q, Î, l)
    G = ∅
    for X ∈ X' do
        if X ∈ C then
            π* = arg max{π : Î(G, X_π; Y), π ∈ Π_{l,n}(Q)}
            G = G ∪ {X_{π*}}
        else
            G = G ∪ {X}
    return Î(G; Y)
```

consistent discrete estimator $\hat{I}$, and maximum number of bins l, the estimation process starts by marginally sorting the $X \in \mathcal{X}$ according to $Q, l, \hat{I}$ (Q, l, are used for $X \in C$), and create the empty set $\mathcal{G}$ for discrete variables. Then, continuous variables $X \in C$ are discretized jointly with $\mathcal{G}$ and added to $\mathcal{G}$, while the discrete $X \in \mathcal{D}$ are added to $\mathcal{G}$. The mixed estimator result is $\hat{I}(\mathcal{G}; Y)$. If T_Q is the cost for optimization based on Q, $T_{\hat{I}}$ the cost of estimator $\hat{I}$, and $|C| = m$, the algorithm complexity is dominated by $O(mT_Q T_{\hat{I}})$. For the remainder, we refer to a specific instantiation of Alg. 1 with the estimator and partitioning technique choices, e.g.,, $\hat{I}_{\text{PL}}$ with EF.

4.2 Statistical efficiency

Now that an optimization framework is established, we shift our attention to a brief discussion regarding appropriate qualities discrete consistent estimators should possess for the task of FDD.

We are mainly after estimators that allow for efficient discovery, i.e., come with the means for high-dimensional exhaustive and heuristic search. An optional, yet important requirement, is admitting optimal substructure for applying DP and giving access to a large set of candidate partitions in polynomial time. The third dimension, is that of statistical efficiency: the estimator should give robust estimation from limited data samples for both the **partitioning process**, as well as the **discovery process**. Let us mainly focus on the last requirement, and demonstrate how the consistency of an estimator, alone, does not satisfy it. As an example, we consider the plug-in estimator $\hat{I}_{\text{PL}}$ and start with the following Lemma.

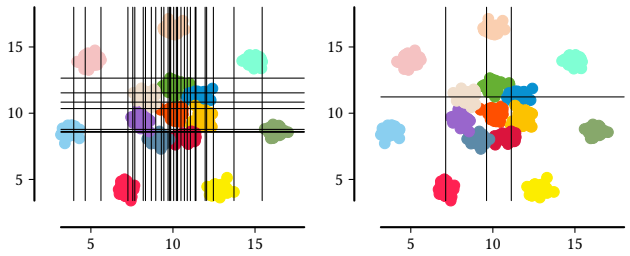
Lemma 4.1. *Given continuous variable C, discrete set $\mathcal{G}$, discrete target Y, and maximum number of bins l, we have that*

(1) $\hat{I}_{PL}(\mathcal{G}, C_\pi; Y) \leq \hat{I}_{PL}(\mathcal{G}, C_{\pi'}; Y)$, *for all* $\pi, \pi' \subseteq \Pi_{l,n}$ *with* $\pi \preceq \pi'$

(2) $\hat{I}_{PL}(\mathcal{G}, C_{Q_k^{EF}}; Y) \leq \hat{I}_{PL}(\mathcal{G}, C_{Q_{2k}^{EF}}; Y)$ *for* $k = 1, \ldots, \lfloor l/2 \rfloor$

(3) $\hat{I}_{PL}(\mathcal{G}, C_\pi; Y) \leq \hat{I}_{PL}(\mathcal{G}, C_{Q_l^{EF}}; Y)$, *for all* $\pi \in \Pi_{l,n}(Q^{cOP})$

Proof. Recall the specialization relation [16, Def. 1]: for two discrete variables A, B, we say that B is a specialization of A, denoted as $A \preceq B$, if for all $i, j \in [n]$ with $A(i) \neq A(j)$, it holds $B(i) \neq B(j)$. It is clear that a refinement relation for $\pi \preceq \pi'$, corresponds to a specialization relation for $C_\pi \preceq C_{\pi'}$. Finally, we have that for three variables A, B, C with $A \preceq B$, that $\hat{I}_{\text{PL}}(A; C) \leq \hat{I}_{\text{PL}}(B; C)$ [16, Prop. 2].



(a) $\hat{I}_{\text{PL}}$, $l = 30$, $c = 3$ **(b)** $\hat{I}_{\text{MIC}}$, $c = 3$

Figure 2: Resulting partitions on a clustering dataset for plug-in $\hat{I}_{\text{PL}}$ combined with two versions of OP (Example 4.2)

For (1), we have that $C_\pi \preceq C_{\pi'}$ for any $\pi \preceq \pi'$, and hence $\hat{I}_{\text{PL}}(\mathcal{G}, C_\pi; Y) \leq \hat{I}_{\text{PL}}(\mathcal{G}, C_{\pi'}; Y)$. For (2) and (3), we have that $Q_k^{\text{EF}} \preceq Q_{2k}^{\text{EF}}$ for $k = 1, \ldots, \lfloor l/2 \rfloor$, and $\pi \preceq Q_l^{\text{EF}}$ for all $\pi \in \Pi_{l,n}(Q^{\text{cOP}})$, respectively. The two statements then follow from (1). □

Lemma 4.1 states that $\hat{I}_{\text{PL}}$ considers refinements to be at least as good of a choice. However, unlike the quantization process in the population, refined partitions on a sample do not necessarily lead to a better estimation error, but rather to over-fitting. For $\hat{I}_{\text{PL}}$ in particular, the over-fitting is controlled by the statistical bias that is a function of the domain sizes $V(\{\mathcal{G}, C_\pi\})$ and $V(Y)$ [23]. In a nutshell, larger $|\pi|$ implies more bias for $\hat{I}_{\text{PL}}(\mathcal{G}, C_\pi; Y)$, and hence $\hat{I}_{\text{PL}}$ tends to trivially select the most refined partition for a Q and l. We demonstrate this with the following example.

Example 4.2. In this example we investigate the resulting partitions from estimating mutual information on a clustering dataset in $\mathbb{R}^2$, where the target variable Y is the cluster assignment. The dataset has 600 data points and 15 clusters [5], and we use C_1, C_2, to refer to x and y-axis, respectively. We are after an estimate of $\hat{I}_{\text{PL}}(C_1, C_2; Y)$, and consider two versions of cOP. For the first, we use a fixed $l = 30$ for both C_i, while for the second we use $l = g(n, b, \mathcal{G}, Y)$ per C_i that is proposed in [21], where $g(n, b, \mathcal{G}, Y) = \left\lceil n^b / (\prod_{G \in \mathcal{G}} V(G) V(Y)) \right\rceil$. We set $b = 0.6$ that is suggested by the authors, and refer to the resulting estimator as $\hat{I}_{\text{MIC}}$. For both we use $c = 3$. We present the results in Fig. 2. On the left, we observe that $\hat{I}_{\text{PL}}$ indeed selects for C_1 the most refined partition possible, i.e., Q_{30}^{EF}, as the Lemma suggested. For C_2, there are 8 bins, but only because there is a perfect cluster separation already for a total of 240 bins in $\mathbb{R}^2$. On the right, $\hat{I}_{\text{MIC}}$ has a maximum $l = 4$ for C_1, and $l = 2$ for C_2 (for C_2, $\mathcal{G}$ already contains the discrete C_1). Again, $\hat{I}_{\text{PL}}$ selects the maximum number of bins for both variables, but here we actually observe under-fitting caused by the criterion $l = g(n, b, \mathcal{G}, Y)$.

We see that $\hat{I}_{\text{PL}}$ can easily under/over fit the data during the partition process, even with more elaborate criteria for l, e.g., the $g(n, b, \mathcal{G}, Y)$ used in MIC. Note that $\hat{I}_{\text{MIC}}$ is an inherent part of MIC, as it identifies the best partition for each $k = 1, \ldots, l = g(n, \emptyset, Y)$. The best partitions are afterwards penalized by their size k, which is not a statistical adjustment accounting for the biased estimates. It is demonstrated that MIC over-fits on noisy data [12].

In addition to the partition process, we consider the task of FDD, i.e., finding the $\mathcal{X}^* \subseteq \mathcal{I}$ maximizing $F(\mathcal{X}^*; Y)$. Translating this to

our example, it would mean to identify the top clustered data out of a potentially huge candidate space of varying dimensionalities. For FDD, the $\hat{F}_{\text{PL}}$ fails by trivially considering $\mathcal{X}^* = \mathcal{I}$ to be a maximizer [15]. As we see, choosing an estimator for FDD is non-trivial: besides being "optimizable" for efficient algorithms and exhibiting optimal substructure, estimators need to be statistically efficient and robust against choices for l, Q, and varying dimensionalities. In Section. 6 we evaluate different choices for $\hat{I}$ and Q.

5 RELIABLE FDD FROM MIXED DATA

In this section, we recall the reliable mutual information estimator, show it exhibits optimal substructure, and give algorithms for FDD.

To perform FDD in high-dimensional data, Mandros et al. [15] propose a correction for the plug-in by subtracting its expected value over all possible sample permutations. Given $\mathcal{G}$ and Y, the **reliable mutual information** is defined as $\hat{I}_0(\mathcal{G}; Y) = \hat{I}_{\text{PL}}(\mathcal{G}; Y) - \mathbb{E}_0(\hat{I}_{\text{PL}}(\mathcal{G}; Y))$. Here, $\mathbb{E}_0(\hat{I}_{\text{PL}}(\mathcal{G}; Y))$ is the expected value under the **permutation model** [14, p. 214], a non-parametric independence model for contingency tables assuming fixed marginal counts. The expected value is equal to $\mathbb{E}_0(\hat{I}_{\text{PL}}(\mathcal{G}; Y)) = \sum_{\sigma \in S_n} \hat{I}_{\text{PL}}(\mathcal{G}; Y_\sigma)/n!$, where S_n denotes the symmetric group for n, i.e., the set of all permutations of $[n]$, and Y_σ denotes the Y samples permuted according to a $\sigma \in S_n$. Exploiting symmetries, this value can be computed in $O(n \max\{V(\mathcal{G}), V(Y)\})$ (see [16, 29] for the computation).

To use $\hat{I}_0$ for FDD and give access to a large space of candidate partitions in polynomial time, we show optimal substructure.

Theorem 5.1. *Given discrete variables $\mathcal{G}$, continuous X, discrete Y, and maximum number of bins l, the optimization problem*

$$\max_{\pi \in \Pi_{l,n}(Q^{cOP})} \hat{I}_0(\mathcal{G}, X_\pi; Y)$$

exhibits for $1 < l \leq m \leq n$ the optimal substructure

$$f(l, m) = \max_{1 \leq i < m} \{ \frac{i}{m} f(l-1, i) + \frac{m-i}{m} \hat{I}_0(G; Y \mid i+1, m) \} \ ,$$

where $f(l, m) = \max_{\pi \in \Pi_{l,m}} \hat{I}_0(\mathcal{G}, X_\pi; Y \mid 1, m)$, and $\hat{I}_0(. ; Y \mid u, v) = \hat{I}(. ; Y \mid u, v) + \sum_{\sigma \in S_n} \hat{I}(. ; Y_\sigma \mid u, v)/n!$, with $\hat{I}(. ; . \mid u, v)$ and $u, v \in [n], v \geq u$, the empirical mutual information restricted to data samples $\{i \in [n] \mid X_s(u) \leq X(i) \leq X_s(v)\}$.

Proof. We postpone the proof for the Appendix. □

Now that optimal substructure has been established, we shift our attention to search algorithms for FDD. That is, given a mixed set $\mathcal{I} = \{X_1, \ldots, X_d\}$ of d input variables and a discrete target variable Y, we are interested in the optimization problem

$$\hat{I}_0(\mathcal{X}^*; Y) = \max\{\hat{I}_0(\mathcal{X}; Y) \colon \mathcal{X} \subseteq \mathcal{I}\} \ ,$$

given a partitioning strategy Q. For discrete data, Mandros et al. show it is NP-Hard to solve, and propose two algorithms: exhaustive search based on branch-and-bound that comes with approximation guarantees, and heuristic search based on the standard bottom-up greedy algorithm [16]. Let us recall the two basic ingredients of these algorithms, and extend them for mixed data.

The first is the **refinement operator**, a function $r \colon \mathcal{P}(\mathcal{I}) \to \mathcal{P}(\mathcal{I})$, with $\mathcal{P}(\mathcal{I})$ the powerset of $\mathcal{I}$, which is used to non-redundantly enumerate the entire search space of candidate solutions $\mathcal{X} \subseteq \mathcal{I}$. For example, the operator corresponding to alphabetic order would be $r(\mathcal{X}) = \{\mathcal{X} \cup \{X_i\} \colon i > \max\{j \colon X_j \in \mathcal{X}\}, i \leq d\}$. The second ingredient is the bounding function. A function $\bar{f}$ is called an **admissible bounding function** for an optimization function f, if it holds that $\bar{f}(\mathcal{X}) \geq f(\mathcal{X}')$ for all $\mathcal{X}'$ with $\mathcal{X} \subseteq \mathcal{X}' \subseteq \mathcal{I}$. The bounding function proposed for the reliable mutual information is $\bar{f}_{\mathbf{spc}}(\mathcal{X}, Y) = \hat{I}_0(\mathcal{X} \cup \{Y\}; Y) = \hat{H}_{\text{PL}}(Y) - \sum_{\sigma \in S_n} \hat{I}_{\text{PL}}(\mathcal{X} \cup \{Y\}; Y_\sigma)/(n!)$. With these, the branch-and-bound algorithm enumerates starting from $\emptyset$, tracks the best solution, and prunes expanding elements with $\bar{f}_{\mathbf{spc}}$ that cannot yield an improvement over the best solution. In addition, the framework provides the option of relaxing the required result guarantee to that of an α-approximation for accuracy parameter $\alpha \in (0, 1]$. An $\alpha \leq 1$ trades accuracy for efficiency in a principled manner. The greedy algorithm uses level-wise search where only the best candidate is refined, coupled with $\bar{f}_{\mathbf{spc}}$ for pruning. The specific details of the algorithms are found in [16].

For mixed data, first note the problem is still NP-Hard. Second, $\bar{f}_{\mathbf{spc}}(\mathcal{X}, Y)$ remains admissible as it is independent of "future" partitions for $\mathcal{X}'$ with $\mathcal{X} \subseteq \mathcal{X}'$. However, unlike the discrete case, here evaluating $\hat{I}_0(\mathcal{X}; Y)$ for a candidate $\mathcal{X} = \{\mathcal{D}, C\}$ is more expensive—Alg. 1 sorts in decreasing order of marginal $\hat{I}_0$, and performs $|C|$ discretizations. For efficiency, we first remove the repetitive sorting by sorting $\mathcal{I}$ initially instead. Then the alphabetic refinement operator only refines with variables of smaller marginal mutual information. Second, we apply the following heuristic: once a variable $C \in \mathcal{C}$ has been discretized, it remains discretized for the remaining of the search branch. We refer to the resulting branch-and-bound and greedy algorithms with **BnB** and **Greedy**, respectively.

6 EVALUATION

In this section, we perform an evaluation on the different aspects of our FDD solution for mixed data. In particular, we investigate the statistical performance of various estimators coupled with partitioning techniques on synthetic data, we evaluate the proposed discovery algorithms on real-world benchmark data, and finally, we qualitatively analyze the partitions selected from estimation.

6.1 Estimator performance

First, we focus on the statistical performance of mixed estimator configurations. We are interested in their consistency with regards to the FDD process. For this, we generate data from models governing functional relationships for which we know the population values for mutual information, perform FDD with exhaustive search to obtain the estimated value of the maximizer variable set, and then plot curves corresponding to absolute estimation error.

In this experiment, we model our functional relationships with the class of **generalized linear models**. We consider a set of four continuous random variables $\mathcal{I} = \{X_1, X_2, X_3, X_4\}$, and one categorical variable Y, and distinguish two cases of functional relationship: $\mathbb{E}(Y \mid \mathcal{I}) = f^{-1}(\alpha_0 + \sum_{j=1}^{4} \alpha_j X_j)$ and $\mathbb{E}(Y \mid \mathcal{I}) = f^{-1}(\beta_0 + \sum_{i=1}^{3} \beta_i \sum_{j=1} \alpha_{j,i} g_i(X_j))$, where f is an appropriate link function and $g_1(X) = \log(X + 2), g_2(X) = X^2, g_3(X) = \cos(2X)$ are non-linear variable transformations. We use $h \in \{\text{lin}, \text{nlin}\}$ to indicate the former and latter cases respectively. The coefficients α, β, follow a bimodal Gaussian distribution that uniformly selects one of $\mathcal{N}(-\log(10), 1)$ and $\mathcal{N}(\log(10), 1)$. The means log(10) and -log(10) are chosen such that the respective classes for binary Y (positive

for $\log(10)$ and negative for $-\log(10)$), are 10 times more likely. To cover a wider range of scenarios, we further parametrize these models in two ways: we consider a varying number $e \in \{1, 2, 3\}$ of explanatory variables, with the remaining $4 - e$ receiving weights $\alpha = 0$, and, we use three different domain sizes $d \in \{2, 5, 10\}$ for Y.

For our **generative models** $p_{\alpha,\beta}(\mathcal{I}, Y)$, variables X_j follow a uniform $\mathbf{U}(-1, 1)$ and Y a multinomial with expectations as above. We omit α, β from notation for readability. Given parameters $d \in \{2, 5, 10\}$, $e \in \{1, 2, 3\}$, and $h \in \{\text{lin}, \text{nlin}\}$, we denote the resulting models with $p^l_{e,d}(\mathcal{I}, Y)$. For the conditional $p^h_{e,2}(Y \mid \mathcal{I})$ we use the the sigmoid function (i.e., logit link function), and the softmax for $p^h_{e,\{5,10\}}(Y \mid \mathcal{I})$ (i.e., multinomial logit). The analytic expressions are found in Table 2 in the Appendix. With these, for any set of coefficients α, β, we can compute the population value $I(\mathcal{I}; Y)$. To **sample data** from the models, we first randomly and uniformly sample 90 conditional probability distributions $p^{(i)}, i = 1, \ldots, 90$, 5 for each combination of e, d, h. To make the results comparable, we ensure for each $p^{(i)}$ the population value $F(p^{(i)})$ lies in $(0, 0.5]$. We denote with $\mathcal{P}^l_{e,d}$ the sets of $p^{(i)}$ corresponding to specific e, d, l. For example, $\mathcal{P}^{\text{lin}}_{2,2}$ is the set of the 5 $p^{(i)}$ corresponding to $d = 2, e = 2, h = \text{lin}$. We consider data sizes $n = \{20, 40, 80, 160, 320, 640, 1280, 2560\}$, and for each $p^{(i)}$ and n, we sample 50 datasets $\mathbf{D}^{(i)}_{n,j}, j \in [1, 50]$. Two sampled datasets are illustrated in Fig. 6 in the Appendix.

Now, given these data, we perform the FDD task with input variables $\mathcal{I}$ and target Y, considering different estimator/partitioning configurations combined with exhaustive search. In addition to the estimators discussed so far, i.e., plug-in $\hat{I}_{\text{PL}}$ (Sec. 3), reliable $\hat{I}_0$ (Sec. 4), and $\hat{I}_{\text{MIC}}$ (Example 4.2), we consider two **additional estimators**: the Vinh et al. estimator [28], defined as $\hat{I}_{\chi,\alpha}(\mathcal{X}; Y) = \hat{I}_{\text{PL}}(\mathcal{X}; Y) - \chi_{\alpha, l(\mathcal{X},Y)}/(2n)$, where $\chi_{\alpha, l(\mathcal{X},Y)}$ is the critical value of the χ^2 distribution corresponding to a significance level $1 - \alpha$ and degrees of freedom $l(\mathcal{X}, Y) = (\prod_{X \in \mathcal{X}} V(X) - 1)(V(Y) - 1)$, and the Suzuki estimator based on the MDL principle [26], defined as $\hat{I}_{\text{MDL}}(\mathcal{X}; Y) = \hat{I}_{\text{PL}}(\mathcal{X}; Y) - l(\mathcal{X}, Y) \log(n)/(2n)$.

To evaluate the performance, we use the **absolute estimation error** tailored for FDD, defined as $r_n(\hat{F}_{\text{MX}}, p^{(i)}) = \mathbb{E}(|F(p^{(i)}) - \hat{F}_{\text{MX}}(\mathcal{X}^*_{i,j,n}; Y)|)$, where $F(p^{(i)})$ is the population fraction of information value for model $p^{(i)}$, and $\mathcal{X}^*_{i,j,n} \subseteq \mathcal{I}$ is the maximizer on $\mathbf{D}^{(i)}_{n,j}$ for a configuration $\hat{F}_{\text{MX}}$. We use the fraction of information instead in order to have the error in $[0, 1]$. The expected value is with respect to $j \in [1, 50]$. We average the absolute errors across different $p^{(i)}$ to obtain averages of the form $r_n(\hat{F}_{\text{MX}}, \mathcal{P}^{\{\text{lin},\text{hlin}\}}_{[a,b],\{2,5,10\}})$. For example, $r_n(\hat{F}_{\text{MX}}, \mathcal{P}^{\{\text{lin},\text{hlin}\}}_{[1,3],\{2,5,10\}})$ corresponds to the average absolute error across all 90 models $p^{(i)}$, while the $r_n(\hat{F}_{\text{MX}}, \mathcal{P}^{\text{hlin}}_{[1,3],2})$ would be the average for $p^{(i)} \in \mathcal{P}^{\text{hlin}}_{1,2} \cup \mathcal{P}^{\text{hlin}}_{2,2} \cup \mathcal{P}^{\text{hlin}}_{3,2}$.

We start with Fig. 3 and plot the average error curves across all $p^{(i)}$, for $\hat{F}_0$ with cOP and EF, $\hat{F}_{\text{PL}}$ with cOP, $\hat{F}_{\text{MIC}}$, and $\hat{F}_{\chi,\alpha}$, F_{MDL} with EF. For $\hat{F}_{\chi,\alpha}$, we tested both $\alpha = 0.95$ and 0.99, and show the latter that has better performance. For cOP and EF we use maximum number of bins $l = 5$, and for cOP $c = 2$. In addition, we consider $\hat{F}_{\text{PL}}$ with pre-discretized data in 5 equal-frequency bins as a baseline, which we refer to as PEF. Let us focus first on the three uncorrected configurations, i.e., $\hat{I}_{\text{PL}}$ with cOP, PEF, and $\hat{F}_{\text{MIC}}$.

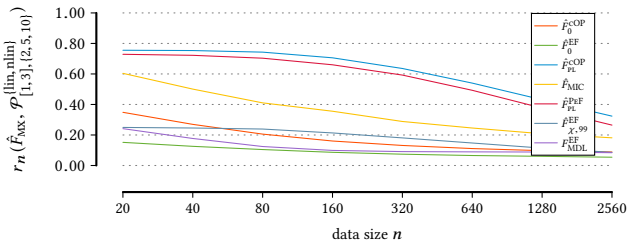


Figure 3: Absolute estimation error averaged across all $p^{(i)}$

All under-perform, with the highest errors and slower convergence rates. Interestingly, we see that PEF performs better than cOP, despite both having $l = 5$. This behavior is attributed to the $\mathcal{I}$ being uniform and independent, and while PEF is well-suited for this, $\hat{F}_{\text{PL}}$ with cOP overfits by finding joint effects. For $\hat{F}_{\text{MIC}}$, the convergence is better, but only because the maximum number $l = g(n, 0.6, \mathcal{G}, Y)$ decreases per $X \in \mathcal{X}$, and the subsequent "coarser" X exchange overfitting for underfitting (as in Example. 4.2). Moving on to the three corrected estimators $\hat{F}_0, \hat{F}_{\chi,99}, \hat{F}_{\text{MIC}}$ combined with EF, we observe lower errors and faster convergence, with $\hat{F}_0$ showing the best performance, and $\hat{F}_{\text{MIC}}$ being "faster" than $\hat{F}_{\chi,99}$. Lastly, the reliable $\hat{F}_0$ combined with cOP has higher error for smaller number of samples, but performs well in terms of convergence speed and "catches" up. Note that $X \in \mathcal{I}$ are uniform, and EF meets this requirement. The cOP with $c = 5, c = 2$, considers only one equal-frequency partition, that of Q^{EF}_5, which cannot be supported for small n due to the correction.

Now let us briefly focus on averages over different configurations. Note that all $p^{(i)}, i \in [90]$, are different models and for the following figures, one should focus mainly on the convergence speed comparison between two plots. In Fig. 4 we show the results averaged over the 45 $p^{(i)}$ corresponding to the additional non-linear layer in the functional relationship, i.e., $h = \text{nlin}$ (right), and over the 45 $p^{(i)}$ with $h = \text{lin}$ (left). Between the two, we observe that convergence speeds are better for the case $h = \text{lin}$, as expected. We also see that EF performs well in both cases. Surprisingly, F_{MDL} does not monotonically converge for $h = \text{hlin}$. Moving on, we average over the 30 $p^{(i)}$ where there is only one explanatory variable, i.e., $e = 1$, and the 30 $p^{(i)}$ with $e = 3$. Additionally, we average over the 30 $p^{(i)}$ with target domain size $V(Y) = 2$, i.e., $d = 2$, and the 30 $p^{(i)}$ with $d = 10$. Due to limited space, Fig. 7 and Fig. 8 corresponding to the former and latter, are found in the Appendix. We report that methods are robust against number of explanatory variables, but there is over fitting for $V(Y) = 2$. Here, the dependency is "easier" to infer and estimators select supersets $\mathcal{X}'$ of $\mathcal{X}^*$, and although $F(\mathcal{X}^*; Y) = F(\mathcal{X}'; Y)$, on the sample there is overestimation.

Overall, we see that correction for FDD is necessary, improving the performance over the plug-in. The reliable $\hat{F}_0$ shows the best performance with EF that accurately fits the uniform data $\mathcal{I}$. Combined with cOP that for $l = 5, c = 2$, considers mostly non-uniform partitions, the error is higher for small n, but the convergence is fast. The EF should be preferred when assumptions are met, e.g, uniform independent input, and cOP for exploratory scenarios.

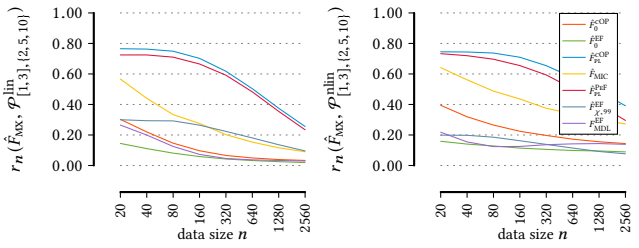


Figure 4: Absolute estimation error averaged across all $p^{(i)}$ with (right) and without (left) the non-linear layer

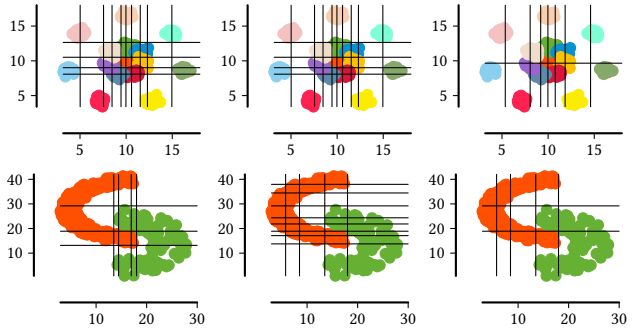
6.2 Discovery performance

Here, we perform FDD on benchmark data from the KEEL data repository [1]. In particular, we use all classification datasets with mixed and continuous input attributes $\mathcal{I}$ and no missing values, resulting in 29 datasets with 25000 and 30 rows and columns on average, respectively. There are 12, 6, and 2, continuous, ordinal discrete, and nominal attributes, respectively, on average per dataset, all summarized in Tab. 1 in the Appendix. We employ the two algorithms BnB and Greedy (Sec. 5) to retrieve the top solution $\mathcal{X}^* \subseteq \mathcal{I}$, both combined with EF and cOP for $l = 5, c = 2$. To increase the difficulty, the ordinal discrete $D \in \mathcal{I}$ per dataset are also partitioned when $V(D) \geq l$. For BnB, and for each of EF and cOP, we set α to be the highest possible in increments of 0.05, such that they terminate in less than 1 hour. For BnB, we report in Tab. 1 the α values, the runtime, the size $|\mathcal{X}^*|$ of the solution, and the value $\hat{F}_0(\mathcal{X}^*; Y)$. Similarly for Greedy, we report runtime and $\hat{F}_0(\mathcal{X}^*; Y)$. The runtimes are averaged over 3 independent executions. This experiment is executed on an Intel Xeon E5-2643 v3 with 256 GB memory. Our Java code is online for research purposes.[1]

We start with BnB and α values. For both EF and cOP, the average α value for BnB to complete in $\leq$ 1 hour is 0.81. There are 14 datasets for EF and 15 for cOP with $\alpha = 1$, which corresponds to an exact solution, while there are 6 datasets for both with $\alpha \in [0.8, 1)$. Here, we see that both methods offer good guarantees with a budget of 1 hour. Regarding the cardinality $|\mathcal{X}^*|$ of the solutions, for EF they have size 3.8 on average, while for cOP 3.5. Again, the two partitioning techniques show similar performance, with cOP returning slightly smaller sets. We hypothesize this is due to the ability of cOP to better adapt on data, and hence extract more information with fewer attributes. Time-wise, EF and cOP require 599 and 743 seconds on average. The cOP is slower, as expected. Finally, the average quality of the solution is 0.52 and 0.53 for EF and cOP, respectively, with cOP recovering 1% more information by considering more candidate partitions. The greedy algorithm is efficient, with EF requiring 51 seconds on average and cOP 43. Interestingly, the quality of the solutions are higher than BnB, with 0.53 and 0.55 on average for EF and cOP, respectively. In fact, the solutions of Greedy have roughly the same quality as those of BnB for high α values, while for smaller α Greedy has better quality.

Overall, we observe that both algorithms BnB and Greedy, with both partitioning techniques EF and cOP, are very effective in practice. For truly exploratory scenarios, cOP should be preferable over EF, unless the assumptions on the data distributions are met by EF. The branch-and-bound algorithm should be used whenever solution guarantees are required. The greedy algorithm, however, is very efficient and hence a better candidate for larger datasets. In addition, it shows good performance in terms of solution quality.

[1] https://github.com/pmandros/fodiscovery



(a) $\hat{I}_0^{\mathbf{cOP}}$, $l = 10$, $c = 3$ **(b)** $\hat{I}_0^{\mathbf{EF}}$, $l = 10$ **(c)** $\hat{I}_{\mathbf{MDL}}^{\mathbf{EF}}$, $l = 10$

Figure 5: Partitions on two clustering datasets in $\mathbb{R}^2$. The target Y is the cluster assignment (colored). (Sec. 6.3)

6.3 Qualitative analysis

Here, we present the resulting partitions from estimating mutual information on clustering datasets in $\mathbb{R}^2$, where the target Y is the cluster assignment [5]. We denote the variables corresponding to x and y-axis with C_1 and C_2, respectively. We use the reliable $\hat{I}_0$ with optimal (cOP) and equal frequency (EF) partitioning, and $\hat{I}_{\text{MDL}}$ with EF. For both cOP and EF we set the maximum number of bins l to 10, and use $c = 3$ in order to have 30 initial equal-frequency bins for cOP. This allows to investigate whether $\hat{I}_0$ overfits by having access to more candidate partitions. For all methods, C_1 is discretized first for better comparison. We present the results in Fig. 5.

The first dataset has 15 clusters. The $\hat{I}_0$ with both cOP and EF results in the same partition in 40 bins, while $\hat{I}_{\text{MDL}}$ in 16. Here, $\hat{I}_0$ performs well at separating the clusters, but $\hat{I}_{\text{MDL}}$ underfits with bins corresponding in the upper-middle area having points from 3 and 4 different clusters. The second dataset has 2 clusters. The $\hat{I}_0$, cOP, configuration with 20 bins in $\mathbb{R}^2$ perfectly separates the clusters, while with EF there are 45 bins. The $\hat{I}_{\text{MDL}}$ has good performance with 15 cells and one non-pure region. In addition to these, we used $\hat{I}_{\chi,99}$ with EF, $\hat{I}_{\text{PL}}$ with cOP, and $\hat{I}_{\text{MIC}}$. Due to limited space we do not show the results. We report that $\hat{I}_{\chi,99}$ has identical results with $\hat{I}_0$ and EF, $\hat{I}_{\text{PL}}$ partitions with the maximum number of bins, while $\hat{I}_{\text{MIC}}$ selects an overly refined partition for C_1, and mostly 2 bins for C_2. Overall, we see that $\hat{I}_0$ results in good partitions for both EF and cOP. For the latter in particular, there is better class separation with less bins. This indicates that $\hat{I}_0$ with cOP selects good partitions, without overfitting on larger spaces of candidates, and can better adapt on more "exotic" distributions. For EF, both $\hat{I}_0$ and $\hat{I}_{\chi,99}$ select finer-grained partitions, while $\hat{I}_{\text{MDL}}$ is conservative. The $\hat{I}_{\text{PL}}$ with cOP and $\hat{I}_{\text{MIC}}$ under-perform, as expected.

7 RELATED WORK

Variants to our FDD problem include functional dependencies for data management [19] and Markov blanket discovery [27]. However, unlike the former here we aim to identify dependencies that hold on the level of the data generating distribution $p(\mathcal{I}, \mathcal{Y})$, and not with regards to the particular dataset $\mathbf{D}$ [8]. Moreover, we are not limited in distributions that can be faithfully represented by a DAG, making our variant more general than standard Markov blanket approaches and well-suited for EDA. However, our results on mixed data estimation are of interest to both communities.

Regarding mutual information estimation, proposed estimators consider mainly the purely discrete and continuous cases. The different families include the discrete [11, 15, 28], while for continuous there is adaptive partitioning [4, 24], k-NN [2, 13], and kernel density estimation [6, 18]. For mixed data, the state-of-the-art k-NN [7] based on the Radon-Nikodym derivative is applicable for multivariate mixtures. None of the above, however, fits to our mixed data FDD scenario. The continuous estimators are defined for Euclidean spaces, where nominal attributes cannot be trivially embedded. Moreover, given purely discrete data, the Radon-Nikodym method recovers the plug-in estimator which trivially fails the FDD task.

Regarding exact algorithms for FDD, Vinh et al. [28] propose an algorithm for $\hat{I}_{\chi,\alpha}$ that bounds the maximum level of the search space, but unlike branch pruning, all candidates up to that level are evaluated and is hence infeasible for large $\mathcal{I}$. Pennerath [20] proposes an efficient algorithm for top-k search with large k.

8 DISCUSSION

We focus on the maximum number of partitions l. The various sublinear to n criteria discussed in this paper, e.g. $\log_2(n)$, correspond to methods that consider univariate pairs. On the one hand, naively extending these for each $C \in \mathcal{C}$ can lead to an exponential increase of partitions in the $|C|$-dimensional space with each data point falling in one hypercube, violating therefore consistency even for optimal estimators (Thm. 3.3). For example, let us assume $n = 10000$. We have $l \approx 13$, and we can already for $|C| = 4$ arrive one point per hypercube. On the other hand, a more appropriate way would be to set $\log_2(n)$ as the maximum number of hypercubes allowed in $\mathbb{R}^{|C|}$, but this can be very conservative–in our example, it would mean to place 10000 data points in 13 hypercubes, regardless of $|C|$. Note that these calculations are done independently of $\mathcal{D}$ that only exacerbates this behavior. For our purposes, we instead considered a fixed l, e.g., 5. This way, and combined with a corrected estimator, we better control for the aforementioned problems. While one could potentially derive a joint criterion accounting for both $|\mathcal{X}|$ and n, we did not consider this investigation here.

9 CONCLUSION

We considered the task of reliable functional dependency discovery from mixed data. We proposed a mixed mutual information estimator framework based on the theoretical process of random variable quantization. We demonstrated how it can be applied for the task of FDD, and instantiated it with the reliable fraction of information. Lastly, we gave algorithms for exact, approximate, and heuristic search. For future work, it would be interesting to consider generalized linear models with correlated explanatory variables, e.g., Gaussian with non-diagonal covariance matrix. That would highlight the importance of the joint discretization our framework considers. Moreover, adaptive partitioning could be applicable, which would allow to consider a different class of candidate partitions.

REFERENCES

[1] J. Alcalà-Fdez, A. Fernàndez, J. Luengo, J. Derrac, and S. Garcìa. 2011. KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework. *J Mult-Valued Log S* 17 (2011), 255–287.

[2] T. Berrett, R. Samworth, and M. Yuan. 2016. Efficient multivariate entropy estimation via k-nearest neighbour distances. *Ann. Stat.* 47 (2016), 288–318.

[3] T. M. Cover and J. A. Thomas. 2006. *Elements of Information Theory*. Wiley-Interscience New York.

[4] G. A. Darbellay and I. Vajda. 1999. Estimation of the information by an adaptive partitioning of the observation space. *IEEETIT* 45 (1999), 1315–1321.

[5] P. Fränti and S. Sieranoja. 2018. K-means properties on six clustering benchmark datasets. *Appl. Intell.* 48 (2018), 4743–4759.

[6] S. Gao, G. V. Steeg, and A. Galstyan. 2015. Estimating Mutual Information by Local Gaussian Approximation. In *UAI*. 278–287.

[7] W. Gao, S. Kannan, S. Oh, and P. Viswanath. 2017. Estimating Mutual Information for Discrete-Continuous Mixtures. In *NIPS*. 5988––5999.

[8] C. Giannella and E. L. Robertson. 2004. On approximation measures for functional dependencies. *Information Systems* 29 (2004), 483–507.

[9] B. R. Goldsmith, M. Boley, J. Vreeken, M. Scheffler, and L. M. Ghiringhelli. 2017. Uncovering structure-property relationships of materials by subgroup discovery. *New Journal of Physics* 19, Article 013031 (2017), 14 pages.

[10] T. Hey, S. Tansley, K. M. Tolle, and others. 2009. *The fourth paradigm: data-intensive scientific discovery*. Vol. 1. Microsoft research Redmond, WA.

[11] J. Jiao, K. Venkat, Y. Han, and T. Weissman. 2015. Minimax Estimation of Functionals of Discrete Distributions. *IEEETIT* 61, 5 (May 2015), 2835–2885.

[12] J. B. Kinney and G. S. Atwal. 2014. Equitability, mutual information, and the maximal information coefficient. *PNAS* 111 (2014), 3354–3359.

[13] A. Kraskov, H. Stögbauer, and P. Grassberger. 2004. Estimating mutual information. *Phys. Rev. E* 69 (2004), 066138.

[14] H. Lancaster. 1969. *The chi-squared distribution*. Wiley.

[15] P. Mandros, M. Boley, and J. Vreeken. 2017. Discovering reliable approximate functional dependencies. In *KDD*. 355–363.

[16] P. Mandros, M. Boley, and J. Vreeken. 2018. Discovering reliable dependencies from data: Hardness and improved algorithms. In *ICDM*. 317–326.

[17] M. Nielsen. 2011. *Reinventing discovery: the new era of networked science*. Princeton University Press.

[18] L. Paninski and M. Yajima. 2008. Undersmoothed Kernel Entropy Estimators. *IEEETIT* 54 (2008), 4384–4388.

[19] T. Papenbrock, J. Ehrlich, J. Marten, T. Neubert, J.-P. Rudolph, M. Schönberg, J. Zwiener, and F. Naumann. 2015. Functional dependency discovery: An experimental evaluation of seven algorithms. *PVLDB* 8 (2015), 1082–1093.

[20] F. Pennerath. 2018. An Efficient Algorithm for Computing Entropic Measures of Feature Subsets. In *ECML-PKDD*. Springer, 483–499.

[21] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti. 2011. Detecting Novel Associations in Large Data Sets. *Science* 334 (2011), 1518–1524.

[22] S. Romano, N. X. Vinh, J. Bailey, and K. Verspoor. 2016. A Framework to Adjust Dependency Measure Estimates for Chance. In *SDM*. 423–431.

[23] M. S. Roulston. 1999. Estimating the errors on measured entropy and mutual information. *Physica D* 125 (1999), 285–294.

[24] J. Silva and S. Narayanan. 2010. Nonproduct Data-Dependent Partitions for Mutual Information Estimation: Strong Consistency and Applications. *IEEE Trans. Sig. Proc.* 58 (2010), 3497–3511.

[25] M. Sugiyama and K. M. Borgwardt. 2013. Measuring Statistical Dependence via the Mutual Information Dimension. In *IJCAI*. 1692––1698.

[26] J. Suzuki. 2019. Mutual Information Estimation: Independence Detection and Consistency. In *ISIT*. 2514–2518.

[27] I. Tsamardinos, C. Aliferis, A. Statnikov, and E. Statnikov. 2003. Algorithms for Large Scale Markov Blanket Discovery. In *FLAIRS*. 376–380.

[28] N. X. Vinh, J. Chan, and J. Bailey. 2014. Reconsidering mutual information based feature selection: A statistical significance view. In *AAAI*.

[29] N. X. Vinh, J. Epps, and J. Bailey. 2009. Information theoretic measures for clusterings comparison: is a correction for chance necessary?. In *ICML*. 1073–1080.

Table 1: Datasets used in Sec. 6.2. Number of attributes is subdivided in to real, integer, and nominal. With / we separate the results for EF (left) and OP (right). The α column corresponds to the highest possible α value such that BnB terminates ≤ 1 hour. The cardinality of the solution is column $|\mathcal{X}^*|$. Last two columns is the score for $\mathcal{X}^*$ by BnB and Greedy, respectively.

						time(s)		$\hat{F}_0(\mathcal{X}^*;Y)$	
dataset	#attr. (r/i/n)	#rows	#classes	α	$\lvert\mathcal{X}^*\rvert$	BnB	Greedy	BnB	Greedy
australian	14 (3/5/6)	690	2	1.00/1.00	5/5	26/25	1/2	0.57/0.56	0.57/0.55
coil2000	85 (0/85/0)	9822	2	0.05/0.05	1/1	1/1	75/38	0.06/0.06	0.14/0.14
fars	29 (5/0/24)	100968	8	0.65/0.65	2/2	4/2	49/43	0.66/0.66	0.68/0.68
german	20 (0/7/13)	1000	2	0.80/1.00	7/6	3040/3065	5/5	0.22/0.21	0.21/0.21
heart	13 (1/12/0)	270	2	1.00/1.00	4/4	8/9	1/2	0.42/0.42	0.43/0.42
ionosphere	33 (32/1/0)	351	2	1.00/1.00	3/3	549/962	1/5	0.61/0.64	0.59/0.66
kddcup	41 (26/0/15)	494020	23	0.95/0.95	2/2	159/122	706/412	0.96/0.97	0.99/0.99
letter	16 (0/16/0)	20000	26	0.95/0.95	5/4	1220/1914	204/122	0.61/0.61	0.62/0.61
lymph.	18 (0/13/5)	148	4	1.00/1.00	4/5	63/85	1/1	0.49/0.48	0.49/0.48
magic	10 (10/0/0)	19020	2	1.00/1.00	5/4	118/435	7/35	0.43/0.43	0.43/0.43
move_libras	90 (90/0/0)	360	15	0.85/0.90	3/2	2043/3183	66/90	0.36/0.36	0.38/0.36
optdigits	64 (0/64/0)	5620	10	0.35/0.45	2/3	16/132	128/122	0.36/0.46	0.59/0.54
page_blocks	10 (4/6/0)	5472	5	1.00/1.00	4/5	58/70	3/8	0.65/0.73	0.65/0.73
penbased	16 (0/16/0)	10992	10	1.00/1.00	5/4	1228/1784	17/28	0.78/0.76	0.78/0.77
ring	20 (20/0/0)	7400	2	0.45/0.35	5/6	1819/777	27/18	0.30/0.35	0.30/0.48
satimage	36 (0/36/0)	6435	7	0.80/0.80	4/4	988/1861	69/104	0.74/0.75	0.73/0.74
segment	19 (19/0/0)	2310	7	1.00/1.00	3/3	153/197	6/9	0.86/0.86	0.86/0.87
sonar	60 (60/0/0)	208	2	0.70/0.70	4/3	435/1808	3/10	0.45/0.41	0.45/0.40
spambase	57 (57/0/0)	4597	2	0.50/0.30	4/3	383/180	15/38	0.50/0.33	0.66/0.57
spectfheart	44 (0/44/0)	267	2	0.65/0.65	3/3	938/1747	4/14	0.34/0.37	0.34/0.33
texture	40 (40/0/0)	5500	11	0.80/0.80	4/4	409/765	46/50	0.76/0.75	0.73/0.77
thyroid	21 (6/15/0)	7200	3	0.55/0.85	3/4	18/24	4/4	0.55/0.85	0.55/0.85
twonorm	20 (20/0/0)	7400	2	0.60/0.20	6/5	1414/200	26/24	0.46/0.20	0.46/0.41
vehicle	18 (0/18/0)	846	4	1.00/1.00	4/3	1275/872	4/10	0.48/0.50	0.49/0.49
vowel	13 (10/3/0)	990	11	1.00/1.00	3/3	12/19	5/5	0.45/0.49	0.47/0.49
wdbc	30 (30/0/0)	569	2	1.00/1.00	5/2	373/286	2/8	0.81/0.82	0.82/0.82
wine	13 (13/0/0)	178	3	1.00/1.00	2/2	1/1	1/1	0.76/0.79	0.74/0.79
wine_red	11 (11/0/0)	1599	11	1.00/1.00	5/4	158/256	12/16	0.21/0.22	0.22/0.23
wine_white	11 (11/0/0)	4898	11	1.00/1.00	5/4	481/779	11/24	0.20/0.19	0.19/0.19
avg.	30 (12/6/2)	25000	7	0.81/0.81	3.8/3.5	599/743	51/43	0.52/0.53	0.53/0.55

APPENDIX

Optimal substructure proof (Theorem 6.1)

Proof. Let us assume w.l.o.g. that $f(m,l)$ corresponds to partition $\pi^* = \{S_1,\ldots,S_l\}$ of l bins, and $V(X_{\pi^*}) = \{x_1,\ldots,x_l\}$. We use $c_j = \sum_{i=1}^{j} n_{x_i}$ for $j \in [l]$, and n^σ denotes the empirical count after a permutation $\sigma \in S_n$ for Y. We have $f(l,m) =$

$$
\begin{aligned}
&- \frac{1}{n!}\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l}\frac{n^\sigma_{ygx_i}}{m}\log\frac{n^\sigma_{ygx_i}}{n_{gx_i}} \\
&+ \sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l}\frac{n_{ygx_i}}{m}\log\frac{n_{ygx_i}}{n_{gx_i}} \\
&= - \frac{1}{n!}\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n^\sigma_{ygx_i}}{m}\log\frac{n^\sigma_{ygx_i}}{n_{gx_i}} \\
&- \frac{1}{n!}\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n^\sigma_{ygx_l}}{m}\log\frac{n^\sigma_{ygx_l}}{n_{gx_l}} \\
&+ \sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n_{ygx_i}}{m}\log\frac{n_{ygx_i}}{n_{gx_i}} \\
&+ \sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n_{ygx_l}}{m}\log\frac{n_{ygx_l}}{n_{gx_l}} \\
&= - \frac{c_{l-1}}{mn!}\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n^\sigma_{ygx_i}}{c_{l-1}}\log\frac{n^\sigma_{ygx_i}}{n_{gx_i}} \\
&- \frac{m-c_{l-1}}{mn!}\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n^\sigma_{ygx_l}}{m-c_{l-1}}\log\frac{n^\sigma_{ygx_l}}{n_{gx_l}} \\
&+ \frac{c_{l-1}}{mn!}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n_{ygx_i}}{c_{l-1}}\log\frac{n_{ygx_i}}{n_{gx_i}} \\
&+ \frac{m-c_{l-1}}{mn!}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n_{ygx_l}}{m-c_{l-1}}\log\frac{n_{ygx_l}}{n_{gx_l}} \\
&= - \frac{c_{l-1}}{mn!}\Big(\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n^\sigma_{ygx_i}}{c_{l-1}}\log\frac{n^\sigma_{ygx_i}}{n_{gx_i}}
\end{aligned}
$$

$$
\begin{aligned}
&- \sum_{y\in Y}\sum_{g\in\mathcal{G}}\sum_{i=1}^{l-1}\frac{n_{ygx_i}}{c_{l-1}}\log\frac{n_{ygx_i}}{n_{gx_i}}\Big)\\
&- \frac{m-c_{l-1}}{mn!}\Big(\sum_{\sigma\in S_n}\sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n^{\sigma}_{ygx_l}}{m-c_{l-1}}\log\frac{n^{\sigma}_{ygx_l}}{n_{gx_l}}\\
&- \sum_{y\in Y}\sum_{g\in\mathcal{G}}\frac{n_{ygx_l}}{m-c_{l-1}}\log\frac{n_{ygx_l}}{n_{gx_l}}\Big)\\
=&\frac{c_{l-1}}{m}\hat{I}_0(G, X_{\pi^*\setminus\{S_l\}}; Y \mid 1, c_{l-1})\\
&+\frac{m-c_{l-1}}{m}\hat{I}_0(G; Y \mid c_{l-1}+1, m)\\
=&\frac{c_{l-1}}{m}f(l-1, c_{l-1}) + \frac{m-c_{l-1}}{m}\hat{I}_0(G; Y \mid c_{l-1}+1, m)\ ,
\end{aligned}
$$

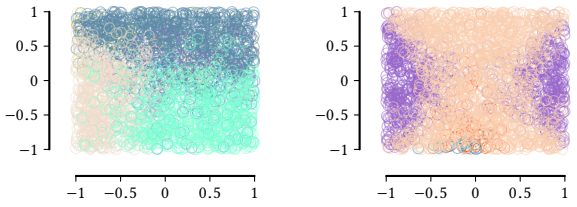
where the last equality holds, otherwise we could increase $f(l,m)$ with a different partition for the first c_{l-1} points. Hence, for $l, m > 1$ we arrive at the following optimal substructure recursive relation

$$
f(l,m) = \max_{1\le i<m}\{\frac{i}{m}f(l-1,i) + \frac{m-i}{m}\hat{I}_0(G; Y \mid i+1, m)\}\ .
$$

□

Table 2: Analytic expressions of $p^h_{e,d}(Y \mid \mathcal{I})$ (Sec. 6.1)

Parameters	Analytic expressions
$h = \text{lin}, d = 2, Y = 1$	$\frac{1}{1+e^{-(\alpha_0+\sum_{j=1}^4\alpha_j X_j)}}$
$h = \text{nlin}, d = 2, Y = 1$	$\frac{1}{1+e^{-(\beta_0+\sum_{i=1}^3\beta_i\sum_{j=1}^4\alpha_{j,i}g_i(X_j)}}$
$h = \text{lin}, d \in \{5,10\}, Y = q$	$\frac{e^{\alpha_{0,q}+\sum_{j=1}^4\alpha_{j,q}X_j}}{\sum_{z=1}^d e^{\alpha_{0,z}+\sum_{j=1}^4\alpha_{j,z}X_j}}$
$h = \text{nlin}, d \in \{5,10\}, Y = q$	$\frac{e^{\beta_{0,q}+\sum_{i=1}^3\beta_{i,q}\sum_{j=1}^4\alpha_{j,i}g_i(X_j)}}{\sum_{z=1}^d e^{\beta_{0,z}+\sum_{i=1}^3\beta_{i,z}\sum_{j=1}^4\alpha_{j,i}g_i(X_j)}}$



(a) linear **(b) non-linear**

Figure 6: Example data sampled for experiment of Sec. 6.1, with 10 classes, 2 explanatory variables, $n = 2560$ data points

Estimator performance additional material

Table 2 shows the analytic expressions of $p^h_{e,d}(Y \mid \mathcal{I})$ for the $p^{(i)}$ used in Section 6.1. By sampling α, β, one can compute $F(\mathcal{I}; Y)$ by integrating. Fig. 6 shows two data sampled for this experiment. In Fig. 7 and Fig. 8 we show the curves averaged for different configurations. In Fig. 8 and for $d = 2$ (left), we do not show $\hat{F}_{\text{MIC}}$ as it could not terminate due to the scale of the experiment.

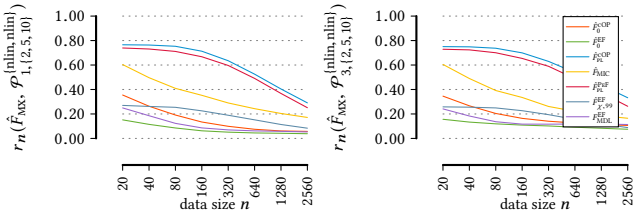


Figure 7: Absolute estimation error averaged across all $p^{(i)}$ with one (left) and three (right) explanatory variables

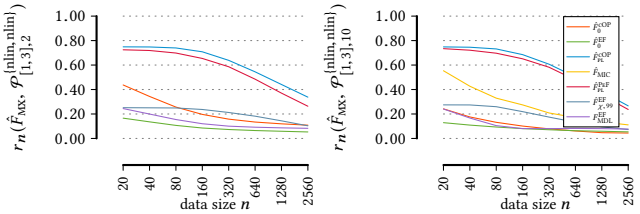


Figure 8: Absolute estimation error averaged across all $p^{(i)}$ with target domain size $V(Y) = 2$ (left), and 10 (right)

Dominated convergence for infinite $V(D)$ (Sec. 3)

Following is a counterexample where $h_k(d)$ is not bounded by such a $a(d)$, but $H(C \mid D = d)$ is uniformly bounded. Let $V(D) = \mathbb{Z}^+, V(C) = [0,1]$ and consider the conditional density $f(c \mid d) = \gamma 1_{c\notin[d^{-1}, d^{-1}+e^{-d})} + (e^d - e^{2d}(c - d^{-1}))/d 1_{c\in[d^{-1}, d^{-1}+e^{-d})}$ where γ makes it a valid pdf. We pick the left end-point for the Riemann approximations. Then, for $k = d$ we have that $h_k(d) = (d-1)/d\gamma\log(\gamma) + 1/d(e^d/d)\log(e^d/d) \approx e^d/d$. However, $H(C \mid D = d) = \frac{1}{1-e^{-d}}\gamma\log(\gamma) + \int_0^{e^{-d}} e^{2d} = \frac{1}{1-e^{-d}}a\log(a) - \frac{1}{4d} - \frac{log(d)}{2d}$. If we pick $p(d) \propto 1/d^2$, there is no upper bound $a(d) \ge \max_k h_k(d) \ge e^d/d$ such that $\sum_d p(d)a(d) < \infty$. In particular, $\sum p(d)h_k(d) \ge p(k)h_k(k) = e^k/k^3$ so that $\lim_k \sum_d p(d)h_k(d)$ does not exist.

Additional proof required for Theorem 4.2

We have $\beta_{Q,Q'}(\mathcal{D}, \mathcal{D}')$

$$
\begin{aligned}
&= \sum_{\mathbf{d}\in\mathcal{D}}\sum_{\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d}, \mathbf{d}')\sum_{i=1}\sum_{j=1}\boldsymbol{\delta}_i\boldsymbol{\delta}'_j f_{C,C'}(\mathbf{c}_i, \mathbf{c}_j \mid \mathbf{d}, \mathbf{d}')\log(\boldsymbol{\delta}_i\boldsymbol{\delta}'_j)\\
&= \sum_{\mathbf{d}\in\mathcal{D}}\sum_{\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d}, \mathbf{d}')\sum_{i=1}\boldsymbol{\delta}_i f_C(\mathbf{c}_i \mid \mathbf{d}, \mathbf{d}')\log(\boldsymbol{\delta}_i)\\
&\quad + \sum_{\mathbf{d}\in\mathcal{D}}\sum_{\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d}, \mathbf{d}')\sum_{j=1}\boldsymbol{\delta}'_j f_{C'}(\mathbf{c}_j \mid \mathbf{d}, \mathbf{d}')\log(\boldsymbol{\delta}'_j)\ .
\end{aligned}
$$

Let us focus on the first term, for which we have

$$
\begin{aligned}
&\sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_{i=1}\boldsymbol{\delta}_i\log(\boldsymbol{\delta}_i)\sum_{\mathbf{d}'\in\mathcal{D}'} p(\mathbf{d}' \mid \mathbf{d}) f_C(\mathbf{c}_i \mid \mathbf{d}, \mathbf{d}')\\
&= \sum_{\mathbf{d}\in\mathcal{D}} p(\mathbf{d})\sum_{i=1}\boldsymbol{\delta}_i\log(\boldsymbol{\delta}_i) f_C(\mathbf{c}_i \mid \mathbf{d})\ ,
\end{aligned}
$$

which is the $\beta_Q(\mathcal{D})$. Similarly, the second term is $\beta_{Q'}(\mathcal{D}')$, and therefore $\beta_{Q,Q'}(\mathcal{D}, \mathcal{D}') = \beta_Q(\mathcal{D}) + \beta_{Q'}(\mathcal{D}')$, concluding the proof.