

The Randomized Information Coefficient: Assessing Dependencies in Noisy Data

Simone Romano · Nguyen Xuan Vinh ·
Karin Verspoor and James Bailey

Received: date / Accepted: date

Abstract When differentiating between strong and weak relationships using information theoretic measures, the variance plays an important role: the higher the variance, the lower the chance to correctly rank the relationships. We propose the Randomized Information Coefficient (RIC), a mutual information based measure with low variance, to quantify the dependency between two sets of numerical variables. We first formally establish the importance of achieving low variance when comparing relationships using the mutual information estimated with grids. Second, we experimentally demonstrate the effectiveness of RIC for (i) detecting noisy dependencies and (ii) ranking dependencies for the applications of genetic network inference and feature selection for regression. Across these tasks, RIC is very competitive over other 16 state-of-the-art measures. Other prominent features of RIC include its simplicity and efficiency, making it a promising new method for dependency assessment.

Keywords dependency measures, noisy relationships, normalized mutual information, randomized ensembles

1 Introduction

There are many possible ways to quantify the dependency between two numerical variables X and Y . If the user is interested in linear dependencies, Pearson's correlation coefficient $r(X, Y)$ can be used. For non-linear cases, information theory provides a well-established measure, the mutual information between X and Y (Cover & Thomas, 2012). More recently, a number of novel information theoretic based measures have been proposed (Reshef et al., 2011; Sugiyama & Borgwardt, 2013). In the last few years, distance based measures have also become popular (Székely et al., 2009; Lopez-Paz et al., 2013) as well as methods that employ kernels to measure dependency (Gretton et al., 2005, 2012). Some of these measures, but not all of them, can also be used to quantify the dependency between two sets

Simone Romano, Nguyen Xuan Vinh, Karin Verspoor, James Bailey
Department of Computing and Information Systems, The University of Melbourne, Victoria
3010, Australia. E-mail: simone.romano@unimelb.edu.au

of numerical variables \mathbf{X} and \mathbf{Y} . Table 1 sketches the difference between dependency measures applicable to variables and sets of variables. All such measures

	Variables	Sets of Variables
Symbol:	$\mathcal{D}(X, Y)$ where X and Y are one dimensional variables	$\mathcal{D}(\mathbf{X}, \mathbf{Y})$ where \mathbf{X} and \mathbf{Y} are resp. a set of p and q variables
Example	$\mathcal{D}(\text{weight}, \text{height})$	$\mathcal{D}(\{\text{weight}, \text{height}\}, \text{BMI})$
Application	Feature filtering for regression, Genetic network inference	Feature selection for regression

Table 1 Differences between dependency measures for variables and set of variables.

are estimated on samples of \mathbf{X} and \mathbf{Y} , and a measure with high variance can encounter problems in discriminating between a strong and a weaker relationship. For example, when testing for independence between \mathbf{X} and \mathbf{Y} , their relationship on a sample is compared to their relationship under the independence assumption (Simon & Tibshirani, 2011). In the case of mutual information, the importance of reducing variance while minimizing the impact on the bias is implied by the statements in Kraskov et al. (2004); Margolin et al. (2006); Schaffernicht et al. (2010), which can be summarized as: *when comparing dependencies, systematic estimation biases cancel each other out*. Therefore smaller variance for mutual information yields a more accurate ranking of relationships. In this paper, we investigate the role of bias and variance of the estimator of mutual information based on grids to compare relationships. Ranking dependencies between variables or set of variables is fundamental for a number of important applications, such as feature selection (Guyon & Elisseeff, 2003) and network inference (Villaverde et al., 2013).

To quantify the dependency between two sets of numerical variables, we propose a low-variance measure based on information and ensemble theory that can capture many relationship types. Our measure, named the Randomized Information Coefficient (RIC), is computed by randomly generating K discretization grids G_k and averaging the normalized mutual information (NI) (Kvalseth, 1987) over all the grids as:

$$\text{RIC}(\mathbf{X}, \mathbf{Y}) \triangleq \frac{1}{K} \sum_{k=1}^K NI((\mathbf{X}, \mathbf{Y})|G_k) \quad (1)$$

Normalization enables us to consider grids with different cardinalities. The normalized mutual information on a grid G is defined as

$$NI((\mathbf{X}, \mathbf{Y})|G) \triangleq \frac{I((\mathbf{X}, \mathbf{Y})|G)}{\max\{H(\mathbf{X}|G), H(\mathbf{Y}|G)\}} \quad (2)$$

where I and H are respectively the mutual information and the entropy function for discrete variables. We choose to normalize by $\max\{H(\mathbf{X}), H(\mathbf{Y})\}$ as it is the tightest upper bound that still preserves the metric properties of NI (Vinh et al., 2010).

The intuition behind this measure is that on average a random grid can encapsulate the relationship between \mathbf{X} and \mathbf{Y} . Both random discretization and ensembles of classifiers have been shown to be effective in machine learning, for example, in random forests (Breiman, 2001). Substantial randomization has been

shown to be even more effective in reducing the variance of predictions (Geurts et al., 2006). Our aim is to exploit this powerful approach to develop an efficient, effective and easy-to-compute statistic for quantifying dependency between two set variables.

Our contributions in this paper are three-fold:

- We propose a low-variance statistic (RIC) based on information and ensemble theory, which is efficient and easy to compute;
- Via theoretical analysis and extensive experimental evaluation, we link our measure’s strong performance on *(i)* discrimination between strong and weak noisy relationships, and *(ii)* ranking of relationships, to its low variance estimation of mutual information;
- We extensively demonstrate the competitive performance of RIC versus 16 state-of-the-art dependency measures using both simulated and real scenarios.

2 Related Work

We first present a brief review of the many available dependency measures and their connections with RIC.

2.1 Correlation and Kernel Based Measures

When the user is only interested in linear dependencies between two variables, the sample Pearson’s correlation coefficient r is powerful. This was extended in Székely et al. (2009) to handle non-linear dependencies between two sets of variables using distance correlation (dCorr). More recently, random projections have been employed to achieve speed improvements (Lopez-Paz et al., 2013), yielding the randomized dependency coefficient (RDC). RDC might be seen as a randomized way to identify the maximal correlation between sets of variables and thus can also be seen as an extension of the alternative conditional expectation (ACE) algorithm proposed in Breiman & Friedman (1985). In our work, the random discretization grids used in RIC can be seen as random projections. However, we do not use a linear measure of dependency such as r because this would require optimization across projections to return a meaningful result. Instead, we compute the normalized mutual information that quantifies non-linear dependencies for each possible projection (grid). This approach allows us to take into account every single grid and each of them contributes to the computation of the average value of NI across grids. No optimization is required.

The correlation between two sets of variables can also be measured employing the joint distribution of the studied variables under kernel embeddings. The Hilbert-Schmidt independence criterion (HSIC) (Gretton et al., 2005) is an example of such measures that has been shown to be competitive in feature selection tasks (Song et al., 2007). RIC measures the dependency between two sets of variables employing their distribution without kernel embeddings: the distribution is efficiently estimated making use of the random grid and no kernels are used because the distribution estimated with the grid can be straightforwardly plugged in the normalized mutual information formula.

2.2 Mutual Information

The mutual information (MI) between two sets of random variables $I(\mathbf{X}, \mathbf{Y})$ is a powerful and well established dependency measure (Cover & Thomas, 2012). A number of different estimators have been proposed for mutual information (Steuer et al., 2002; Kraskov et al., 2004). The standard approach however consists of discretizing the space of possible values that \mathbf{X} and \mathbf{Y} can take and then estimating the probability mass function using the frequency of occurrence. There are many possible approaches to discretization of random variables. For example, a single random variable can be easily discretized according to equal-width or equal-frequency binning, or according to more complex principles such as the minimum description length (Fayyad & Irani, 1993). We note that there is no universally accepted optimal discretization technique. Even though, for sets of variables few sensible discretization have been proposed (Dougherty et al., 1995; Garcia et al., 2013), to our knowledge, there is no extensive survey about the estimation of mutual information with multiple variable discretization approaches.

Mutual information estimators based on discretization in equal width intervals have been discussed in Steuer et al. (2002). Particularly crucial is the choice of the number of bins used to discretize X and Y : too big values lead to overestimation of mutual information due to a finite-sample effect. To mitigate this problem, adaptive partitioning of the discretization grid on the joint distribution (X, Y) has been proposed (Fraser & Swinney, 1986) and optimized for speed (Cellucci et al., 2005). Other competitive mutual information estimators used in practice are Kraskov’s k nearest neighbors estimator (Kraskov et al., 2004) and the kernel density estimator (Moon et al., 1995). An extensive comparison of these estimators can be found in Khan et al. (2007). Mutual information has been successfully employed for a variety of applications, such as feature selection (Nguyen et al., 2014b) and reverse engineering genetic networks (Villaverde et al., 2013). Given the evident number of application scenarios of mutual information and its undeniable efficacy, we choose to use the discretization-based MI estimator as the main building block of RIC. We further make use of normalization because it helps to deflate mutual information on finite samples, bounding the output values in $[0, 1]$ (Romano et al., 2014).

2.3 Other Information Theoretic Measures

More recently, new measures based on information theory, such as the maximal information coefficient (MIC) presented in Reshef et al. (2011) and the mutual information dimension (MID) (Sugiyama & Borgwardt, 2013), have been proposed. MID is based on discretization and it aims to outperform other measures in white noise scenarios. In particular, it outperforms MIC under white noise. Other prominent features of MID include its efficiency with an average running time $\mathcal{O}(n \log n)$, and the ability to characterize multi-functional relationships with a score of 1. MIC is another successful measure of dependence whose value is interpretable in various settings. Its value is obtained by performing discretization using grids over the joint distribution (X, Y) . MIC satisfies a useful property called equitability, which allows it to act as a proxy for the coefficient of determination R^2 of a functional relationship (Reshef et al., 2015b).

Family	Acronym	Name	Reference	Sets of vars.	Best Compl.	Worst Compl.
Mutual information estimators	I_{ew}	Mutual Information (Discretization equal width)	Steuer et al. (2002)	✗	$\mathcal{O}(n^{1.5})$	
	I_{ef}	Mutual Information (Discretization equal frequency)	Steuer et al. (2002)	✗	$\mathcal{O}(n^{1.5})$	
	I_A	Mutual Information (Adaptive Partitioning)	Cellucci et al. (2005)	✗	$\mathcal{O}(n^{1.5})$	
	I_{mean}	Mutual Information (Mean Nearest Neighbours)	Faivishevsky & Goldberger (2009)	✓	$\mathcal{O}(n^2)$	
	I_{KDE}	Mutual Information (Kernel Density Estimation)	Moon et al. (1995)	✓	$\mathcal{O}(n^2)$	
	I_{kNN}	Mutual Information (Nearest Neighbours)	Kraskov et al. (2004)	✓	$\mathcal{O}(n^{1.5})$	$\mathcal{O}(n^2)$
Correlation based	r^2	Squared Pearson's Correlation	-	✗	$\mathcal{O}(n)$	
	ACE	Alternative Conditional Expectation	Breiman & Friedman (1985)	✗	$\mathcal{O}(n)$	
	dCorr	Distance Correlation	Székely et al. (2009)	✓	$\mathcal{O}(n \log n)$	$\mathcal{O}(n^2)$
	RDC	Randomized Dependency Coefficient	Lopez-Paz et al. (2013)	✓	$\mathcal{O}(n \log n)$	
Kernel based	HSIC	Hilbert-Schmidt Independence Criterion	Gretton et al. (2005)	✓	$\mathcal{O}(n^2)$	
Information theory based	MIC	Maximal Information Coefficient	Reshef et al. (2011)	✗	$\mathcal{O}(n)$	$\mathcal{O}(n^{3.6})$
	GMIC	Generalized Mean Information Coefficient	Luedtke & Tran (2013)	✗	$\mathcal{O}(2^n)$	
	MID	Mutual Information Dimension	Sugiyama & Borgwardt (2013)	✗	$\mathcal{O}(n \log n)$	$\mathcal{O}(n^2)$
	MIC_e	Maximal Information Coefficient	Reshef et al. (2015b)	✗	$\mathcal{O}(n)$	$\mathcal{O}(n^{2.25})$
	TIC_e	Total Information Coefficient	Reshef et al. (2015b)	✗	$\mathcal{O}(n)$	$\mathcal{O}(n^{2.25})$
	RIC	Randomized Information Coefficient	-	✓	$\mathcal{O}(n^{1.5})$	

Table 2 Dependency measures available in literature compared by their applicability to sets of variables and their best and worst case computational complexity.

Reshef et al. (2015b) also proposed two new statistics based on grids in this recent preprint: MIC_e which is an improved estimator of the population value of MIC; and the total information coefficient (TIC_e) to achieve high power when testing for independence between variables. In a thorough study, Reshef et al. (2015a) compared many different dependency measures between variables and demonstrated that MIC_e and TIC_e are the state-of-the-art to obtain high equitability and high power respectively. MIC_e optimizes the normalized mutual information over all grid cardinalities and grid cut-offs. TIC_e still optimizes the possible cut-offs for a grid, but returns the sum over grid cardinalities instead. Independently, another statistic based on grids and normalized mutual information has been suggested in the attempt to maximize power: the generalized mean information coefficient (GMIC) (Luedtke & Tran, 2013). Nonetheless, only TIC_e has been shown to be asymptotically consistent and to be the state-of-the-art to achieve power when testing for independence.

In this paper we introduce RIC. RIC is a dependency measure to compare sets of random variables based on normalized mutual information which is efficient and

easy to compute. Table 2 shows a list of dependency measures currently available in literature. Not all of them is applicable to set of variables and some show high computational complexity with regards to the number of points n . Some complexities can be obtained with particular parameter choices or clever implementation techniques. We refer to the respective papers for a detailed analysis. Moreover, recent advances in this area have delivered faster computational techniques for the most recently proposed measures of dependence. For example, the approximated estimator for the population value of MIC can be sped up (Tang et al., 2014; Zhang et al., 2014), and the new exact estimator MIC_e provides very competitive computational complexity. Moreover, very recently a new technique for fast computation of distance correlation has been proposed (Huo & Szekely, 2014).

3 The Randomized Information Coefficient

The Randomized Information Coefficient (RIC) between the set \mathbf{X} of p variables and the set \mathbf{Y} of q variables is defined as the expected Normalized Mutual Information (NMI) across all possible discretization grids that encapsulate the joint probability distribution for (\mathbf{X}, \mathbf{Y}) :

$$\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) \triangleq \int_G \mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G)P(G)dG = \int_G \frac{\mathcal{I}(\mathbf{X}, \mathbf{Y}|G)}{\max\{\mathcal{H}(\mathbf{X}|G), \mathcal{H}(\mathbf{Y}|G)\}}P(G)dG \quad (3)$$

A grid G for the sets of variables \mathbf{X} and \mathbf{Y} is the Cartesian product of the two partitions G_X and G_Y : i.e., $G = G_X \times G_Y$. G_X is a partition of the domain of the variables in \mathbf{X} in r disjoint sets S_u^X . G_Y is a partition of the domain of the variables in \mathbf{Y} in c disjoint sets S_v^Y . There are infinitely many partitions G_X and G_Y , therefore there exists a continuum of discretization grids G . The probability distribution of the grids G is defined via the probability distribution of the partitions G_X and G_Y . Both partitions G_X and G_Y are defined via a number of discretization cut-offs chosen independently. The probability density function (pdf) of a cut-off defined on \mathbf{X} is the pdf of \mathbf{X} , and the pdf of a cut-off defined on \mathbf{Y} is the pdf of \mathbf{Y} . We limit the total number of cut-offs for \mathbf{X} and for \mathbf{Y} to D_{\max} . Being G the Cartesian product of the two partitions, the probability distribution of the grids G is defined by the probability distribution of the list of cut-offs $\gamma_1, \dots, \gamma_{D_{\max}^2}$. Therefore, $P(G) = P(\gamma_1, \dots, \gamma_{D_{\max}^2}) = P(\gamma_1) \cdots P(\gamma_{D_{\max}^2})$.

The grids G_X , G_Y , and G discretize the domain of \mathbf{X} , \mathbf{Y} , and (\mathbf{X}, \mathbf{Y}) respectively so that the true mutual information \mathcal{I} and the true entropy \mathcal{H} can be computed with the following well know formulas:

$$\mathcal{I}(\mathbf{X}, \mathbf{Y}|G) \triangleq \sum_{u=1}^r \sum_{v=1}^c P(\mathbf{X} \in S_u^X, \mathbf{Y} \in S_v^Y) \log \frac{P(\mathbf{X} \in S_u^X, \mathbf{Y} \in S_v^Y)}{P(\mathbf{X} \in S_u^X) \cdot P(\mathbf{Y} \in S_v^Y)} \quad (4)$$

$$\mathcal{H}(\mathbf{X}|G) \triangleq - \sum_{u=1}^r P(\mathbf{X} \in S_u^X) \log P(\mathbf{X} \in S_u^X) \quad (5)$$

$$\mathcal{H}(\mathbf{Y}|G) \triangleq - \sum_{v=1}^c P(\mathbf{Y} \in S_v^Y) \log P(\mathbf{Y} \in S_v^Y) \quad (6)$$

$\mathcal{RIC}(\mathbf{X}, \mathbf{Y})$ is a measure of dependence between the set of variables \mathbf{X} and the set of variables \mathbf{Y} . Being based on mutual information, the true value of RIC is always non-negative and it is equal to 0 under independence of \mathbf{X} and \mathbf{Y} .

Theorem 1 *It holds true that:*

- i) $\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) = 0$ if and only if \mathbf{X} and \mathbf{Y} are independent;
- ii) $\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) \leq 1$.

Proof

i)

(\mathbf{X} and \mathbf{Y} are independent $\Rightarrow \mathcal{RIC}(\mathbf{X}, \mathbf{Y}) = 0$)

If the variables in \mathbf{X} are independent from the variables in \mathbf{Y} , for any randomization grid G it holds true that $\mathcal{I}(\mathbf{X}, \mathbf{Y}|G) = 0$. Therefore, $\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) = 0$.

($\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) = 0 \Rightarrow \mathbf{X}$ and \mathbf{Y} are independent)

For any randomization grid G , the mutual information is always non-negative: $\mathcal{I}(\mathbf{X}, \mathbf{Y}|G) \geq 0$. Thus, the normalized mutual information is also non-negative: $\mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G) \geq 0$. Therefore being RIC the expected value of a non-negative quantity, $\mathcal{RIC}(\mathbf{X}, \mathbf{Y}) = \int_G \mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G)P(G)dG = 0$ implies that $\mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G)$ is equal to 0 for any possible G . If the normalized mutual information is equal to 0 also the mutual information is equal to 0: $\mathcal{I}(\mathbf{X}, \mathbf{Y}|G) = 0$. This implies that \mathbf{X} and \mathbf{Y} are independent according to the discretization imposed by G (Cover & Thomas, 2012). This is true for every possible discretization grid G , therefore also \mathbf{X} and \mathbf{Y} are independent.

ii)

For any grid G , $\mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G) \leq 1$ because $\mathcal{I}(\mathbf{X}, \mathbf{Y}|G) \leq \max\{\mathcal{H}(\mathbf{X}|G), \mathcal{H}(\mathbf{Y}|G)\}$. Thus,

$$\begin{aligned} \mathcal{RIC}(\mathbf{X}, \mathbf{Y}) &= \int_G \mathcal{NMI}(\mathbf{X}, \mathbf{Y}|G)P(G)dG \\ &\leq \int_G P(G)dG = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} P(\gamma_1, \dots, \gamma_{D_{\max}^2})d\gamma_1, \dots, d\gamma_{D_{\max}^2} \\ &= \int_{-\infty}^{\infty} P(\gamma_1)d\gamma_1 \dots \int_{-\infty}^{\infty} P(\gamma_{D_{\max}^2})d\gamma_{D_{\max}^2} = 1. \end{aligned}$$

□

Mind that RIC is equal to 0 when variables in \mathbf{X} are independent from the variables in \mathbf{Y} even if the variables in either the set \mathbf{X} or the set \mathbf{Y} are dependent to each others.

RIC is computed on a data set $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=0, \dots, n-1}$ of n data points according to Equation (1) making use of a finite set of K randomization grids G . When a grid G is applied to a data set, we denote with $(\mathbf{X}, \mathbf{Y})|G$ the contingency table between \mathbf{X} and \mathbf{Y} . A contingency table counts the occurrences of the data points of the studied data set in the portions of the domain defined by S_u^X , S_v^Y , and $S_u^X \cap S_v^Y$ with $1 \leq u \leq r$ and $1 \leq v \leq c$. Let a_u and b_v be the count of data points in the portion of the domain defined by S_u^X and S_v^Y respectively. Let n_{uv} be the number of data points in the portion of the domain defined by $S_u^X \cap S_v^Y$. Table 3 shows an example of contingency table. The mutual information $\mathcal{I}(\mathbf{X}, \mathbf{Y}|G)$, the

		G_Y				
		b_1	\cdots	b_v	\cdots	b_c
G_X	a_1	n_{11}	\cdots	\cdot	\cdots	n_{1c}
	\vdots	\vdots		\vdots		\vdots
	a_u	\cdot		n_{uv}		\cdot
	\vdots	\vdots		\vdots		\vdots
	a_r	n_{r1}	\cdots	\cdot	\cdots	n_{rc}

Table 3 Contingency table $(\mathbf{X}, \mathbf{Y})|G$ on a data set $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=0, \dots, n-1}$ defined by the grid G .

entropy $H(\mathbf{X}|G)$, and the entropy $H(\mathbf{Y}|G)$ are estimated a data set of n points as follows:

$$I(\mathbf{X}, \mathbf{Y}|G) \triangleq \sum_{u=1}^r \sum_{v=1}^c \frac{n_{uv}}{n} \log \frac{n_{uv} \cdot n}{a_u b_v} \quad (7)$$

$$H(\mathbf{X}|G) \triangleq - \sum_{u=1}^r \frac{a_u}{n} \log \frac{a_u}{n} \quad H(\mathbf{Y}|G) \triangleq - \sum_{v=1}^c \frac{b_v}{n} \log \frac{b_v}{n} \quad (8)$$

Here we propose a few practical ways to obtain contingency tables $(\mathbf{X}, \mathbf{Y})|G$ based on the random grid G . First of all, by performing K_r random discretizations for both \mathbf{X} and \mathbf{Y} , we can *efficiently* compute $K = K_r^2$ random grids obtained using all pairs of random discretizations. This allows us to generate fewer random discretizations than by independently generating each grid. The other required parameter is D_{\max} , which determines the maximum number of random bins to discretize one variable. Once both variables are discretized, the NMIPROC procedure can be used to compute the normalized mutual information. Algorithm 1 presents the pseudo-code for RIC computation.

Algorithm 1 RIC computation.

```

RIC( $\mathbf{X}, \mathbf{Y}, K_r, D_{\max}$ )
1 for  $k \leftarrow 1$  to  $K_r$  do
2    $BinLabelX_k \leftarrow \text{RANDOMDISCR}(\mathbf{X}, D_{\max})$ 
3    $BinLabelY_k \leftarrow \text{RANDOMDISCR}(\mathbf{Y}, D_{\max})$ 
4 for  $k \leftarrow 1$  to  $K_r$  do
5   for  $k' \leftarrow 1$  to  $K_r$  do
6      $RIC+ = \text{NMIPROC}(BinLabelX_k, BinLabelY_{k'})$ 
7 return  $RIC/K_r^2$ 

```

Discretization of Random Variables: Next we present in Algorithm 2 the random discretization procedure for a single random variable X . A variable is discretized using a number of cut-offs D chosen at random in $[1, D_{\max} - 1]$. Each cut-off is chosen by sampling a random example of the variable with uniform distribution. The bin label for each data point can easily be encoded with integer values using $\mathbb{I}(\text{cut-off} < x_i)$ with D passes through the data points, where \mathbb{I} is the indicator function. The idea is inspired by random ferns (Kursa, 2012; Özuysal et al., 2007): a type of random forest that achieves even higher speed. This can also

be viewed as a random hash function (Wang et al., 2012) or a random projection on a finite set (Lopez-Paz et al., 2013). This procedure can be easily implemented in any programming language, for example C++. No sorting is required. The worst

Algorithm 2 Random discretization of a random variable X .

```

RANDOMDISCR( $X, D_{\max}$ )
1 Choose the number of random cut-offs  $D$  at random between  $[1, D_{\max} - 1]$ 
2 for  $d \leftarrow 1$  to  $D$  do
3    $cut\text{-}off \leftarrow$  random data point
4   for  $i \leftarrow 0$  to  $n - 1$  do
5      $BinLabelX(x_i) += \mathbb{I}(cut\text{-}off < x_i)$ 
6 return  $BinLabelX$ 

```

case computational complexity of this procedure is $\mathcal{O}(D_{\max} \cdot n)$.

Discretization of Sets of Random Variables: An efficient approach to randomly discretize a set of p random variables \mathbf{X} consists not only in choosing cut-offs at random but also to randomly choose the variables to discretize: i.e. build a random fern (Kursa, 2012) on the set of features \mathbf{X} . This is very computationally efficient: the worst case computational complexity is $\mathcal{O}(D_{\max} \cdot n)$ which is independent from the number of variables p . However, the straightforward implementation of a random fern presented in Algorithm 3 does not allow to have fine control on the number of generated bins D_{\max} : the number of maximum bins D_{\max} is exponential in the number of cut-offs D , i.e. $D_{\max} = 2^D$. Therefore D cannot be greater than $\log_2 D_{\max} - 1$. Moreover, many bins can be empty due to repeated choices of the same variable.

Algorithm 3 Random *fern* discretization of a set \mathbf{X} of random variables

```

RANDOMDISCRFERN( $\mathbf{X}, D_{\max}$ )
1 Choose the number of random cut-offs  $D$  at random between  $[1, \log_2 D_{\max} - 1]$ 
2 for  $d \leftarrow 0$  to  $D - 1$  do
3    $j \leftarrow$  random index of variable in  $\mathbf{X}$ 
4    $cut\text{-}off \leftarrow$  random data point for  $X_j$ 
5   for  $i \leftarrow 0$  to  $n - 1$  do
6      $BinLabelX(\mathbf{x}_i) += 2^d \cdot \mathbb{I}(cut\text{-}off < x_{ij})$ 
7 return  $BinLabelX$ 

```

We therefore use the following randomized approach to discretize a set of p variables \mathbf{X} in exactly D bins, maintaining linear worst case complexity in the number of variables and records: $\mathcal{O}(D_{\max} \cdot n \cdot p)$. By choosing D random data points as seeds, we can easily discretize a set of variables into D non-empty bins by assigning each data point to its closest seed. We make use of the Euclidean norm to find the distances between points. For the ease of implementation, both random cut-offs in Algorithm 3 and random seeds in Algorithm 4 are chosen via sampling with replacement.

Algorithm 4 Random seeds discretization of a set \mathbf{X} of random variables

RANDOMDISCRSEEDS(\mathbf{X}, D_{\max})

- 1 Choose the number of random seeds D at random between $[2, D_{\max}]$
 - 2 Choose a set $S = \{\mathbf{s}_1 \dots \mathbf{s}_j \dots \mathbf{s}_D\}$ of D random seeds among the data points
 - 3 **for** $i \leftarrow 0$ **to** $n - 1$ **do**
 - 4 $BinLabelX(\mathbf{x}_i) = \arg \min_{j: \mathbf{s}_j \in S} \text{Dist}(\mathbf{x}_i, \mathbf{s}_j)$
 - 5 **return** $BinLabelX$
-

The worst case computational complexity for Algorithm 1 to compute RIC between the set \mathbf{X} of p variables and the set \mathbf{Y} of q variables is thus determined by the discretization algorithm:

- $\mathcal{O}(K_r \cdot D_{\max} \cdot n + K_r^2(n + D_{\max}^2))$ if random ferns are used;
- $\mathcal{O}(K_r \cdot D_{\max} \cdot n \cdot (p + q) + K_r^2(n + D_{\max}^2))$ if random seeds are used.

K_r controls the trade-off between accuracy and computational time. The more randomizations K_r are used, the lower the variance, but the longer the computational time. Based on experimentation we consider $K_r = 20$ a reasonable value. The number of maximum bins D_{\max} should be chosen in order to avoid increasing the grid resolution towards the limit of $NI = 1$ where each point belongs to a single cell. In the worst case, for uniformly distributed variables and n samples we would like to have at least one point per cell of the contingency table in Table 3. This implies:

$$\frac{n}{D_{\max} D_{\max}} \geq 1 \Rightarrow \frac{n}{D_{\max}^2} \geq 1 \Rightarrow D_{\max}^2 \leq n \Rightarrow D_{\max} = \lfloor \sqrt{n} \rfloor$$

However, a larger value of D_{\max} might help to identify more complex relationships, at the cost of higher variance. D_{\max} can be tuned to obtain optimal performance. Given that in our analysis we used $D_{\max} = \mathcal{O}(\sqrt{n})$, RIC's worst case computational complexity in the number of data samples is $\mathcal{O}(n^{1.5})$.

4 Variance Analysis of Grid Estimators of Mutual Information

In this section, we theoretically justify the use of random grids to obtain small variance with the RIC statistic. Then, we prove that a lower variance is beneficial when comparing dependencies and ranking relationships according to the grid estimator of mutual information.

4.1 Ensembles for Reducing the Variance

The main motivation for our use of random discretization grids is that averaging across independent random grids allows reduction of variance (Geurts, 2002). By using random grids, it is possible to achieve small correlation between the different estimations of NI . RIC variance tends to be a small value if the estimations are uncorrelated.

Theorem 2 Let $NI_G = NI((\mathbf{X}, \mathbf{Y})|G)$ be the normalized mutual information estimated on a random grid G and RIC as per Eq. (1). If NI estimations for RIC are uncorrelated then:

$$\lim_{K \rightarrow \infty} \text{Var}(\text{RIC}) = \text{Var}_G(E[NI_G|G])$$

Proof The variance of RIC can be decomposed using Eve's law of total variance according to the i.i.d. random variables grids G_k with $k = 1 \dots K$ as follows, $\text{Var}(\text{RIC})$ is equal to:

$$\begin{aligned} \text{Var}(\text{RIC}) &= \text{Var}_{G_1 \dots G_K} \left(E[\text{RIC}|G_1 \dots G_K] \right) + E_{G_1 \dots G_K} \left[\text{Var}(\text{RIC}|G_1 \dots G_K) \right] \\ &= \text{Var}_{G_1 \dots G_K} \left(E \left[\frac{1}{K} \sum_{k=1}^K NI_{G_k} | G_1 \dots G_K \right] \right) \\ &\quad + E_{G_1 \dots G_K} \left[\text{Var} \left(\frac{1}{K} \sum_{k=1}^K NI_{G_k} | G_1 \dots G_K \right) \right] \\ &= \text{Var}_{G_1 \dots G_K} \left(\frac{1}{K} \sum_{k=1}^K E[NI_{G_k} | G_k] \right) + E_{G_1 \dots G_K} \left[\frac{1}{K^2} \left(\sum_{k=1}^K \text{Var}(NI_{G_k} | G_k) \right. \right. \\ &\quad \left. \left. + \sum_{k \neq k'} \text{Cov}(NI_{G_k}, NI_{G_{k'}} | G_k, G_{k'}) \right) \right] \\ &= \text{Var}_G(E[NI_G|G]) + E_{G_1 \dots G_K} \left[\frac{1}{K^2} \left(\sum_{k=1}^K \text{Var}(NI_{G_k} | G_k) \right. \right. \\ &\quad \left. \left. + \sum_{k \neq k'} \text{Corr}(NI_{G_k}, NI_{G_{k'}} | G_k, G_{k'}) \text{Var}(NI_{G_k} | G_k) \text{Var}(NI_{G_{k'}} | G_{k'}) \right) \right] \end{aligned}$$

If $\text{Corr}(NI_{G_k}, NI_{G_{k'}} | G_k, G_{k'}) = 0$ for all k and k' , then:

$$\begin{aligned} \text{Var}(\text{RIC}) &= \text{Var}_G(E[NI_G|G]) + E_{G_1 \dots G_K} \left[\frac{1}{K^2} \sum_{k=1}^K \text{Var}(NI_{G_k} | G_k) \right] \\ &= \text{Var}_G(E[NI_G|G]) + \frac{E_G[\text{Var}(NI_G|G)]}{K} \end{aligned}$$

that when $K \rightarrow \infty$ is equal to $\text{Var}_G(E[NI_G|G])$. \square

The expected value $E[NI_G|G]$ is less dependent on the data because of the random grid G and shows small variance across grids. Intuitively, this result suggests that some variance of the data can be captured with the random grids. We empirically validate this result in Figure 1. In practice, it is very difficult to obtain completely uncorrelated NI estimations. Nonetheless, the use of random grids allows us to strongly decrease their correlation.

We aim to show that the decrease in variance is due to the random grid G , by comparing the variance of NI_F where F is a fixed grid with equal width bins for X and Y . The number of bins for each variable is fixed to 9 for both G and F , and cut-offs are generated in the range $[-2, 2]$ and $[-3, 3]$ for X and Y , respectively. The chosen joint distribution (X, Y) is induced on $n = 100$ points with $X \sim \mathcal{N}(0, 1)$ and $Y = X + \eta$ with $\eta \sim \mathcal{N}(0, 1)$. The variance of RIC decreases

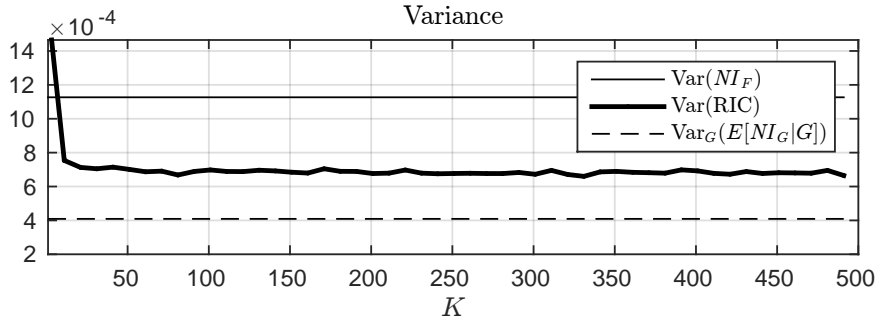


Fig. 1 Variance of RIC compared to the variance of NI_F on a fixed equal width grid F . According to Theorem 2 if estimations are uncorrelated, the variance of RIC tends to the variance of $E[NI_G|G]$ which is less dependent to the data. In practice, estimations are always correlated. Nonetheless, the use of random grids helps in decreasing the correlation between them.

as K increases because the random grids enable us to decorrelate the estimations of NI . In general, if we allow grids of different cardinality (different number of cut-offs) and large K , the variance can be decreased even further.

Using RIC in Algorithm 1 we can efficiently compute $K = K_r^2$ grids. Increasing the number of random grids by increasing K_r is always beneficial. However, this is particularly important when the sample size n is small. In Figure 2 we show the behavior of RIC's variance at the variation of K_r for different sample size n for the same relationship discussed above. The variance reaches the plateau already at $K_r = 50$ when $n = 500$. On the other hand, when the sample size is small, e.g. $n = 50$, the variance is still decreasing at $K_r = 100$. K_r might be chosen according to the sample size n : i.e. larger if the sample size n is small and smaller if the sample size n is large. Nonetheless, having a large K_r is always beneficial in general, at the cost of computational time.

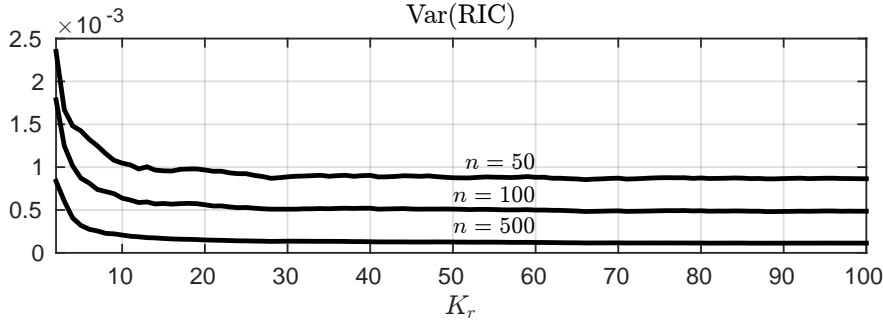


Fig. 2 Variance of RIC in Algorithm 1 at the increase of the number of random grids for different sample size n . Increasing K_r is always beneficial. However, it is particularly important when n is small. For example, the variance of RIC still decreases for $K_r > 50$ for this particular relationship between X and Y .

4.2 Importance of Variance in Comparing Relationships using the Grid Estimator of Mutual Information

When mutual information is used as a proxy for the strength of the relationship, a small estimation variance is likely to be more useful than a smaller bias when comparing relationships, as implied by some observations in Kraskov et al. (2004); Margolin et al. (2006); Schaffernicht et al. (2010). The reason is that systematic biases cancel each other out. We formalize these observations as follows:

Theorem 3 *Let $\text{bias}(\hat{\phi}) = \phi - E[\hat{\phi}]$ be the bias of the estimator $\hat{\phi}$. Let $\hat{\phi}(s) = \hat{\phi}_s$ and $\hat{\phi}(w) = \hat{\phi}_w$ be estimations of ϕ on the strong relationship s and the weak relationship w , where the true values are $\phi_s > \phi_w$. The probability of making an error $P(\hat{\phi}_s \leq \hat{\phi}_w)$ is bounded above by:*

$$\frac{\text{Var}(\hat{\phi}_s) + \text{Var}(\hat{\phi}_w)}{\text{Var}(\hat{\phi}_s) + \text{Var}(\hat{\phi}_w) + \left(\phi_s - \phi_w - (\text{bias}(\hat{\phi}_s) - \text{bias}(\hat{\phi}_w))\right)^2}$$

if $E[\hat{\phi}_s] > E[\hat{\phi}_w]$ or equivalently if $\phi_s - \phi_w > \text{bias}(\hat{\phi}_s) - \text{bias}(\hat{\phi}_w)$.

Proof Let $\hat{\Delta} = \hat{\phi}_w - \hat{\phi}_s$, if $E[\hat{\Delta}] < 0$ then:

$$P(\hat{\Delta} \geq 0) = P(\hat{\Delta} - E[\hat{\Delta}] \geq -E[\hat{\Delta}]) \leq \frac{\text{Var}(\hat{\Delta})}{\text{Var}(\hat{\Delta}) + E[\hat{\Delta}]^2}$$

according to the 1-sided Chebyshev inequality also known as Cantelli's inequality (Ross, 2012). \square

Remark: *If there is a systematic bias component, the variance of a dependency measure is important also to identify if a relationship exists. The probability of making an error in determining if a relationship exists (independence testing between X and Y with $\hat{\phi}$) is just a special case of Theorem 3 where $\phi_w = 0$.*

Regarding the grid estimator of mutual information I_F on a fixed grid F with n_F bins, there is always a systematic bias component which is a function of the number of samples n and the number of bins n_F (Moddemeijer, 1989). This systematic bias component cancels out in $\text{bias}(I_{F,s}) - \text{bias}(I_{F,w})$. If the non-systematic estimation bias is small enough, then the denominator of the upper bound is dominated by the true difference $\mathcal{I}_s - \mathcal{I}_w$. Therefore, the upper bound decreases because of the numerator, i.e., the sum of the variances. Of course variance is just part of the picture. It is worth to decrease the variance of an estimator if the estimand has some utility. Moreover, many estimators have a bias and variance trade-off. Deliberately reducing the variance at the expense of bias is not a good idea. Variance can be reduced if there is a strong systematic estimation bias component and if the effect on the non-systematic bias is minimal.

We empirically compare the probability of error as stated in Theorem 3 for the estimation mutual information \mathcal{I} with grids. RIC can be used to estimate mutual information if we average across grids of the same cardinality and do not normalize mutual information on the grids. Let $s = (X, Y_s)$ and $w = (X, Y_w)$ be the strong and the weak relationships where $X \sim \mathcal{N}(0, 1)$, $Y_s = X + \eta_s$ and $Y_w = X + \eta_w$ with $\eta_s \sim \mathcal{N}(0, .7)$ and $\eta_w \sim \mathcal{N}(0, 1)$. Indeed, if $X \sim \mathcal{N}(0, \sigma_X^2)$ and $Y = X + \eta$ with

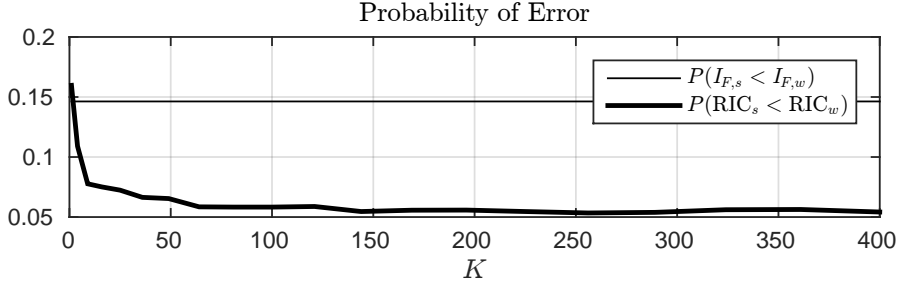


Fig. 3 Probability of error in identifying the strong relationship. RIC's probability is smaller due to its small variance.

$\eta \sim \mathcal{N}(0, \sigma_\eta^2)$ it is possible to analytically compute the mutual information between X and Y : $\mathcal{I}(X, Y) = 0.5 \log_2(1 + \sigma_X^2 / \sigma_\eta^2)$. In Figure 3 we compare the probability of error $P(\text{RIC}_s < \text{RIC}_w)$ for RIC as an estimator of mutual information and the probability of error $P(I_{F,s} < I_{F,w})$ for the estimator I_F on a fixed equal width grid F , with an increase of the number of random grids K for RIC. We generate 13 bins for X and Y for both F and RIC's grids. The distributions are induced on $n = 100$ samples. The probability of error is smaller for RIC because of its small variance. Indeed, the probability of error decreases with the increase of K , just as the variance decreases with bigger K . The bias stays constant when K varies and it contributes less to a small probability of error.

In Figure 4 we show in more detail the contribution of the bias and the variance to the probability of error. The upper bounds for the probability of error for RIC and I_F are respectively:

$$U(\text{RIC}) = \frac{\text{Var}(\text{RIC}_s) + \text{Var}(\text{RIC}_w)}{\text{Var}(\text{RIC}_s) + \text{Var}(\text{RIC}_w) + \left(\mathcal{I}_s - \mathcal{I}_w - (\text{bias}(\text{RIC}_s) - \text{bias}(\text{RIC}_w))\right)^2}$$

$$U(I_F) = \frac{\text{Var}(I_{F,s}) + \text{Var}(I_{F,w})}{\text{Var}(I_{F,s}) + \text{Var}(I_{F,w}) + \left(\mathcal{I}_s - \mathcal{I}_w - (\text{bias}(I_{F,s}) - \text{bias}(I_{F,w}))\right)^2}$$

Figure 4 shows also the behaviour of the upper bound at the variation of K term by term. The bias difference for RIC as an estimator of mutual information \mathcal{I} is a bit bigger than the bias difference for I_F . Nonetheless, the probability of error decreases mainly because of the variance decrease of RIC.

Moreover, when the dependency measure with a systematic bias is used for ranking relationships, we can still show that reducing the estimator variance plays an important role.

Corollary 1 *When ranking m relationships according to the true ranking $\phi_1 > \phi_2 > \dots > \phi_m$, the probability $P(\hat{\phi}_1 > \hat{\phi}_2 > \dots > \hat{\phi}_m)$ of accurately obtaining the correct ranking using the estimators $\hat{\phi}_i, i = 1, \dots, m$ is bounded below by:*

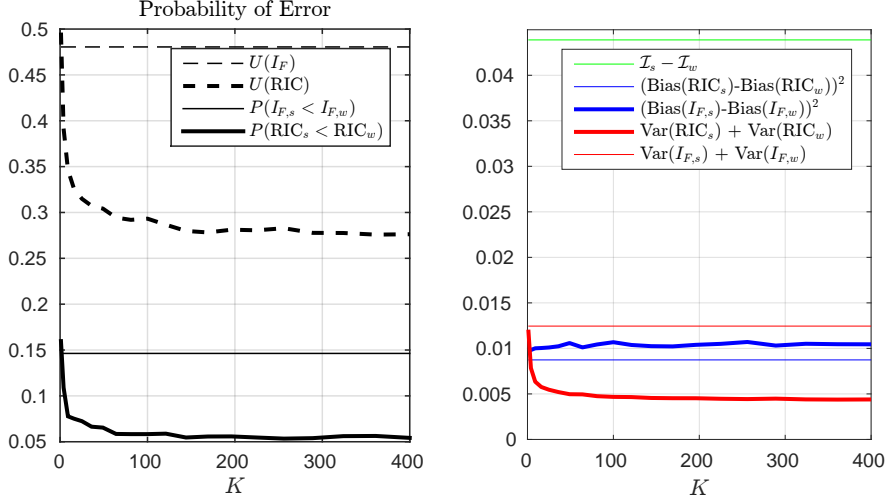
$$1 - \sum_{i=1}^{m-1} \frac{\text{Var}(\hat{\phi}_{i+1}) + \text{Var}(\hat{\phi}_i)}{\text{Var}(\hat{\phi}_{i+1}) + \text{Var}(\hat{\phi}_i) + \left(\phi_{i+1} - \phi_i - (\text{bias}(\hat{\phi}_{i+1}) - \text{bias}(\hat{\phi}_i))\right)^2}$$

if $E[\hat{\phi}_{i+1}] > E[\hat{\phi}_i]$ or equivalently if $\phi_{i+1} - \phi_i > \text{bias}(\hat{\phi}_{i+1}) - \text{bias}(\hat{\phi}_i) \forall i = 1 \dots m-1$.

Fig. 4 Probability of error in identifying the strong relationship. RIC's probability is smaller due to its small variance.

(a) Probability of Error and Upper Bound

(b) Terms of the upper bound



Proof Let $\mathcal{E}_i = \{\hat{\phi}_{i+1} > \hat{\phi}_i\}$ be an event then:

$$\begin{aligned} P(\hat{\phi}_1 > \hat{\phi}_2 > \dots > \hat{\phi}_m) &= P(\mathcal{E}_1 \cap \mathcal{E}_2 \cap \dots \cap \mathcal{E}_{m-1}) = 1 - P(\mathcal{E}_1^c \cup \mathcal{E}_2^c \cup \dots \cup \mathcal{E}_{m-1}^c) \\ &\geq 1 - \sum_{i=1}^{m-1} P(\mathcal{E}_i^c) \end{aligned}$$

where \mathcal{E}_i^c is the complementary event to \mathcal{E}_i : $\mathcal{E}_i^c = \{\hat{\phi}_{i+1} \leq \hat{\phi}_i\}$. The corollary follows using the upper bound for $P(\mathcal{E}_i^c)$ proved in Theorem 3:

$$P(\mathcal{E}_i^c) \leq \frac{\text{Var}(\hat{\phi}_{i+1}) + \text{Var}(\hat{\phi}_i)}{\text{Var}(\hat{\phi}_{i+1}) + \text{Var}(\hat{\phi}_i) + \left(\phi_{i+1} - \phi_i - (\text{bias}(\hat{\phi}_{i+1}) - \text{bias}(\hat{\phi}_i))\right)^2} \quad \square$$

As we empirically demonstrated above for the grid estimator of mutual information, $\text{bias}(\hat{\phi}_{i+1}) - \text{bias}(\hat{\phi}_i)$ tends to be small if there is some systematic bias component, and thus a small variance is the main contributor to the accuracy.

Remark about bootstrapping: It is also natural to consider whether using bootstrapping improves the discrimination performance of a statistic by decreasing the variance. When bootstrapping, the statistic is actually estimated on around 63% of the samples and this decreases the discrimination ability of each measure. Similarly, sampling without replacement of a smaller number of points and averaging across different estimation of a measure is not expected to perform well. The best way to decrease the variance is thus to inject randomness in the estimator itself. This is the rationale for RIC. We achieve this goal by using a strong measure such as mutual information and injects randomness in its estimation in order to decrease the global variance.

5 Experiments on Dependency Between Two Variables

In this section, we compare RIC¹ with 16 other state-of-the-art statistics that quantify the dependency between two variables X and Y . We focus on three tasks: identification of noisy relationships, inference of network of variables, and feature filtering for regression. Table 4 shows the list of competitor measures compared in this paper and the parameters used in their analysis. The parameters used are the default parameters suggested by the authors of the measures in their respective papers. Indeed, only on the task of feature filtering for regression it is possible to tune parameters with cross-validation on a given data set. The tasks of inference of network of variables and identification of noisy relationships are unsupervised learning tasks and do not allow parameter tuning when applied to a new data set. Nonetheless, most of the default parameters are not tuned for hypothesis testing. Therefore, we decided to follow the approach used in Reshef et al. (2015a). In this comprehensive empirical study, leading measures of dependence are compared in terms of two important features: equitability and power against independence. Similarly in this paper, we discuss the power against independence on different noise models as well as the equitability of the measures. When testing the power of a measure for a particular noise model, we identify the best parameters for independence testing by maximizing the power on average on a set of relationships and different noise levels.

The measures in the first group of Table 4 are mutual information estimators. I_{ew} and I_{ef} are respectively the equal-width and equal-frequency bin grid estimator of mutual information. I_A ² is the adaptive grid estimator of mutual information that assures the number of points for each cell to be at least 5. We chose to fix the number of bins D for I_{ew} and I_{ef} to $\lfloor \sqrt{n/5} \rfloor$ because no universally accepted value was found in the literature. Kraskov’s k nearest neighbours estimator I_{kNN} ³ uses a fixed parameter $k = 6$ and the kernel density estimator I_{KDE} ⁴ uses the parameter $h_0 = \frac{4}{p+q+2} n^{-1/(p+q+4)} = n^{-1/6}$ when comparing two variables given that the number of variables is $p + q = 2$. This is one possible kernel width and suggested as a default value in Steuer et al. (2002). We also compare a novel information theoretic estimator of mutual information which has a nice analytical form and can be obtained from the average of I_{kNN} for k from 1 to $n - 1$ (Faivishevsky & Goldberger, 2009). All other measures were used with the default parameters suggested in their respective papers as described in Table 4: dCorr⁵, RDC, ACE⁶, HSIC⁷, MIC⁸, GMIC, MID⁹. As we discussed above, we tuned the parameters of each measure when testing for independence. Being the state-of-the-art in this task, we also introduced TIC_e in the analysis. Regarding RIC on computing dependency between two variables, we decided to generate discretizations for X and

¹ RIC implementation is available at <https://sites.google.com/site/randinfocoeff/>

² From <http://www.iim.csic.es/~gingproc/mider.html> (Villaverde et al., 2014).

³ From <http://jlizier.github.io/jidt/> (Lizier, 2014).

⁴ From <http://tinyurl.com/ojlkrla> (Margolin et al., 2006).

⁵ From <http://tinyurl.com/ozadxzr>.

⁶ From <http://tinyurl.com/oja3k3v>.

⁷ From <http://people.kyb.tuebingen.mpg.de/arthur/indep.htm>.

⁸ From <https://github.com/minepy/minepy>.

⁹ From <https://github.com/mahito-sugiyama/MID>.

Y according Algorithm 2; we generate for each discretization a random number of cut-offs D chosen at random uniformly in $[1, D_{\max} - 1]$.

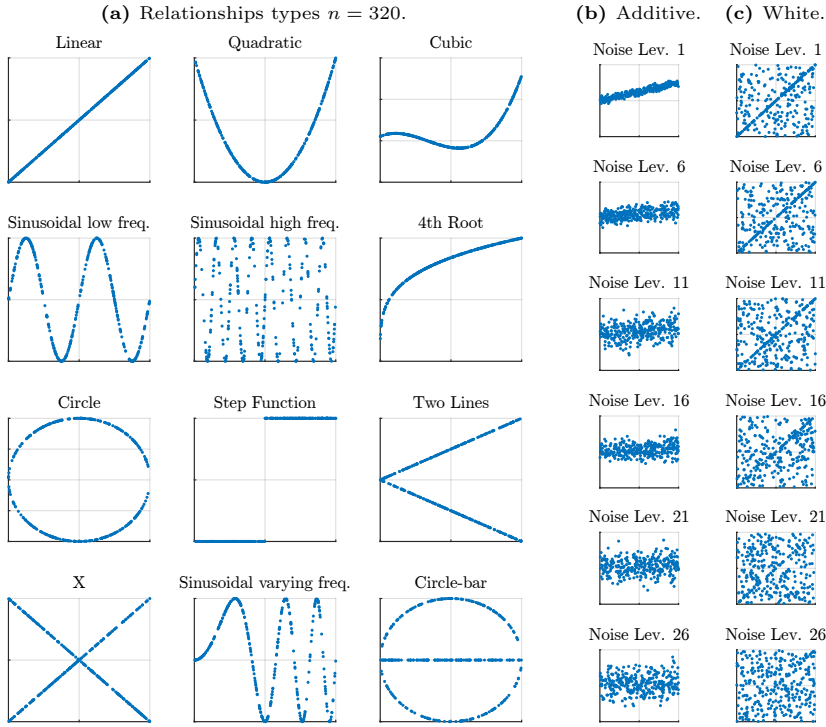
Family	Acr.	Name	Parameters
Mutual information estimators	I_{ew}	Mutual Information (Discretization equal width)	$D = \lfloor \sqrt{n/5} \rfloor$
	I_{ef}	Mutual Information (Discretization equal frequency)	$D = \lfloor \sqrt{n/5} \rfloor$
	I_A	Mutual Information (Adaptive Partitioning)	-
	I_{mean}	Mutual Information (Mean Nearest Neighbours)	-
	I_{KDE}	Mutual Information (Kernel Density Estimation)	$h_0 = n^{-1/6}$
Correlation based	r^2	Squared Pearson's Correlation	-
	ACE	Alternative Conditional Expectation	$\epsilon = 10^{-12}$
	dCorr	Distance Correlation	-
	RDC	Randomized Dependency Coefficient	$k = 20, s = 1/6$
Kernel based	HSIC	Hilbert-Schmidt Independence Criterion	$\sigma_X, \sigma_Y = \text{med. dist.}$
Information theory based	MIC	Maximal Information Coefficient	$\alpha = 0.6$
	MIC_e	Maximal Information Coefficient	$\alpha = 0.6$
	GMIC	Generalized Mean Information Coefficient	$\alpha = 0.6, p = -1$
	MID	Mutual Information Dimension	-
	TIC_e	Total Information Coefficient	$\alpha = 0.65$
	RIC	Randomized Information Coefficient	$K_r = 20, D_{\max} = \lfloor \sqrt{n} \rfloor$

Table 4 Dependency measures compared in this paper and parameters used in the tasks of network inference, feature filtering for regression, and estimation of running times.

5.1 Identification of Noisy Relationships

We consider the task of discriminating between noise and a noisy relationship, i.e., determining whether a dependency exists by testing for independence between X and Y , across a large number of dependency types. In Figure 5(a), 12 different relationships between X and Y are induced on $n = 320$ data points.

We use the same setting as in Simon & Tibshirani (2011). In this study, the measure performance on a relationship is assessed by power at level $\alpha = 0.05$. For each test case, we generated 500 random data sets with X and Y being completely independent. These constitute the *negative* class or the *complete noise* class. Then, for each noise level between 1 and 30, we generate 500 other data sets to create the *positive* class or the *noisy relationship* class. We evaluate the ability of different measures to discriminate between complete noise and the noisy relationship classes by computing the power (sensitivity) for the positive class at level $\alpha = 0.05$. Experiments were carried out on two different noise models, namely *additive noise*

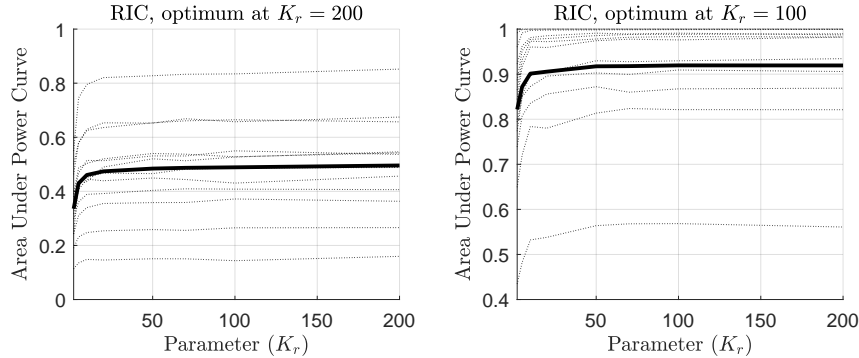
Fig. 5 Relationships between two variables and example of *additive* and *white* noise.

and *white noise*. In the first scenario we add different levels of Gaussian noise by varying the noise standard deviation σ_η . In the second scenario we substitute some points of the relationship with uniform noise. Figures 5(b) and 5(c) show examples of noise levels for the linear relationship in the additive noise model and white noise model respectively. Given that all measures present good discrimination ability in the white noise model, level 1 (lowest noise) starts by assigning 40% of points to the relationship and 60% to uniformly distributed noise.

Given that the default parameters of each measure are not tuned for independence testing, we decided to follow the approach of Reshef et al. (2015a): for a particular noise model we identify the parameters that maximize the average power for all level of noise and all relationship types. This analysis can be found in the Appendix A. For example, a bigger number of nearest neighbors is beneficial to I_{kNN} to achieve more power under the additive noise model (Kraskov et al., 2004). Furthermore, measures which make use of kernels should employ kernels with larger width to maximize power under additive noise. Even though these parameters cannot be tuned on a new data set because the task is unsupervised, the analysis provided in Appendix A might guide the user when the particular noise model is known for a data set. As discussed in Section 4.1, increasing K_r for RIC helps to decrease its variance. This is particularly important in order to achieve power when testing against independence. Figure 6(a) and 6(b) show the

Fig. 6 Each line is the area under the power curve for each relationship tested in this paper. The solid black line shows their average. All results are shown at the variation of the parameter K_r for RIC which determines the number of random discretization grids. The power against independence of RIC always increases at the increase of K_r because its variance decreases. This is particularly important when the number of data points is small: e.g., $n = 100$.

(a) Power of RIC at small sample size: $n = 100$. (b) Power of RIC on larger samples: $n = 1,000$.



area under power curve for each relationship tested in this paper and their average at the variation of K_r for RIC. Increasing K_r is very beneficial to increase power when the number of data points is small: $n = 100$. This is an interesting feature of RIC. Increasing K_r gives more power but also increases the computational running time. Nonetheless, higher K_r is needed only if the sample size is small.

We show the performance of RIC with $D_{\max} = \lfloor \sqrt{n/4} \rfloor$ and $K_r = 200$ as obtained by parameter tuning. Detailed results for each relationship types are provided in Section A of the Appendix. Note that because not all the relationships in Figure 5(a) are functional, it is not possible to plot power against a normalized x -axis as in Reshef et al. (2015a). In Reshef et al. (2015a) the power of functional relationships is plotted against the R^2 between the true underline function between variables and its noisy version. In this paper, we follow the approach in Simon & Tibshirani (2011) where the x -axis represents some non-normalized amount of noise added to the relationship between the variables. Therefore, the amount of noise for a particular value on the x -axis and a particular relationship is not comparable with the amount of noise added to another relationship at the same point on the x -axis. Nonetheless on our set of relationships, we would like to point out that all the power plots are monotonically decreasing and they do not look to be intersecting each other. In particular, if a dependency measure \mathcal{D}_1 shows higher power than a measure \mathcal{D}_2 at a given level of noise, \mathcal{D}_1 will also have higher power than \mathcal{D}_2 at a higher level of noise. Please refer to Figure 20 in the Appendix A.

In order to compare the different measures on multiple data sets (relationships) we use the framework proposed in Demšar (2006): we show the average rank across data sets for each measure. According to this framework we compare a statistic based on ranking which is not influenced by the absolute value of the metric of performance. Therefore, this evaluation is not influenced by the fact the x -axis in the power plots cannot be normalized. Moreover, in order to provide graphical intuition about their performance, we show their average rank sorted in ascending

order using bar plots. Figure 7 present the performances on the *additive noise model*. RIC computed with $D_{\max} = \lfloor \sqrt{n/4} \rfloor$ and $K_r = 200$ shows very competitive performance.

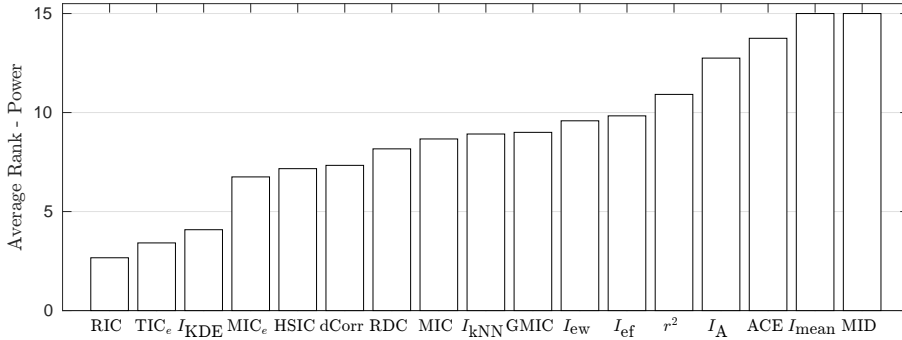


Fig. 7 Average rank of measures across relationships when the target is power maximization under the additive noise model. RIC with $D_{\max} = \lfloor \sqrt{n/4} \rfloor$ and $K_r = 200$ is very competitive in this scenario.

RIC outperforms all mutual information estimators, in particular the discretization based I_{ew} , I_{ef} , I_A , and the k NN based I_{kNN} . The kernel based density estimator I_{KDE} looks more competitive in noisy scenarios than all the other mutual information estimators, as also pointed out in Khan et al. (2007). The performance of I_{mean} is particularly surprising: even if I_{mean} is a smooth estimator of mutual information, which guarantees low variance, it cannot discriminate very noisy relationships well. A careful look at its derivation reveals that I_{mean} takes into account k NN with k very large, e.g. $k = n - 1$. In fact, even I_{kNN} in this case cannot discriminate between noise and noisy relationships. MIC with parameters optimized for independence testing shows to outperform distance correlation. MIC outperforms GMIC with parameter $p = -1$ when the parameter α is tuned independently for both of them. In particular, MIC obtains its optimum at $\alpha = 0.35$ and GMIC at $\alpha = 0.65$. The comparison carried out in Luedtke & Tran (2013) considered $\alpha = 0.6$ for both measures, concluding that GMIC was superior with this setting. Moreover, the new version MIC_e shows to improve on MIC results. The new information theoretic based measure MID also presents less competitive discrimination ability on this set of relationships because it is better suited for the white noise model. RDC achieves good results overall, in particular on the scenarios where it seems possible to linearize the relationship via a random projection: low frequency sinusoids and circle relationships. When the relationship is linear, r^2 is the best choice in terms of discrimination ability. This property reflects the motivation for dCorr, which was proposed as a distance based extension to the non-linear scenarios: its performance is very competitive on the linear, and 4th root case. However, it fails in the high frequency sinusoidal case and circle relationship. The best measure among the competitors is the newly proposed TIC_e which is explicitly designed for independence testing. These results indicate that when the purpose is to identify an arbitrary relationship in the additive noise scenario, RIC delivers extremely competitive performance on average. If the user

is interested in a specific relationship type, it will be best to choose a particular dependency measure known to be specifically good for that scenario. The results in the appendix may help guide the user in this choice.

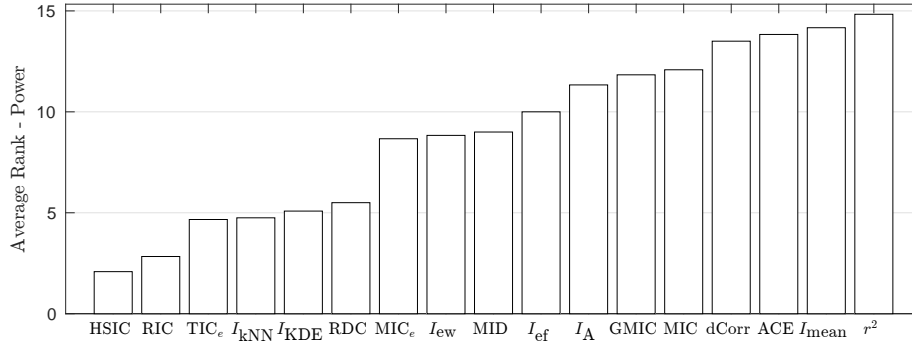


Fig. 8 Average rank of measures across relationships when the target is power maximization under the white noise model. RIC with $D_{\max} = \lfloor \sqrt{n} * 10 \rfloor$ and $K_r = 200$ is competitive but yet it is outperformed by HSIC.

RIC also shows good performance under the *white noise* model but it is outperformed by HSIC. Average results are shown in Figure 8. The optimal parameters under white noise are different from the optimal parameters under additive noise for many measures as shown in Appendix A. Regarding the grid estimators of mutual information, RIC, TIC_e, MIC, MIC_e, and MID, a denser grid is better suited for the white noise scenario because points are uniformly distributed on the joint domain (X, Y) . I_{kNN} presents competitive performance under white noise when k is small. As in the additive noise model, TIC_e proved to be strong competitor to RIC in this scenario. Instead, dCorr seems to be little competitive under the white noise model. HSIC with very small kernel width performs the best under white noise.

5.2 Equitability

In this section, we assess the equitability of the measures discussed in this paper. A dependence measure is equitable if it provides similar scores to equally noisy relationships of different kinds, relative to some measure of noise (Reshef et al., 2011, 2015a,b). For example, in the case of functional relationships, one natural instantiation of equitability is for an equitable measure of dependence to assign similar scores to relationships with the same coefficient of determination R^2 between the true underlying function and its noisy version. Therefore, for functional relationships an equitable measure is 1 if the dependency between the variables is noiseless.

As in (Reshef et al., 2015a), we discuss results on two sets of functional relationships shown in Table 5: functional relationships from (Simon & Tibshirani, 2011) shown in Figure 5(a) that we employed in the power analysis in Section 5.1,

(Simon & Tibshirani, 2011)	
1	$y = x + \sigma \cdot \varepsilon$
2	$y = 4(x - 0.5)^2 + \sigma \cdot \varepsilon$
3	$y = 128(x - \frac{1}{3})^3 - 48(x - \frac{1}{3})^2 - 12(x - \frac{1}{3}) + \sigma \cdot \varepsilon$
4	$y = \sin(4\pi x) + \sigma \cdot \varepsilon$
(Reshef et al., 2011)	
1	$y = x + \sigma \cdot \varepsilon$
2	$y = 4(x - 0.5)^2 + \sigma \cdot \varepsilon$
3	$y = 4(2.3x - 1.3)^3 + (2.3x - 1.3)^2 - 4(2.3x - 1.3) + \sigma \cdot \varepsilon$
4	$y = \sin(8\pi x) + \sigma \cdot \varepsilon$

Table 5 The two different sets of functional relationships used in the equitability analysis. The first one comes from (Simon & Tibshirani, 2011) and the second one from (Reshef et al., 2011). In particular, they differ in the definition of the relationships 3 and 4: the cubic and the sinusoidal relationships. We add to every function a different amount of gaussian noise $\varepsilon \sim \mathcal{N}(0, 1)$ varying the multiplicative constant σ .

and the a set of relationships from (Reshef et al., 2011). These functional relationships are defined as follows: $y = f(x) + \sigma \cdot \varepsilon$ where $f(\cdot)$ is a function, σ is a constant, and ε is Gaussian noise with 0 mean and variance 1. In order to quantify if a measure is equitable, we estimate the R^2 between a given noiseless function and its noisy version on a data set of $n = 320$ points. We vary the amount of noise using a different constant σ . Each value for the R^2 is matched against the score obtained by a measure for the same noisy relationship at the same level of noise. Scatter-plots for each measure are shown in Figure 9. The narrower the scatter-plot for a measure, the more equitable a measure is.

In exploratory data analysis, often there is no ground-truth. For example, there is no ground-truth when the task is identifying the top pair of dependent variables among all the possible pairs. In this case, it is not possible to tune the parameters for a particular measure. In this analysis, we relied on the default values provided for the measures in the respective papers. These are shown in Table 4. With their default parameters, the best measures in terms of equitability are MIC and ACE. More specifically, ACE seems to consistently score noiseless functional relationships with a value 1 but seems to fail when the amount of noise increases. MIC and its improved version MIC_e instead show good equitability across the board. On the other hand, RIC is not an equitable measure. The scatter-plot for RIC is similar to GMIC and TIC_e scatter-plots. Indeed, all these measures use multiple grids to compute mutual information related statistics and aggregate their values. This aggregated grid-based approach seems to be more beneficial when the task is identify a relationship with high power.

We rank the measures in terms of equitability in Figure 10 and Figure 11. For each scatter-plot we identify the worst case for equitability: i.e. the maximum range of values for R^2 associated to a single value for a measure. That single value corresponds to two completely different levels of noise. For example, the Pearson correlation squared r^2 is equal to 0 for both a completely noiseless sinusoidal relationship and a completely noisy one. Indeed, r^2 is consistently ranked as last in Figure 10 and Figure 11. MIC_e shows to be the best overall in this task. Note also that MIC_e seems to better match the R^2 on this sets of relationships: it is very close to 0 when R^2 is 0 and 1 when R^2 . Other work in literature proposed to enforce this property using adjustment for chance (Romano et al., 2016; Wang

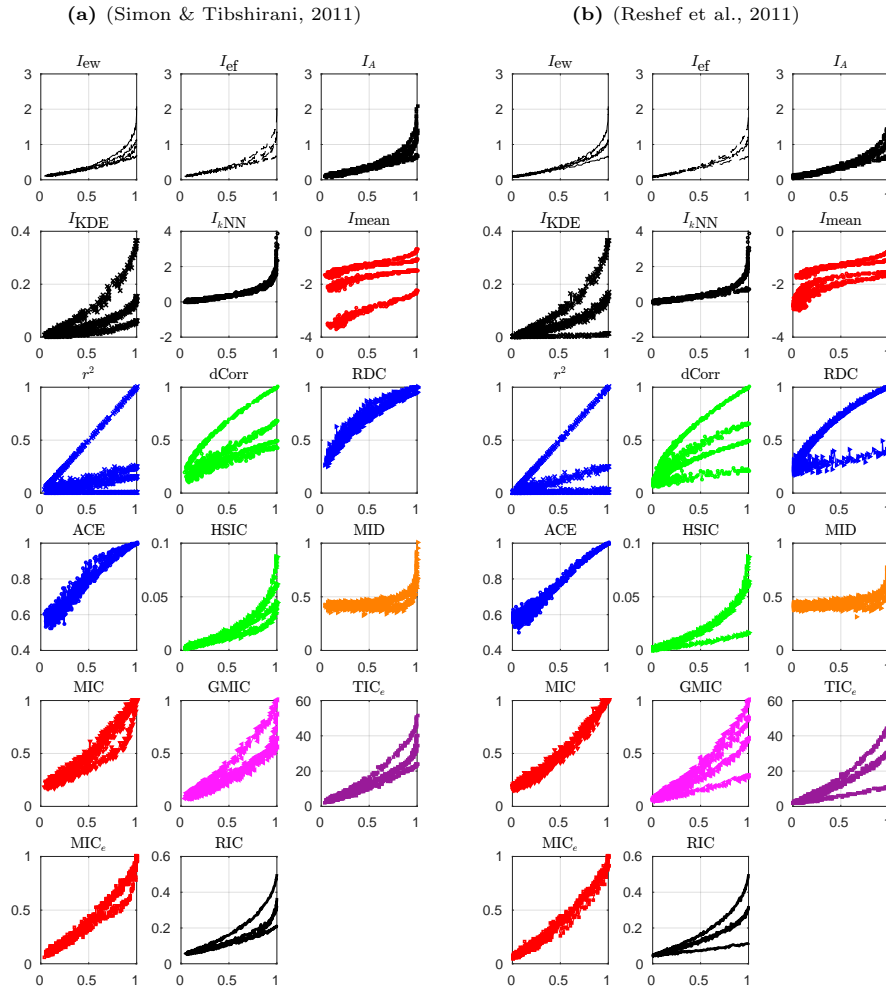


Fig. 9 Equitability of each measure on the two different sets of functional relationships from Table 5. Different values for the R^2 between the noiseless functional relationship and its noisy version are plotted against the value the dependency measure would obtain for that particular amount of noise. The narrower the scatter-plot is, the better the equitability.

et al., 2017). This is an important property which enables MIC_e to be used as proxy of the R^2 .

5.3 Application to Network Inference

We next employ the measures for biological network reverse engineering, which is a popular and successful application domain for dependency measures (Villaverde et al., 2013). The applications include cellular, metabolic, gene regulatory, and sig-

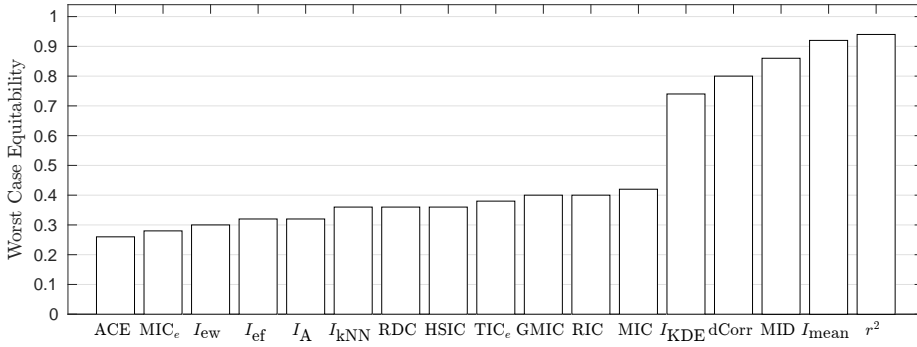


Fig. 10 Worst case equitability for relationships in (Simon & Tibshirani, 2011): i.e. the maximum range of values for R^2 associated to a single value for a measure. That single value corresponds to two completely different levels of noise.

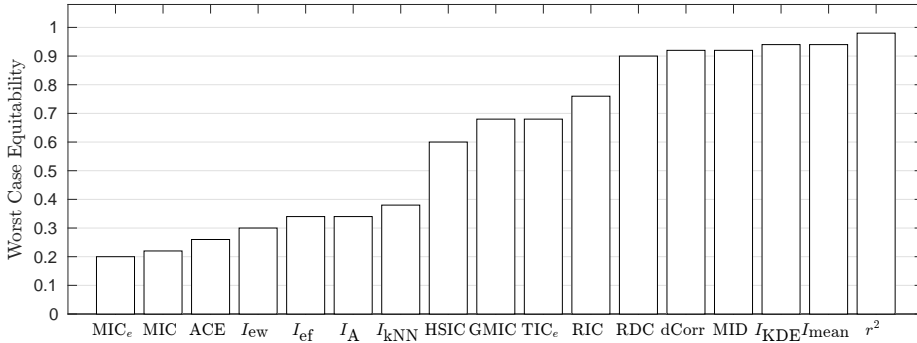


Fig. 11 Worst case equitability for relationships in (Reshef et al., 2011): i.e. the maximum range of values for R^2 associated to a single value for a measure. That single value corresponds to two completely different levels of noise..

nalling networks. Each of the m variables is associated with a time series of length n . In order to identify the strongest relationships between variables (e.g., genes), a dependency measure \mathcal{D} is employed. Due to the natural delay of biochemical interactions in biological networks, the strongest dependency might occur only after some time (Xuan et al., 2012). For this reason, we incorporate time delay into the dependency measures as $\mathcal{D}_{\text{delayed}} = \max_{\tau \in [-\tau_m, +\tau_m]} \mathcal{D}(X(t - \tau), Y(t))$, where \mathcal{D} is any measure from Table 4 and τ_m is the maximum time delay. We collected 10 datasets where the true interactions between the variables are known. A dependency measure is effective on this task if its output is high on real interactions (*positive* class) and low on non-interacting pairs of variables (*negative* class). We evaluate the performance of a measure with the average precision (AP), also known as the area under the precision-recall curve. In order to obtain meaningful comparisons and perform statistical hypothesis testing, we performed 50 bootstrap repetitions for each dataset and computed the mean AP (mAP) across the repetitions.

Network Inference Datasets				Regression Datasets			
#	Name	n	m	#	Name	n	m
1	Glycolysis	57	10	1	Pyrim	74	27
2	Enzyme-cat	250	8	2	Bodyfat	252	14
3	Small-chain	100	4	3	Triazines	186	60
4	Irma-on-off	125	5	4	Wisconsin	194	32
5	Mapk	210	12	5	Crime	111	144
6	Dream4-10	105	10	6	Pole	1000	48
7	Dream4-100	210	100	7	Qsar	384	482
8	SynT1	100	200	8	Qsar2	384	186
9	SynT1-s	30	200				
10	SynT2	30	40				

Table 6 Summary of the datasets used for network inference (left) and regression (right): n is the data points and m is the number of variables.

We made use of the MIDER framework (Villaverde et al., 2014) for evaluating the performance of dependency measures. The first 7 datasets were retrieved from the MIDER framework. The last 3 datasets were generated using SynTren (Van den Bulcke et al., 2006), a generator of synthetic gene expression data. **SynT1** and **SynT1-s** were generated starting from the Escherichia coli transcriptional regulatory network provided with the framework with default noise parameters where **SynT1-s** has shorter time series. **SynT2** was generated starting from the synthetic direct acyclic graph provided with the framework. Based on the data sampling rate, we set $\tau_m = 3$ for these datasets, which cover most plausible time-delayed interactions. Table 6 shows a summary of the datasets used.

The small amount of data available and the high amount of noise in biological time series posed a very challenging task for all statistics. Mutual information estimators have been extensively employed for this task (Villaverde et al., 2013). Just recently, HSIC has been tested on network inference (Lippert et al., 2009) and even more recently dCorr has been shown to be competitive on this task (Guo et al., 2014). In this task, it is important to powerfully discriminate between independent and non-linearly dependent variables. Indeed, measures with high power as discussed in Section 5.1 might have an advantage (Guo et al., 2014). Of course, a measure can be even more competitive if it is also equitable. Nonetheless, this is a different task from equitability assessment and equitability is only part of the picture. This explains the performance of dCorr and HSIC in the literature (Lippert et al., 2009; Guo et al., 2014).

To our knowledge, there is no prior comprehensive survey of the performance of RDC, I_{mean} , MIC, MIC_e , GMIC and MID on this task. We perform a comprehensive evaluation of RIC plus 16 other dependency measures on network inference. The results are shown in Table 7.

We use RIC with parameters $D_{\text{max}} = \lfloor \sqrt{n} \rfloor$ and $K_r = 20$ because on these tasks it is important to achieve high discrimination between strong relationships as well as weak relationships. Figure 12 presents the average rank of the measures across all tested networks. Overall, RIC performs consistently well across all datasets. It outperforms by far all the discretization based mutual information estimators as well as other information theoretic based measures including MIC, GMIC and MID. Among the mutual information estimators, I_{KDE} and $I_{k\text{NN}}$ show

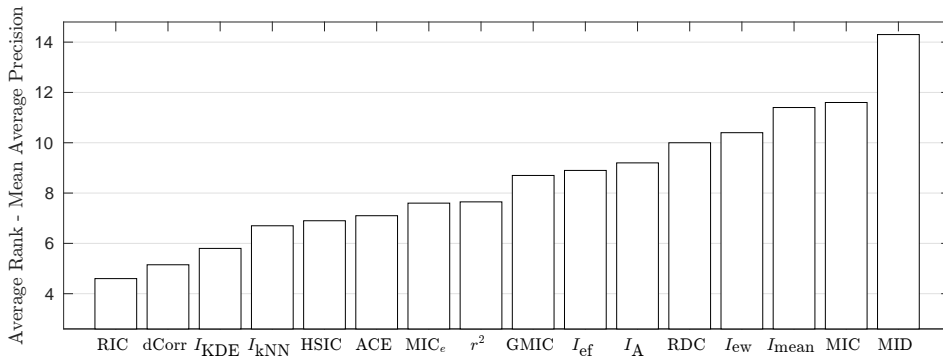


Fig. 12 Average rank across networks on the task of biological network with time inference. RIC outperforms on average all other measures.

very good results. RIC’s main competitor was dCorr, which also shows very good performance mainly due to the crucial importance of the linear relationships between variables. Its results are very correlated with r^2 results, which in some cases provides the best result for a single data set. This is mainly due to its high ability to discriminate linear relationships well. We found RIC particularly competitive on short time series with a large number of variables.

As well known within the machine learning community, there is no “free lunch”. In the context of this application, this wisdom is evident, observing in Table 7 that no method always performs the best or worst in every case. MID for example, is badly affected by additive noise commonly observed in biological time series and thus showed overall less competitive performance. Nonetheless, it achieved the best performance on **Irma-on-off**, an *in vivo* yeast semi-synthetic network.

We want to reiterate that this is an unsupervised task. Therefore it is not possible to tune parameters with cross-validation on a given data set. The tasks of inference of network of variables and identification of noisy relationships are unsupervised learning tasks and do not allow parameter tuning when applied to a new data set. When the user is provided with a new data set, this can only rely on the default parameters of a measure. Of course these could be tweaked to identify different top pairs of relationships. Nonetheless, all different sets of top relationships obtained with different parameters should be inspected individually because no ground truth is available for validation. Moreover, the real data sets discussed in this paper are a sample of the possible real data sets that can be analysed. Yet our data sets are independent and the analysis discussed in the paper can provide a picture of the behavior of different measures with default parameters.

5.4 Feature Filtering for Regression

In this section, we evaluate the performance of RIC and the other statistics as feature filtering techniques. A dependency measure \mathcal{D} can be used to rank the m features X_i on a regression task based on their prediction ability for the target variable Y . Only the top m^* features according to \mathcal{D} are used to build a regressor

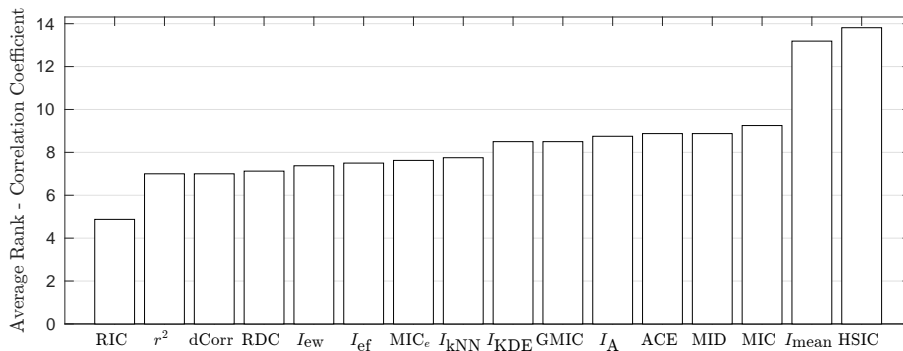


Fig. 13 Average rank of measures when the task is maximizing the correlation coefficient between the predicted and the target value of a k NN regressor. The k NN regression is built on top of m^* features. Results were averaged across $m^* \leq 10$ and all datasets.

for Y . Table 8 shows the average correlation coefficient between the predicted and the actual target value using the top $m^* \leq 10$ features using a k -NN regressor (with $k = 3$). Each value is obtained by averaging 3 random trials of 10-fold cross-validation for each $m^* \leq 10$.

The datasets collected have at least 10 features and in the case of $n > 1000$ records, we randomly sampled 1000 records to speed up the running time of dCorr, HSIC, I_{KDE} , I_{mean} , MIC, MIC_e , and GMIC with default parameters. Records with missing values were deleted. We analyzed the performance on 8 datasets: 5 from the UCI machine learning repository¹⁰, the **Pole** telecommunication data¹¹, and 2 datasets **Qsar** and **Qsar2** from the website of the 3rd Strasbourg Summer School on Chemoinformatics¹². The list of datasets used is shown in Table 6.

As for the task of network inference in Section 5.3, it is important for a measure to be both equitable and powerful when detecting relationships. Powerful measures despite being non-equitable have been shown to perform well on this task: e.g. the HSIC (Song et al., 2007).

As in section 5.3 we use RIC with parameters $D_{max} = \lfloor \sqrt{n} \rfloor$ to avoid low density grids that are better suited for testing of independence tasks. Overall, as can be observed from Figure 13, RIC performs consistently well on average. RIC is also particularly useful when the number of features m is high and especially when their relationships to the target variable Y are noisy. These represent the most challenging scenarios as can be justified by the low correlation coefficient achievable using the selected features, e.g., on the **Pyrim** and **Triazines** datasets. We also note the good performance of RIC on datasets where there are features that can take only a predefined number of values: e.g., discrete numerical features. **Pole**, **Qsar**, and **Qsar2** include these type of features. For such features it is very difficult to either optimize a kernel or a grid size or find the optimal data transformation to obtain the maximal correlation with ACE, which explains the less competitive performance of HSIC, I_{KDE} , I_A , I_{mean} , and ACE. RIC is not affected by this problem as there is no optimization and grids are generated at

¹⁰ <http://ics.uci.edu/~mllearn>

¹¹ <http://tunedit.org/>

¹² <http://infochim.u-strasbg.fr/>

random. Note that the good performance on feature selection for RIC is also due to the fact that features with high entropy are penalized because of the normalization factor in Eq. (2).

5.5 Run Time Comparison

Here we compare the running times of each measure in Table 4 varying the amount of records n on two independent variables X and Y uniformly distributed. The average run time on 30 simulations is shown in Figure 14(a) for each measure. RIC is very competitive in terms of speed and can be grouped with the fastest measures: I_{ef} , I_{ew} , I_{kNN} , I_A , r^2 , MID, ACE, and RDC. On the other hand, dCorr, I_{KDE} , HSIC, I_{mean} , MIC, MIC_e , GMIC, and TIC_e appear to be slower according to the implementations discussed at the beginning of Section 5 and the parameter setting from Table 4. As discussed in the related work section, different parameter setting yield more competitive running times for some measures. For example, TIC_e can obtain close to linear complexity in the number of records if $\alpha = 0.2$. In our analysis, we chose to set $\alpha = 0.65$ because it is the choice that allows us to maximize power when testing for independence under additive noise.

Fig. 14 Running time in seconds (best viewed in color).

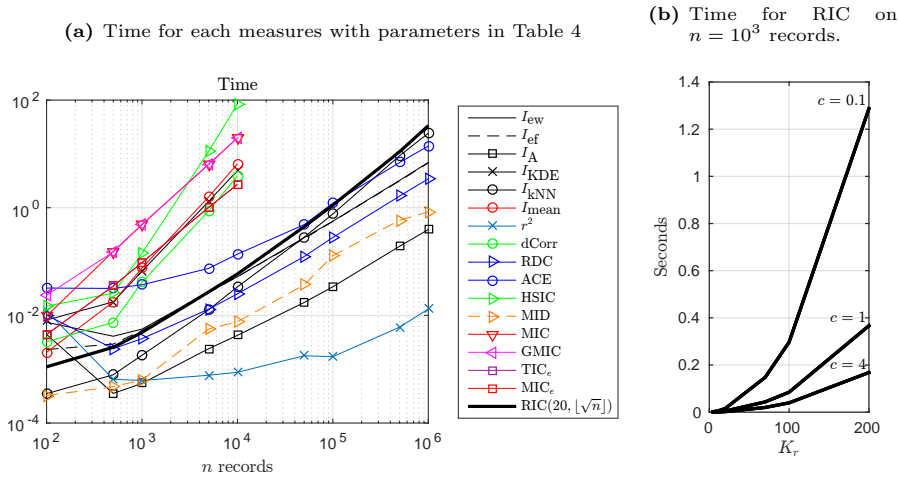


Figure 14(a) shows the running time for RIC with default parameters $K_r = 20$ and $D_{\max} = \lfloor\sqrt{n}\rfloor$. Similarly to other measures, the running time for RIC depends to its parameter setting. Figure 14(b) shows the different time taken by RIC on $n = 10^3$ records according to different K_r and different c where $D_{\max} = \lfloor\sqrt{n/c}\rfloor$. By increasing K_r we increase the number of random grids and by increasing c with decrease the grid coarsity. Figure 14(b) shows different plots at the variation of K_r for $c = 4$, $c = 1$, and $c = 0.1$ which respectively yield to $D_{\max} = \lfloor\sqrt{n/4}\rfloor$, $D_{\max} = \lfloor\sqrt{n}\rfloor$, and $D_{\max} = \lfloor\sqrt{n \cdot 10}\rfloor$. These settings are respectively the ones we

used for: independence testing under additive noise; network inference and feature filtering; and independence testing under white noise. The latter scenario proved to be the most challenging in terms of RIC running time.

Large K_r increases the computational time. Nonetheless, large K_r is not always required. As discussed in Section 5.1 even though it is always beneficial to increase K_r to further decrease the variance of RIC, this is particularly important when n is small. Thus, K_r can always be tuned by the user according to the sample size of the data set analyzed and the disposable computational budget.

Table 7 Mean average precision (mAP) on 10 networks: n length of time series; m number of variables. Each cell shows mAP \pm std and either (+),(=), or (-) means statistically greater, equal, or smaller according to the 1-sided paired t -test ($\alpha = 0.05$) than RIC results.

	Glycolysis	Enzyme-cat	Small-chain	Irma-on-off	Mapk	Dream4-10	Dream4-100	SynT1	SynT1-s	SynT2
(n, m)	(57, 10)	(250, 8)	(100, 4)	(125, 5)	(210, 12)	(105, 10)	(210, 100)	(100, 200)	(30, 200)	(30, 40)
RIC	67.5 \pm 3.7	91.4 \pm 2.0	91.4 \pm 1.2	70.5 \pm 3.5	57.6 \pm 2.2	64.6 \pm 6.1	10.3 \pm 0.7	7.9 \pm 0.4	6.6 \pm 0.7	14.1 \pm 3.8
dCorr	67.8 \pm 3.0(=)	88.6 \pm 2.3(-)	91.7 \pm 0.0(+)	68.6 \pm 2.6(-)	50.0 \pm 1.0(-)	68.8 \pm 5.9(+)	12.6 \pm 0.6(+)	7.4 \pm 0.3(-)	6.7 \pm 0.6(=)	16.0 \pm 2.5(+)
J_{KDE}	67.9 \pm 3.0(=)	93.5 \pm 2.1(+)	88.1 \pm 8.1(+)	71.5 \pm 5.6(=)	59.0 \pm 3.1(+)	61.1 \pm 7.0(-)	9.7 \pm 0.7(-)	7.8 \pm 0.3(-)	6.4 \pm 0.6(=)	10.2 \pm 1.8(-)
J_{kNN}	65.5 \pm 4.6(-)	91.5 \pm 3.8(=)	90.0 \pm 5.1(-)	72.7 \pm 6.6(+)	68.8 \pm 2.0(+)	51.4 \pm 6.1(-)	8.5 \pm 0.8(-)	7.8 \pm 0.4(-)	7.0 \pm 0.8(+)	10.3 \pm 1.8(-)
r^2	68.4 \pm 2.7(=)	86.0 \pm 1.9(-)	91.7 \pm 0.0(+)	69.0 \pm 2.9(-)	46.9 \pm 1.3(-)	56.7 \pm 6.1(-)	12.5 \pm 0.6(+)	6.7 \pm 0.4(-)	6.3 \pm 0.6(+)	14.5 \pm 2.3(=)
RDC	61.6 \pm 7.6(-)	89.2 \pm 4.4(-)	84.4 \pm 9.9(-)	68.3 \pm 5.3(-)	63.0 \pm 2.7(+)	57.8 \pm 4.7(-)	11.3 \pm 0.9(+)	6.8 \pm 0.7(-)	4.2 \pm 0.5(-)	10.4 \pm 2.7(-)
J_A	62.4 \pm 4.6(-)	89.3 \pm 3.5(-)	82.7 \pm 10.8(-)	70.9 \pm 5.6(=)	57.7 \pm 3.5(-)	61.1 \pm 7.0(-)	9.4 \pm 0.6(-)	7.4 \pm 0.4(-)	4.6 \pm 0.6(-)	10.3 \pm 2.4(-)
J_{ef}	62.3 \pm 4.3(-)	91.7 \pm 3.6(=)	86.4 \pm 7.5(-)	73.0 \pm 5.1(+)	56.0 \pm 3.0(-)	61.1 \pm 7.0(-)	10.3 \pm 0.6(-)	7.2 \pm 0.5(-)	4.5 \pm 0.7(-)	10.2 \pm 2.5(-)
J_{ew}	63.5 \pm 4.7(-)	78.2 \pm 7.6(-)	90.9 \pm 2.1(=)	69.8 \pm 4.3(+)	50.9 \pm 2.2(-)	58.4 \pm 8.4(-)	10.1 \pm 0.8(-)	6.8 \pm 0.5(-)	4.0 \pm 0.4(-)	9.1 \pm 1.4(-)
MIC	66.5 \pm 4.5(=)	94.3 \pm 4.4(+)	86.1 \pm 8.3(+)	71.2 \pm 5.5(+)	55.3 \pm 6.8(-)	56.9 \pm 6.1(-)	11.1 \pm 0.8(+)	7.4 \pm 0.5(-)	5.5 \pm 0.6(-)	11.7 \pm 2.5(-)
GMIC	64.4 \pm 4.9(-)	89.0 \pm 9.6(-)	84.9 \pm 10.5(-)	68.8 \pm 3.7(-)	45.1 \pm 6.8(-)	56.1 \pm 8.8(-)	11.4 \pm 0.8(+)	6.8 \pm 0.5(-)	5.5 \pm 0.7(-)	10.1 \pm 1.9(-)
GMIC	66.6 \pm 4.2(=)	89.0 \pm 3.8(-)	90.3 \pm 3.0(-)	71.2 \pm 5.5(+)	53.5 \pm 6.6(-)	57.2 \pm 4.2(-)	10.4 \pm 0.7(=)	7.3 \pm 0.5(-)	5.7 \pm 0.7(-)	12.5 \pm 2.8(-)
MID	35.7 \pm 8.6(-)	47.4 \pm 11.7(-)	75.2 \pm 14.8(-)	68.8 \pm 3.7(-)	37.5 \pm 4.8(-)	39.6 \pm 6.2(-)	3.9 \pm 0.4(-)	2.3 \pm 0.4(-)	1.6 \pm 0.1(-)	8.8 \pm 1.5(-)
ACE	67.4 \pm 5.0(=)	88.5 \pm 6.2(-)	84.4 \pm 10.7(-)	75.6 \pm 7.0(+)	62.0 \pm 2.0(+)	53.8 \pm 7.0(-)	9.9 \pm 0.8(-)	7.4 \pm 0.4(-)	6.1 \pm 0.6(-)	11.0 \pm 2.4(-)
HSC	64.8 \pm 3.7(-)	87.7 \pm 3.8(-)	91.5 \pm 1.2(=)	68.4 \pm 2.4(-)	51.9 \pm 2.3(-)	64.5 \pm 5.7(=)	9.9 \pm 0.9(-)	7.5 \pm 0.7(-)	7.1 \pm 1.1(+)	11.7 \pm 2.4(-)
J_{mean}	46.8 \pm 2.0(-)	90.2 \pm 0.0(-)	91.6 \pm 0.7(=)	69.6 \pm 3.4(=)	33.0 \pm 1.4(-)	65.4 \pm 5.6(=)	8.1 \pm 0.5(-)	4.6 \pm 0.2(-)	2.7 \pm 0.2(-)	7.5 \pm 0.9(-)

Table 8 Correlation coefficient between the predicted and actual target value on 8 datasets using k NN ($k = 3$). The values are the mean correlation coefficient across the k NN regressors built on top of m^* features with $m^* \leq 10$. n number of records; m number of features. Each cell shows mean correlation coefficient \pm std and either (+),(=), or (-) means statistically greater, equal, or smaller according to the 1-sided paired t -test ($\alpha = 0.05$) than RIC results.

	Pyrim	Bodyfat	Triazines	Wisconsin	Crime	Pole	Qsar	Qsar2
(n, m)	(74, 27)	(252, 14)	(186, 60)	(194, 32)	(111, 144)	(1000, 48)	(384, 482)	(384, 186)
RIC	0.261 \pm 0.120	0.642 \pm 0.115	0.215 \pm 0.120	0.034 \pm 0.013	0.892 \pm 0.042	0.685 \pm 0.156	0.277 \pm 0.091	0.479 \pm 0.053
dCorr	0.205 \pm 0.046(-)	0.643 \pm 0.114(=)	0.118 \pm 0.062(-)	0.041 \pm 0.012(=)	0.852 \pm 0.057(-)	0.686 \pm 0.218(=)	0.310 \pm 0.025(+)	0.382 \pm 0.130(-)
J_{KDE}	0.231 \pm 0.068(-)	0.635 \pm 0.117(-)	0.148 \pm 0.095(-)	0.039 \pm 0.012(=)	0.614 \pm 0.047(-)	0.686 \pm 0.221(=)	0.291 \pm 0.029(=)	0.424 \pm 0.151(-)
J_{kNN}	0.216 \pm 0.051(-)	0.639 \pm 0.116(-)	0.098 \pm 0.035(-)	0.038 \pm 0.011(=)	0.893 \pm 0.051(=)	0.621 \pm 0.134(-)	0.300 \pm 0.094(+)	0.423 \pm 0.025(-)
r^2	0.264 \pm 0.064(=)	0.644 \pm 0.114(=)	0.125 \pm 0.050(-)	0.041 \pm 0.009(+)	0.870 \pm 0.045(-)	0.414 \pm 0.311(=)	0.273 \pm 0.037(-)	0.375 \pm 0.134(-)
RDC	0.206 \pm 0.052(-)	0.642 \pm 0.115(=)	0.199 \pm 0.079(=)	0.017 \pm 0.008(-)	0.891 \pm 0.042(=)	0.679 \pm 0.197(=)	0.280 \pm 0.058(=)	0.430 \pm 0.060(-)
J_A	0.235 \pm 0.088(-)	0.640 \pm 0.115(=)	0.062 \pm 0.047(-)	0.037 \pm 0.021(=)	0.891 \pm 0.041(=)	0.010 \pm 0.007(-)	0.284 \pm 0.104(=)	0.418 \pm 0.046(-)
J_{ef}	0.190 \pm 0.068(-)	0.641 \pm 0.115(=)	0.171 \pm 0.053(-)	0.036 \pm 0.015(=)	0.889 \pm 0.059(-)	0.661 \pm 0.145(-)	0.278 \pm 0.104(=)	0.429 \pm 0.028(-)
J_{ew}	0.249 \pm 0.064(=)	0.642 \pm 0.114(=)	0.188 \pm 0.097(-)	0.033 \pm 0.007(=)	0.859 \pm 0.040(-)	0.693 \pm 0.156(=)	0.264 \pm 0.085(-)	0.441 \pm 0.046(-)
MIC	0.186 \pm 0.072(-)	0.641 \pm 0.115(=)	0.051 \pm 0.023(-)	0.010 \pm 0.009(-)	0.776 \pm 0.040(-)	0.694 \pm 0.156(=)	0.293 \pm 0.030(=)	0.448 \pm 0.039(-)
GMIC	0.187 \pm 0.088(-)	0.641 \pm 0.115(=)	0.158 \pm 0.067(-)	0.028 \pm 0.008(-)	0.819 \pm 0.059(-)	0.774 \pm 0.180(+)	0.301 \pm 0.046(+)	0.428 \pm 0.056(-)
GMIC	0.206 \pm 0.069(-)	0.634 \pm 0.118(=)	0.141 \pm 0.056(-)	0.026 \pm 0.005(-)	0.808 \pm 0.055(-)	0.734 \pm 0.179(+)	0.292 \pm 0.058(=)	0.468 \pm 0.054(-)
MID	0.241 \pm 0.167(=)	0.605 \pm 0.137(-)	0.160 \pm 0.062(-)	0.047 \pm 0.030(+)	0.178 \pm 0.047(-)	0.808 \pm 0.215(+)	0.194 \pm 0.130(-)	0.186 \pm 0.074(-)
ACE	0.221 \pm 0.051(-)	0.641 \pm 0.115(=)	0.111 \pm 0.073(-)	0.011 \pm 0.008(-)	0.894 \pm 0.042(=)	0.000 \pm 0.000(-)	0.270 \pm 0.056(=)	0.439 \pm 0.023(-)
HSC	0.174 \pm 0.068(-)	0.638 \pm 0.116(=)	0.057 \pm 0.063(-)	0.028 \pm 0.011(=)	0.853 \pm 0.046(-)	0.000 \pm 0.000(-)	0.001 \pm 0.001(-)	0.000 \pm 0.000(-)
J_{mean}	0.178 \pm 0.073(-)	0.636 \pm 0.117(-)	0.073 \pm 0.076(-)	0.034 \pm 0.011(=)	0.853 \pm 0.046(-)	0.000 \pm 0.000(-)	0.001 \pm 0.001(-)	0.001 \pm 0.000(-)

6 Experiments on Dependency Between Two Sets of Variables

In this section, we perform comparisons between the performance of measures which quantify the dependency between two sets of p variables \mathbf{X} and q variables \mathbf{Y} . This is different from finding a subset of variables that are significantly correlated. In that case, new advances in that area yielded interesting measures to compare (Nguyen et al., 2014a; Nguyen & Vreeken, 2015). In our paper, we compare the measures discussed in Table 4. The Pearson’s correlation coefficient, ACE, I_A , MIC, GMIC, and MID are not applicable in these scenarios and there is no straightforward method to extend them to sets of variables available in literature.

6.1 Identification of Multi-variable Noisy Relationships

Here we extend the experiments of section 5.1 to sets of variables \mathbf{X} and \mathbf{Y} . In particular, we test the power in identifying relationships between \mathbf{X} with $p = 3$ variables and a single variable Y with the additive noise model. In order to use the same 12 relationships displayed in Figure 5(a), we map the set of features \mathbf{X} on a single feature $X' = \frac{X_1 + \dots + X_p}{p}$ and obtain Y according a given relationship plus additive noise. Figure 15 shows an example of a quadratic relationships between Y and $\mathbf{X} = (X_1, X_2)$ ($p = 2$) with additive noise.

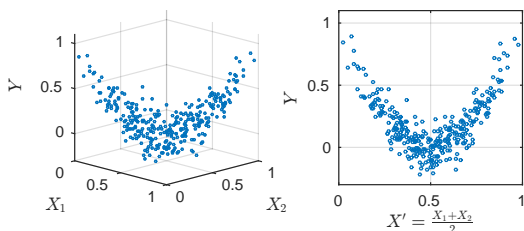


Fig. 15 Example of a quadratic relationship between Y and $\mathbf{X} = (X_1, X_2)$ on the left plot. The plot on the right shows how Y is obtained through the mapping of \mathbf{X} into $X' = \frac{X_1 + X_2}{2}$.

We fix the number of variables $p = 3$ for \mathbf{X} because some measures require specific tuning in regards to the number of variables considered. For example, the most straightforward way to extend the discretization based estimators of mutual information I_{ew} and I_{ef} is to independently discretize all the variables in each set. This requires carefully choosing the number of discretization bins for each variable in \mathbf{X} and each variable in \mathbf{Y} . If the same number of bins D_X is chosen for all the variables in \mathbf{X} and the same number of bins D_Y is chosen for all the variables in \mathbf{Y} , it is possible to end up with as many as $D_X^p \cdot D_Y^q$ total bins. This issue makes it practically infeasible to use I_{ew} and I_{ef} in high p, q scenarios. Given this limitation of the discretization based estimators of mutual information, we also made use of a multi-variable discretization approach of the set of variables \mathbf{X} which allows a more sensible choice of the total number of bins. Even if methods for multi-variable discretization are available in literature (Garcia et al., 2013; Dougherty et al., 1995) to our knowledge there is no extensive survey about the performance

of estimation of mutual information with multi-variable approaches. Therefore, we chose to discretize \mathbf{X} and \mathbf{Y} with the clustering algorithm k -means and then compute the mutual information. We name this measure $I_{k\text{-mean}}$. This allows us to choose the total number of bins (clusters) to be produced.

In our case, where $p = 3$ and $q = 1$ we chose compute I_{ew} and I_{ef} fixing $D_Y = 5$ and compute D_X in order to limit the number of total bins in regards to the number n of data points: $D_X^p \cdot D_Y \leq \frac{n}{5} \Rightarrow D_X = \lfloor \frac{\log n/25}{\log p} \rfloor$. When $n = 320$, $D_X = 2$. We tuned the parameters of every other measure in order to maximize the average power on all relationships. Please refer to Appendix B for more details. Regarding RIC, in order to have full control on the number of bins produced, we compared the multi-variable discretization approach that uses random seeds as described in Algorithm 4. More specifically, we fixed the number of random seeds to $\lfloor \sqrt{n/c} \rfloor$ given that also choosing the number of random seeds at random might result in configurations with as few as 2 seeds, which strongly deteriorates the discrimination ability of mutual information on multiple variables. The parameter c for RIC that maximizes the power on average is $c = 6$ which generates $\lfloor \sqrt{n/6} \rfloor$ seeds. This setting is very similar to the optimal parameter setting found for testing for independence between variables under additive noise in Section 5.1. Most of the measures obtain similar optimal parameters to the ones obtained when testing for independence between variables. Just I_{KDE} seems to require even larger kernel width when comparing sets of variables.

Figures 16 show average rank of each measure across different relationships. Individual results are shown in appendix B. RIC with $D_{\max} = \lfloor \sqrt{n/6} \rfloor$ and $K_r = 200$ looks more competitive than all other measures but I_{KDE} . Therefore, the strongest competitor seems to be I_{KDE} that with a careful choice of kernels achieves very good performance on simple relationships such as the linear, quadratic, and cubic. We also can see that the discretization based estimators of mutual information do not do a good job because they dramatically fail on some data set. Moreover, $I_{k\text{-means}}$ which produces the same number of bins as RIC has clearly lower performance than the latter. The superior performance of RIC is thus due to the randomization.

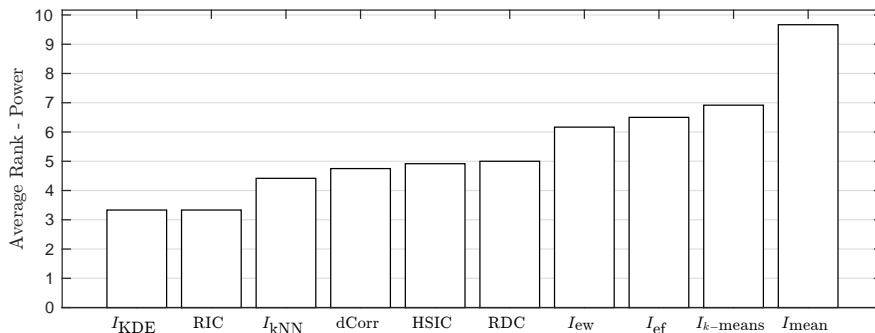


Fig. 16 Average rank across relationships for the multi-variable additive noise model. RIC with $D_{\max} = \lfloor \sqrt{n/6} \rfloor$ and $K_r = 200$ shares the top position with I_{KDE} .

6.2 Feature Selection for Regression

We also tested multi-variable measures of dependency in the task of feature selection using a similar framework to section 5.4. Rather than filtering the features according to their individual importance to the target variable Y , we proceed by forward selection. The optimal set of p features according to a dependency measure is identified by finding the best set of features $\mathbf{X} = \mathbf{X}^{p-1} \cup X_i$, with \mathbf{X}^{p-1} representing the set chosen at the previous iteration of forward selection and X_i chosen among the possible $m - (p - 1)$ features of a dataset. A multi-variable dependency measure can be fully employed in this case because we require to compute the dependency between \mathbf{X} features and the target variable Y at each step of the iteration.

As in section 5.3 and 5.4 we use RIC with parameters $D_{\max} = \lfloor \sqrt{n} \rfloor$ to avoid low density grids that are better suited for testing of independence tasks under additive noise. We use the random seeds discretization approach of Algorithm 4 with a fixed number of random seeds. We also choose to fix $D_X = 2$ and $D_Y = 5$ for the naive discretization based estimators of mutual information. Average results for all the measures are shown in Figure 17 and a table with detailed comparisons is presented in appendix B. We notice that the ranking by performance of classifier changes from the one obtained using the feature filtering approach, although RIC again shows competitive performance against the other approaches. All estimators of mutual information lose positions except for the I_{KDE} kernel based estimator. It seems that on multiple variables kernels are more effective than in the univariate scenario. Indeed, HSIC also gains a few positions. RDC's average performance stays the same and it still gets outperformed by dCorr. dCorr performs really well when computed on sets of variables. As previously noted, even in this case RIC outperforms $I_{k\text{-means}}$ and this result is due to the randomized approach.

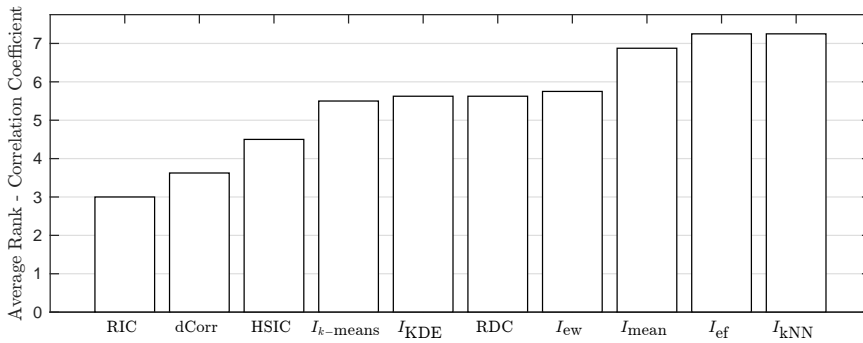


Fig. 17 Average rank when the target is maximization of the correlation coefficient between the predicted and the target value for a k NN regressor. The k NN regression is built on top of m^* features chosen by forward selection. Results are averaged across $m^* \leq 10$ and all datasets.

7 Conclusion

We presented the Randomized Mutual Information (RIC), an information theoretic measure of dependency between two sets random variables \mathbf{X} and \mathbf{Y} , that makes use of an ensemble of random grids. Our theoretical analysis justifies the benefits of having a low-variance estimator of mutual information based on grids for the task of ranking relationships, where systematic biases cancel each other out. By reducing the estimation variance of mutual information with grids, RIC is extremely competitive for ranking different relationships. We experimentally demonstrated its strong performance on univariate X and Y on the task of discrimination of noisy relationships, network inference and feature filtering for regression. We have shown that RIC can be extended to multivariate \mathbf{X} and \mathbf{Y} with a subtle discretization scheme. We recommend RIC's use with the default parameters: maximum number of random cut-offs $D_{\max} = \lfloor \sqrt{n} \rfloor$ and number of random discretizations $K_r = 20$ for both \mathbf{X} and \mathbf{Y} in general applications. However, D_{\max} can be decreased when testing for independence under additive noise and K_r can be increased to decrease the variance, at the cost of computational time.

Acknowledgements Supported by AWS in Education Grant Award.

References

- Breiman, L. Random forests. In *Machine Learning*, volume 45, pp. 5–32, 2001.
- Breiman, L. and Friedman, J. H. Estimating optimal transformations for multiple regression and correlation. *Journal of the American statistical Association*, 80 (391):580–598, 1985.
- Cellucci, C., Albano, A. M., and Rapp, P. Statistical validation of mutual information calculations: Comparison of alternative numerical algorithms. *Physical Review E*, 71(6):066208, 2005.
- Cover, T. M. and Thomas, J. A. *Elements of information theory*. John Wiley & Sons, 2012.
- Demšar, J. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.
- Dougherty, J., Kohavi, R., Sahami, M., et al. Supervised and unsupervised discretization of continuous features. In *Machine learning: proceedings of the twelfth international conference*, volume 12, pp. 194–202, 1995.
- Faivishevsky, L. and Goldberger, J. Ica based on a smooth estimation of the differential entropy. In *Advances in neural information processing systems*, pp. 433–440, 2009.
- Fayyad, U. and Irani, K. Multi-interval discretization of continuous-valued attributes for classification learning. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 1993.
- Fraser, A. M. and Swinney, H. L. Independent coordinates for strange attractors from mutual information. *Physical review A*, 33(2):1134, 1986.
- Garcia, S., Luengo, J., Sáez, J. A., López, V., and Herrera, F. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *Knowledge and Data Engineering, IEEE Transactions on*, 25(4):734–750, 2013.

- Geurts, P. *Bias/Variance Tradeoff and Time Series Classification*. PhD thesis, Department d'Électronique, Électronique et Informatique. Institut Montefiore. Université de Liège., 2002.
- Geurts, P., Ernst, D., and Wehenkel, L. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006.
- Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. Measuring statistical dependence with hilbert-schmidt norms. In *Algorithmic learning theory*, pp. 63–77. Springer, 2005.
- Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.
- Guo, X., Zhang, Y., Hu, W., Tan, H., and Wang, X. Inferring nonlinear gene regulatory networks from gene expression data based on distance correlation. *PloS one*, 9(2):e87446, 2014.
- Guyon, I. and Elisseeff, A. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003.
- Huo, X. and Szekely, G. J. Fast computing for distance covariance. *arXiv preprint arXiv:1410.1503*, 2014.
- Khan, S., Bandyopadhyay, S., Ganguly, A. R., Saigal, S., Erickson III, D. J., Protopopescu, V., and Ostrouchov, G. Relative performance of mutual information estimation methods for quantifying the dependence among short and noisy data. *Physical Review E*, 76(2):026209, 2007.
- Kraskov, A., Stögbauer, H., and Grassberger, P. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.
- Kursa, M. B. Random ferns method implementation for the general-purpose machine learning. *CoRR*, abs/1202.1121, 2012.
- Kvalseth, T. O. Entropy and correlation: Some comments. *Systems, Man and Cybernetics, IEEE transactions on*, 17(3):517–519, 1987.
- Lippert, C., Stegle, O., Ghahramani, Z., and Borgwardt, K. M. A kernel method for unsupervised structured network inference. In *International Conference on Artificial Intelligence and Statistics*, pp. 368–375, 2009.
- Lizier, J. T. Jidt: an information-theoretic toolkit for studying the dynamics of complex systems. *arXiv preprint arXiv:1408.3270*, 2014.
- Lopez-Paz, D., Hennig, P., and Schölkopf, B. The randomized dependence coefficient. In *Advances in Neural Information Processing Systems*, pp. 1–9, 2013.
- Luedtke, A. and Tran, L. The generalized mean information coefficient. *arXiv preprint arXiv:1308.5712*, 2013.
- Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D., and Califano, A. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*, 7 (Suppl 1):S7, 2006.
- Moddemeijer, R. On estimation of entropy and mutual information of continuous distributions. *Signal processing*, 16(3):233–248, 1989.
- Moon, Y.-I., Rajagopalan, B., and Lall, U. Estimation of mutual information using kernel density estimators. *Physical Review E*, 52(3):2318, 1995.
- Nguyen, H. V. and Vreeken, J. Universal dependency analysis. *arXiv preprint arXiv:1510.08389*, 2015.
- Nguyen, H. V., Müller, E., Vreeken, J., Efros, P., and Böhm, K. Multivariate maximal correlation analysis. In *Proceedings of the 31st International Conference*

- on *Machine Learning (ICML-14)*, pp. 775–783, 2014a.
- Nguyen, X. V., Chan, J., Romano, S., and Bailey, J. Effective global approaches for mutual information based feature selection. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 512–521. ACM, 2014b.
- Özuysal, M., Fua, P., and Lepetit, V. Fast keypoint recognition in ten lines of code. In *CVPR*, 2007.
- Reshef, D. N., Reshef, Y. A., Finucane, H. K., Grossman, S. R., McVean, G., Turnbaugh, P. J., Lander, E. S., Mitzenmacher, M., and Sabeti, P. C. Detecting novel associations in large data sets. *Science*, 334(6062):1518–1524, 2011.
- Reshef, D. N., Reshef, Y. A., Sabeti, P. C., and Mitzenmacher, M. M. An empirical study of leading measures of dependence. *arXiv preprint arXiv:1505.02214*, 2015a.
- Reshef, Y. A., Reshef, D. N., Finucane, H. K., Sabeti, P. C., and Mitzenmacher, M. M. Measuring dependence powerfully and equitably. *arXiv preprint arXiv:1505.02213*, 2015b.
- Romano, S., Bailey, J., Nguyen, V., and Verspoor, K. Standardized mutual information for clustering comparisons: One step further in adjustment for chance. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1143–1151, 2014.
- Romano, S., Vinh, N. X., Bailey, J., and Verspoor, K. A framework to adjust dependency measure estimates for chance. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pp. 423–431. Society for Industrial and Applied Mathematics, 2016.
- Ross, S. *A first course in probability*. Pearson, 2012.
- Schaffernicht, E., Kaltenhaeuser, R., Verma, S. S., and Gross, H.-M. On estimating mutual information for feature selection. In *Artificial Neural Networks ICANN 2010*, pp. 362–367. Springer, 2010.
- Simon, N. and Tibshirani, R. Comment on detecting novel associations in large data sets. *arXiv preprint arXiv:1401.7645*, 2011.
- Song, L., Smola, A., Gretton, A., Borgwardt, K. M., and Bedo, J. Supervised feature selection via dependence estimation. In *Proceedings of the 24th international conference on Machine learning*, pp. 823–830. ACM, 2007.
- Steuer, R., Kurths, J., Daub, C. O., Weise, J., and Selbig, J. The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, 18(suppl 2):S231–S240, 2002.
- Sugiyama, M. and Borgwardt, K. M. Measuring statistical dependence via the mutual information dimension. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pp. 1692–1698. AAAI Press, 2013.
- Székely, G. J., Rizzo, M. L., et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.
- Tang, D., Wang, M., Zheng, W., and Wang, H. Rapidmic: Rapid computation of the maximal information coefficient. *Evolutionary bioinformatics online*, 10:11, 2014.
- Van den Bulcke, T., Van Leemput, K., Naudts, B., van Remortel, P., Ma, H., Verschoren, A., De Moor, B., and Marchal, K. Syntren: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC bioinformatics*, 7(1):43, 2006.

- Villaverde, A. F., Ross, J., and Banga, J. R. Reverse engineering cellular networks with information theoretic methods. *Cells*, 2(2):306–329, 2013.
- Villaverde, A. F., Ross, J., Morán, F., and Banga, J. R. Mider: network inference with mutual information distance and entropy reduction. *PloS one*, 9(5):e96732, 2014.
- Vinh, N. X., Epps, J., and Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11:2837–2854, 2010.
- Wang, J., Kumar, S., and Chang, S.-F. Semi-supervised hashing for large-scale search. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(12):2393–2406, 2012.
- Wang, Y., Romano, S., Nguyen, V., Bailey, J., Ma, X., and Xia, S.-T. Unbiased multivariate correlation analysis. 2017.
- Xuan, N., Chetty, M., Coppel, R., and Wangikar, P. Gene regulatory network modeling via global optimization of high-order dynamic bayesian network. *BMC Bioinformatics*, 13(1):131, 2012.
- Zhang, Y., Jia, S., Huang, H., Qiu, J., and Zhou, C. A novel algorithm for the precise calculation of the maximal information coefficient. *Scientific reports*, 4, 2014.

Supplementary Material

A Identification of Noisy Relationships

Power at level $\alpha = 0.05$ of discrimination between complete noise and noisy relationship for each relationship type presented in the paper is shown in Fig 20 and 21. These two figures show result for each measure with optimal parameters for independence testing. Parameter tuning is performed on Figure 18 and Figure 19. Regarding the discretization based estimators of mutual information I_{ew} and I_{ef} , we varied the parameter c in $D = \lfloor \sqrt{n/c} \rfloor$ to change the grid coarsity. Similarly, we varied the parameter c in $D_{\max} = \lfloor \sqrt{n/c} \rfloor$ for RIC. The parameter s in RDC should be set around 1/6 when comparing variables (Lopez-Paz et al., 2013). In order to identify the best parameter settings for power in RDC, we explored values around 1/6 by varying \tilde{p} in $s = \frac{1}{6} \cdot \tilde{p}$. Similarly, the kernel widths in HSIC are usually set to the median distance of the data points according X and according Y . This is why we explored the values $\sigma_X = \text{med. dist. in } X \cdot \tilde{p}$ and $\sigma_Y = \text{med. dist. in } Y \cdot \tilde{p}$. The parameter \tilde{p} can be seen as a percentage of the default parameters.

Fig. 18 Parameter tuning to maximize the power of each measure on average for the **additive** noise model when comparing **variables**. These plots show the average area under power curve and their average across relationship types.

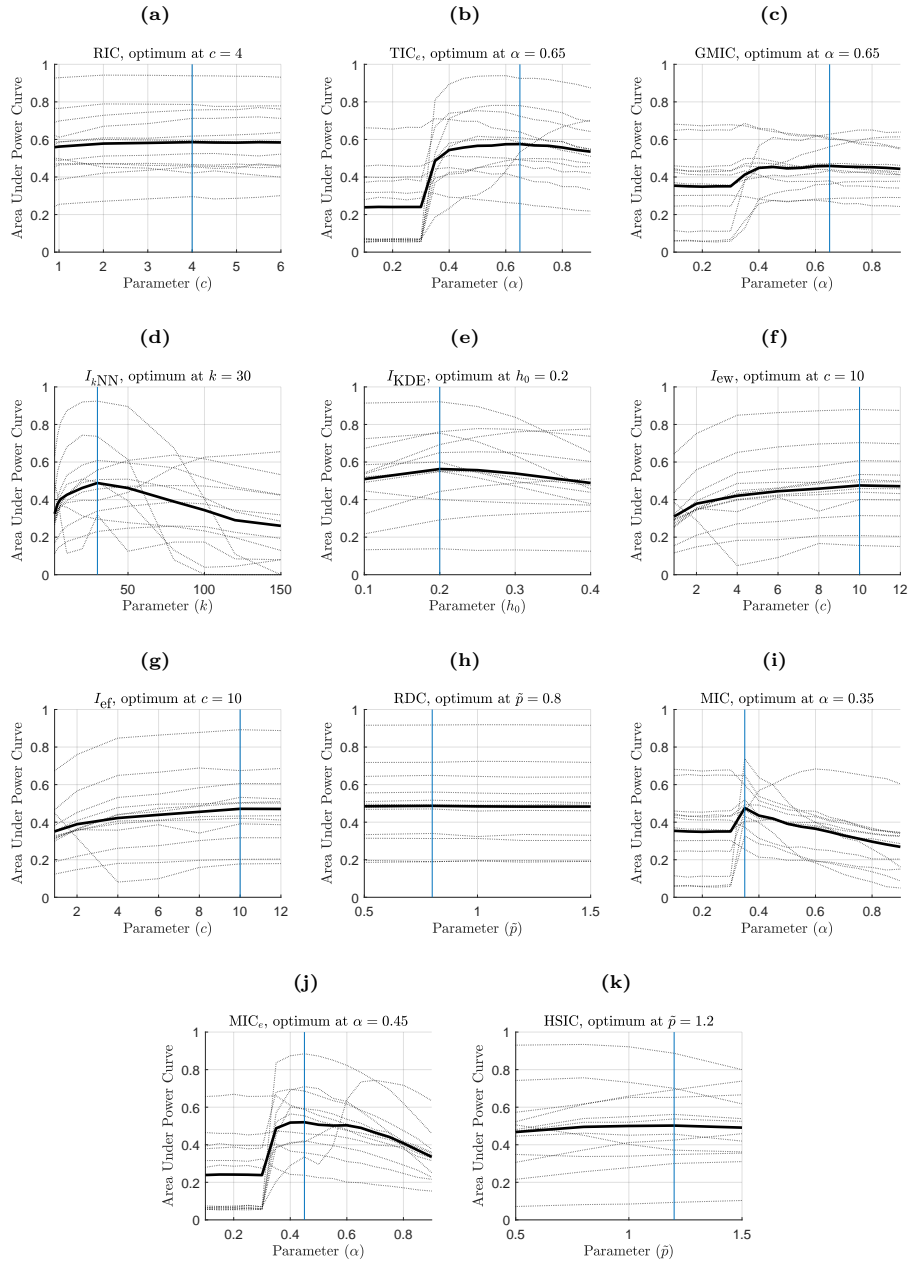
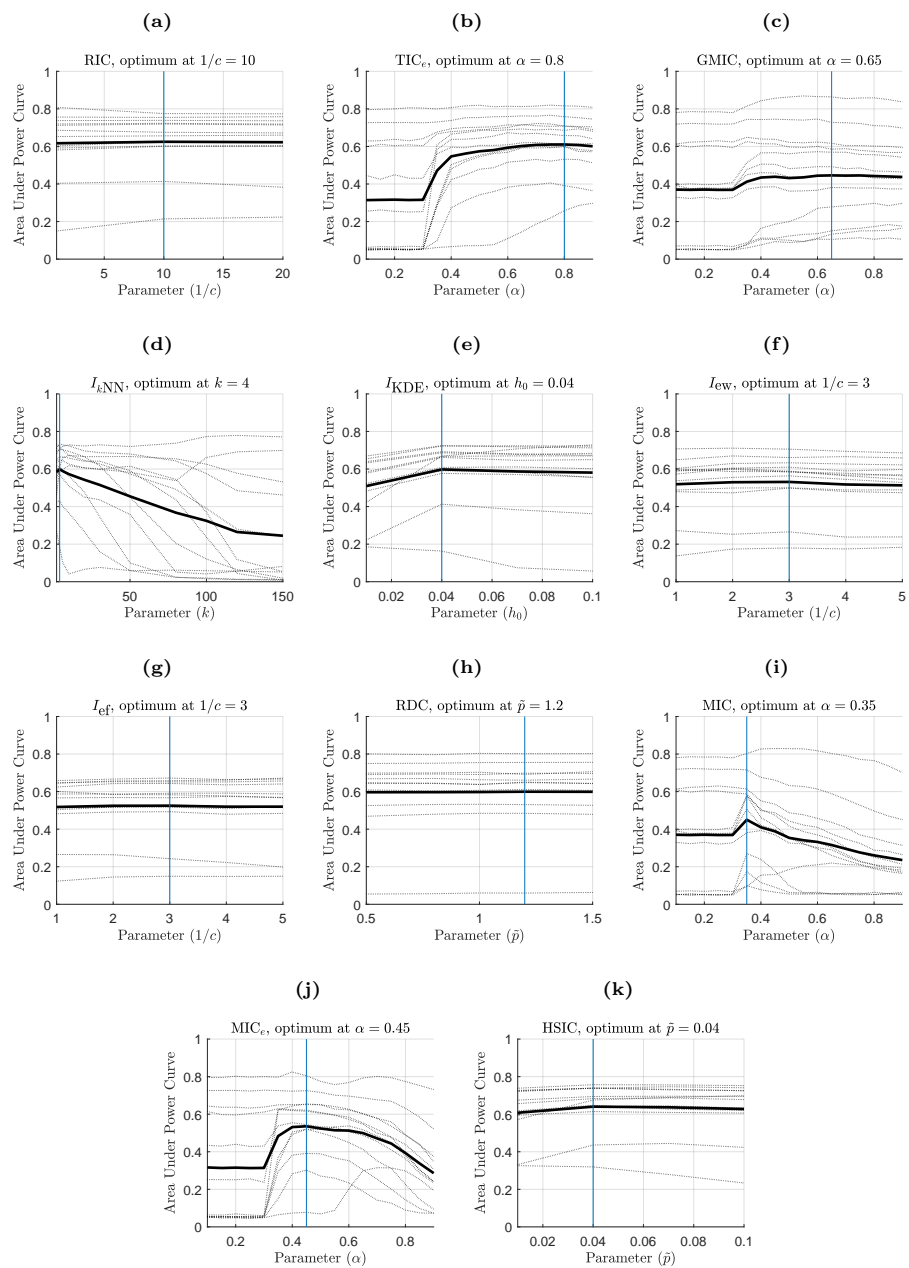


Fig. 19 Parameter tuning to maximize the power of each measure on average for the **white** noise model when comparing **variables**. These plots show the average area under power curve and their average across relationship types.



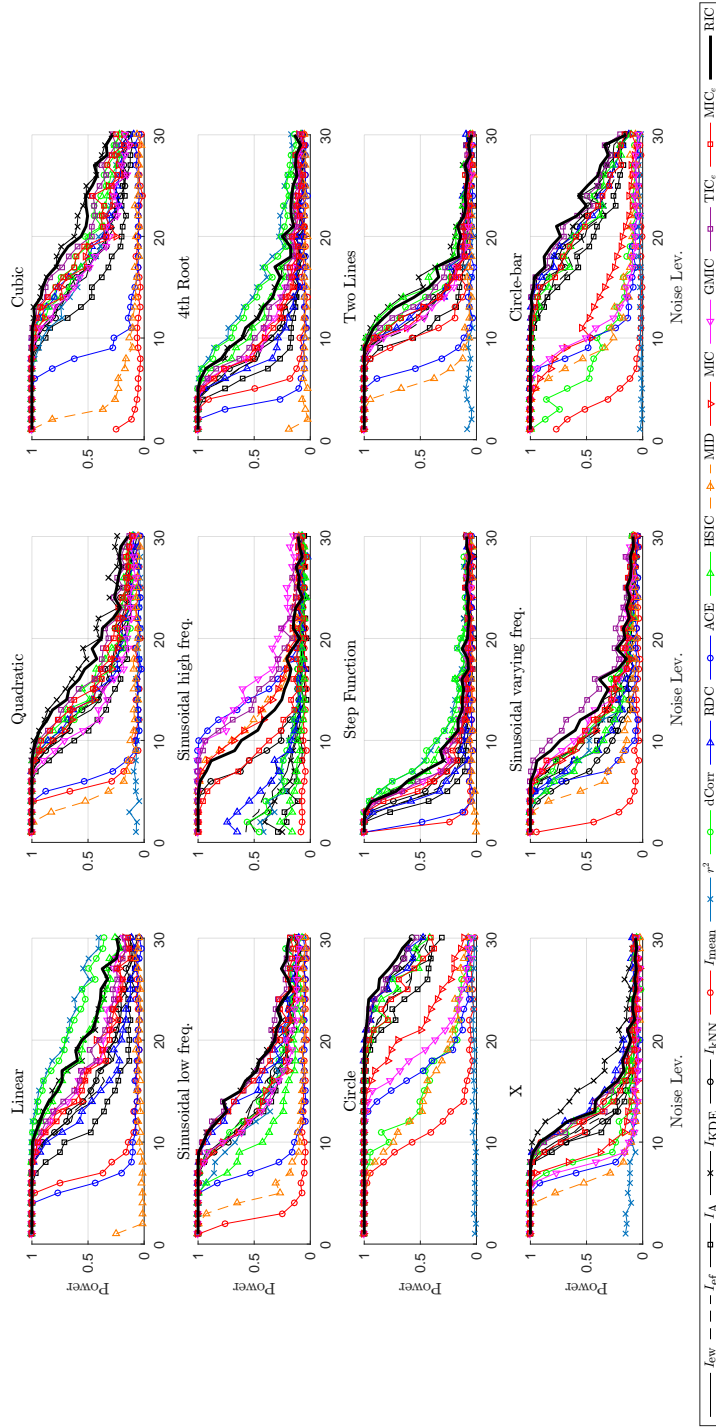


Fig. 20 Power in the additive noise scenario with optimal parameters (best viewed in color).

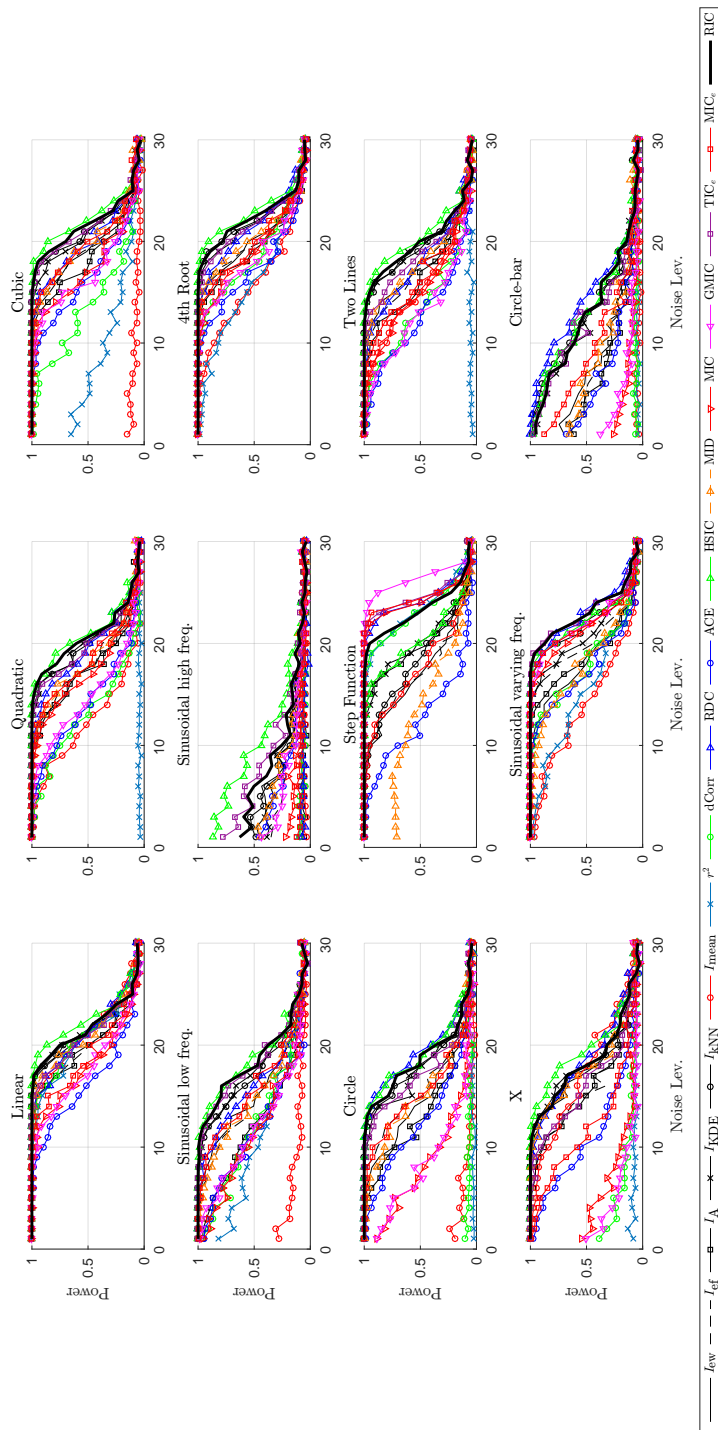


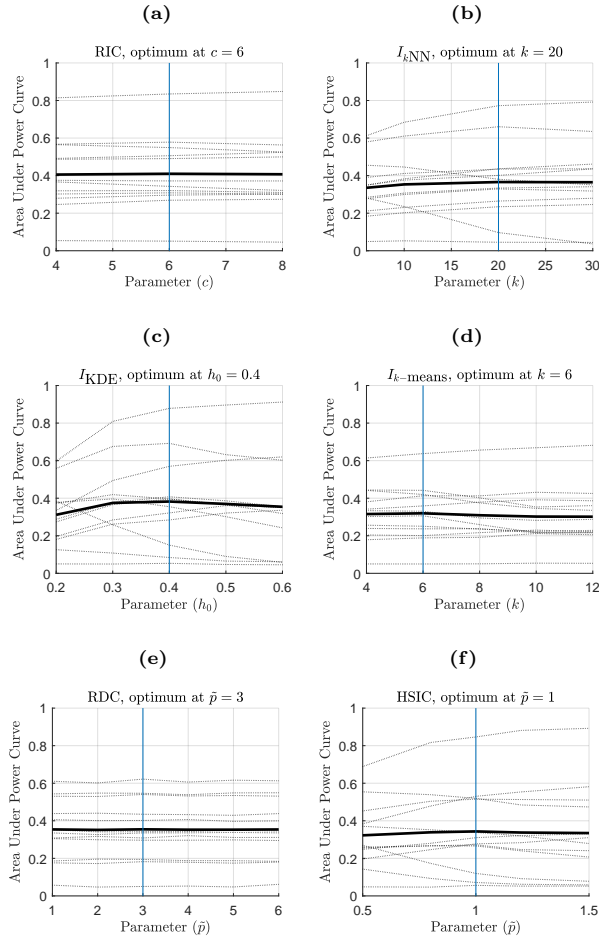
Fig. 21 Power in the white noise scenario with optimal parameters (best viewed in color).

B Multi-variable relationships

Power at level $\alpha = 0.05$ of discrimination between complete noise and noisy relationship for each relationship type presented in the paper is shown in Figure 23. The relationship is between a uniformly generated variable \mathbf{X} with $p = 3$ components and a single Y ($q = 1$). Parameter tuning for this task can be found in Figure 22. All the measures but I_{KDE} find similar optimal parameters to Section A when comparing variables. I_{KDE} requires larger kernel width when comparing sets of variables. Furthermore, RDC seem to be little sensitive to the parameter s . In that case we optimized s_X and s_Y independently with \tilde{p} where: $s_X = \frac{1}{6p} \cdot \tilde{p} = \frac{1}{6 \cdot 3} \cdot \tilde{p}$ and $s_Y = \frac{1}{6q} \cdot \tilde{p} = \frac{1}{6 \cdot 1} \cdot \tilde{p}$.

Table 9 shows results related to feature selection via forward selection.

Fig. 22 Parameter tuning to maximize the power of each measure on average for the **additive** noise model when comparing **sets of variables**. These plots show the average area under power curve and their average across relationship types.



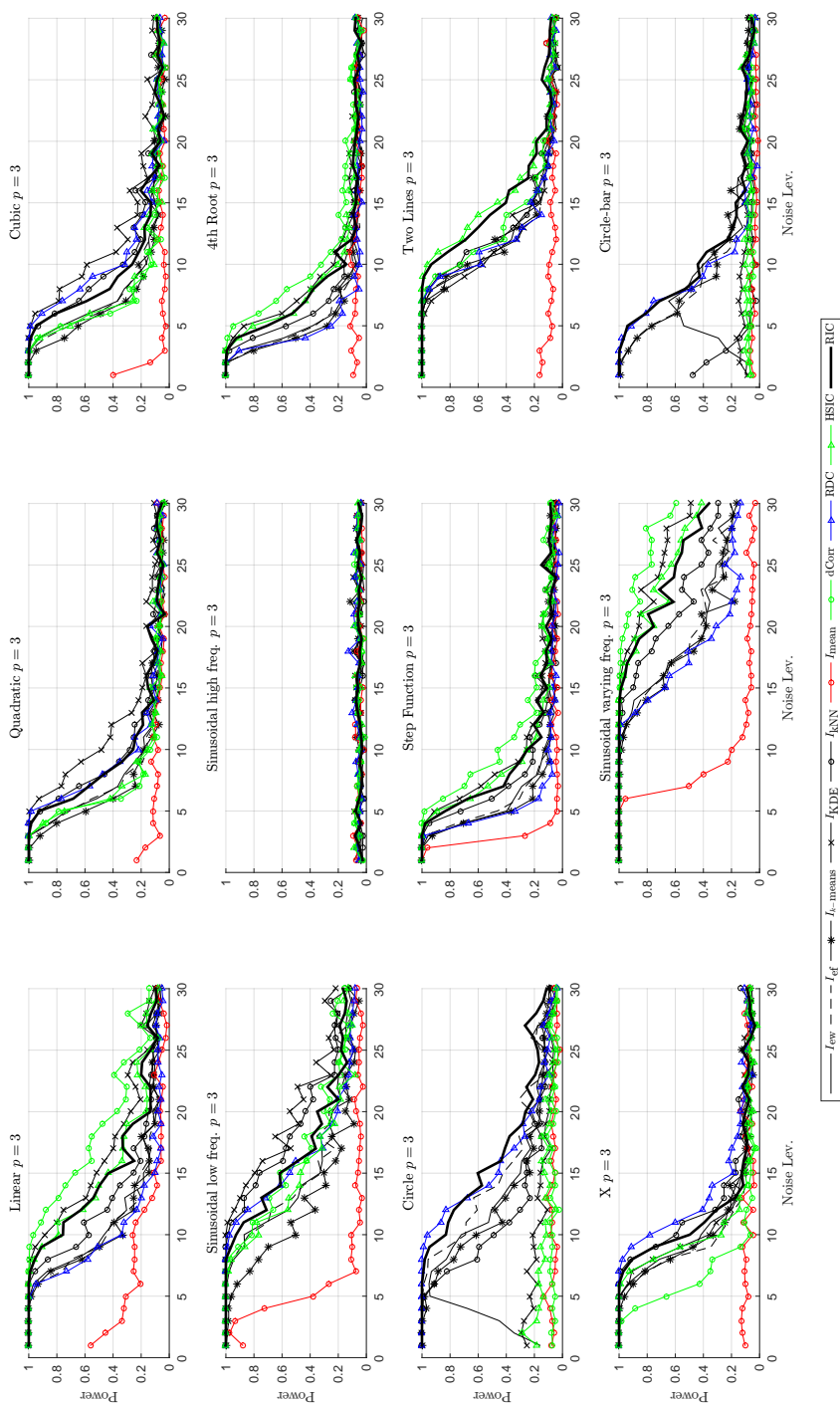


Fig. 23 Power in the additive noise scenario when comparing sets of variables with optimal parameters (best viewed in color).

Table 9 Correlation coefficient between the predicted and actual target value on 8 datasets using k NN ($k = 3$). The values are the mean correlation coefficient across the k NN regressors built on top of m^* features selected with forward selection with $m^* \leq 10$. n number of records; m number of features. Each cell shows mean correlation coefficient \pm std and either (+),(=), or (-) means statistically greater, equal, or smaller according to the 1-sided paired t -test ($\alpha = 0.05$) than RIC results.

(n, m)	Pyrim (74,27)	Bodyfat (252,14)	Triazines (186,60)	Wisconsin (194,32)	Crime (111,144)	Pole (1000,48)	Qsar (384,482)	Qsar2 (384,186)
RIC	0.291 \pm 0.122	0.668 \pm 0.106	0.125 \pm 0.045	0.050 \pm 0.014	0.931 \pm 0.056	0.778 \pm 0.235	0.347 \pm 0.126	0.401 \pm 0.026
I_{few}	0.288 \pm 0.100 (=)	0.523 \pm 0.158 (-)	0.104 \pm 0.064 (-)	0.073 \pm 0.026 (+)	0.704 \pm 0.120 (-)	0.743 \pm 0.204 (-)	0.330 \pm 0.068 (=)	0.394 \pm 0.014 (=)
I_{ef}	0.248 \pm 0.089 (-)	0.649 \pm 0.113 (-)	0.063 \pm 0.034 (-)	0.032 \pm 0.004 (-)	0.675 \pm 0.120 (-)	0.698 \pm 0.253 (-)	0.358 \pm 0.095 (=)	0.345 \pm 0.050 (-)
I_k -means	0.373 \pm 0.100 (+)	0.663 \pm 0.108 (=)	0.093 \pm 0.047 (-)	0.038 \pm 0.008 (-)	0.781 \pm 0.016 (-)	0.763 \pm 0.198 (=)	0.224 \pm 0.083 (-)	0.329 \pm 0.030 (-)
I_{KDE}	0.270 \pm 0.084 (=)	0.628 \pm 0.121 (-)	0.073 \pm 0.044 (-)	0.042 \pm 0.013 (=)	0.785 \pm 0.140 (-)	0.764 \pm 0.260 (=)	0.321 \pm 0.071 (=)	0.411 \pm 0.146 (=)
I_{kNN}	0.261 \pm 0.073 (=)	0.637 \pm 0.117 (-)	0.049 \pm 0.021 (-)	0.027 \pm 0.012 (-)	0.510 \pm 0.288 (-)	0.804 \pm 0.217 (+)	0.278 \pm 0.061 (-)	0.391 \pm 0.030 (=)
I_{mean}	0.293 \pm 0.122 (=)	0.626 \pm 0.019 (-)	0.107 \pm 0.035 (-)	0.032 \pm 0.015 (-)	0.905 \pm 0.140 (-)	0.425 \pm 0.102 (-)	0.182 \pm 0.102 (-)	0.232 \pm 0.053 (-)
dCorr	0.275 \pm 0.063 (=)	0.672 \pm 0.104 (=)	0.104 \pm 0.054 (-)	0.073 \pm 0.016 (+)	0.920 \pm 0.053 (-)	0.772 \pm 0.279 (=)	0.352 \pm 0.054 (=)	0.362 \pm 0.129 (-)
RDC	0.333 \pm 0.146 (+)	0.550 \pm 0.172 (-)	0.173 \pm 0.046 (+)	0.026 \pm 0.013 (-)	0.761 \pm 0.115 (-)	0.715 \pm 0.203 (-)	0.296 \pm 0.071 (-)	0.415 \pm 0.046 (+)
HSIC	0.327 \pm 0.104 (=)	0.653 \pm 0.112 (-)	0.084 \pm 0.027 (-)	0.057 \pm 0.021 (=)	0.775 \pm 0.000 (-)	0.787 \pm 0.214 (=)	0.361 \pm 0.046 (=)	0.144 \pm 0.041 (-)