CASMI—An Entropic Feature Selection Method in Turing's Perspective

Jingyi Shi [1,*][iD], Jialin Zhang [1][iD] and Yaorong Ge [2]

[1] Department of Mathematics and Statistics, Mississippi State University, Mississippi State, Starkville, MS 39762, USA; jzhang@math.msstate.edu

[2] Department of Software and Information Systems, The University of North Carolina at Charlotte, Charlotte, NC 28223, USA; yge@uncc.edu

* Correspondence: jshi@math.msstate.edu

Abstract: Health data are generally complex in type and small in sample size. Such domain-specific challenges make it difficult to capture information reliably and contribute further to the issue of generalization. To assist the analytics of healthcare datasets, we develop a feature selection method based on the concept of coverage adjusted standardized mutual information (CASMI). The main advantages of the proposed method are: (1) it selects features more efficiently with the help of an improved entropy estimator, particularly when the sample size is small; and (2) it automatically learns the number of features to be selected based on the information from sample data. Additionally, the proposed method handles feature redundancy from the perspective of joint-distribution. The proposed method focuses on non-ordinal data, while it works with numerical data with an appropriate binning method. A simulation study comparing the proposed method to six widely cited feature selection methods shows that the proposed method performs better when measured by the Information Recovery Ratio, particularly when the sample size is small.

Keywords: feature selection; small sample; healthcare dataset; sample coverage; entropy estimation

1. Introduction

Inspired by the recent advancement in big data, health informaticians are attempting to assist health care providers and patients from a data perspective, with the hope of improving quality of care, detecting diseases earlier, enhancing decision making, and reducing healthcare costs [1]. In the process, health informaticians have been confronted with the issue of generalization [2]. Analyzing real health data involves many practical problems that could contribute to the issue of generalization; for example, the unknown amount of information (signal) versus error (noise), the curse of dimensionality, and the generalizability of models. All these trivial problems boil down to the essential problem issued by a limited sample. With the limitation of the sample size, the information from the sample cannot represent the information of the population to a desirable extent. For this reason, a simple way to address these trivial problems is to collect a sufficiently large sample, which is unfortunately often impractical in healthcare because of multiple reasons. For example:

1. The term sufficiently large is relative to the dimensionality of data and the complexity of feature spaces. Health data are generally large in dimensionality, particularly when dummy variables (one-hot-encoding) are adopted to represent enormous categories of complex qualitative features (such as extracted words from clinical notes). As a result, a dataset with a sample size of 1,000,000 may not be sufficient, depending on its feature spaces.
2. There may not be sufficient patient cases for a rare disease. Even if there are ample potential cases, it may be cost-prohibitive for clinical trials to achieve a sufficient sample.

Without a sufficiently large sample, dimension reduction becomes a major research direction in health data analytics, as reducing dimensionality can partly relieve the issues from a limited sample. These dimension reduction techniques mainly focus on feature selection and feature projection, where feature selection can be further applied to the features created by feature projection. In this article, we focus on feature selection. Feature selection has become an important research area, dating back at least to 1997 [3,4]. Since then, many feature selection methods have been proposed and well discussed in multiple recent review papers, such as [5–7]. To apply these feature selection methods to health data, domain-specific challenges must be considered.

Health data can be numerical and categorical. For example, many machine readings (e.g., heart rate, blood pressure, and blood oxygen level) are numerical, while gene expression data are categorical. A healthcare dataset could contain numerical data only, categorical data only, or a combination of both data types. The fundamental distinction between numerical data and categorical data is whether the data space is ordinal or non-ordinal. As a result, data consisting of only numbers are not necessarily numerical data; for example, gene expression data can be coded to numbers using dummy variables, but it should be still considered as categorical. When the data space is ordinal (numerical data only), classical methods—which detect the association using ordinal information—are more powerful in capturing the associations in data. When the data space is non-ordinal (categorical data only), ordinal information does not naturally exist; hence, continuing to use classical methods onto coded data loses their original advantages and has additional estimation issues. Namely, involving dummy variables increases the dimensionality of data and further exacerbates the estimation problem using a limited sample. This particularly happens when an involved categorical feature has a complex feature space that requires a tremendous number of dummy variables to represent all the different categories.

To handle categorical data for feature selection, only information–theoretic quantities (e.g., entropy and mutual information [8]) serve the purpose. When a dataset is a combination of both data types, it is inconclusive about whether to use classical or information–theoretic methods. In general, if one believes that the numerical data in the dataset carry more information than the categorical data, then classical methods can be used. If one believes the categorical data carry more information, then information–theoretic methods should be used, and the numerical data should be binned to categorical data. One should be advised that coding categorical data for classical methods increases dimensionality and issues more difficulties in estimation, while binning numerical data for information–theoretic methods inevitably loses ordinal information. It should also be noted that, although ordinal information could provide extra information about associations among the data, the ordinal information could also mislead a person's judgement when associations actually exist, but there is no visual pattern among the data. The way that classical methods work is very similar to our visualization; if there is a pattern that can be visually observed, then it can also be detected by some classical methods. However, not all associations among numerical data are visually observable, in which case, classical methods would fail to detect the associations. On the other hand, if there is a visual pattern among data, binning the data (losing the ordinal information) would not necessarily lead to a loss of associations among data; it depends on the binning methods and performance of the information–theoretic methods.

Classical feature selection methods include, but are not limited to, Fisher score [9], ReliefF [10], trace ratio [11], Laplacian score [12], Zhao and Liu's spectral feature selection (SPEC) [13], l_p-regularized [14], $l_{p,q}$-regularized [14], efficient and robust feature selection (REFS) [15], multi-cluster feature selection (MCFS) [16], unsupervised feature selection algorithm (UDFS) [17], nonnegative discriminative feature selection (NDFS) [18], t-score [19], and least absolute shrinkage and selection operator (LASSO) [20]. All these classical feature selection methods require information from ordinal spaces, such as moments (e.g., mean and variance) and spacial information (e.g., nearest location and norms). Information–theoretic feature selection methods include, but are not limit to, mutual information maximisation (MIM) [21], mutual information feature selection (MIFS) [22], joint mutual information (JMI) [23], minimal conditional mutual information maximisation (CMIM) [24,25],

minimum redundancy maximal relevancy (MRMR) [26], conditional infomax feature extraction (CIFE) [27], informative fragments (IF) [24], double input symmetrical relevance (DISR) [28], minimal normalised joint mutual information maximisation (NJMIM) [29], chi-square score [30], gini index [31], and CFS [32]. All these information–theoretic methods use ordered probabilities, which always exist in non-ordinal spaces. For example, frequencies, category probabilities (proportions), Shannon's entropy, mutual information, and symmetric uncertainty are all functions of ordered probabilities.

In many cases, all (or most) of the data in a healthcare dataset could be categorical. To analyze the categorical data in such a dataset, information–theoretic feature selection methods are preferred because they could capture the associations among features without using dummy variables, where classical methods require dummy variables that would increase the dimensionality. Most existing information–theoretic methods use entropy or mutual information (a function of entropy) to measure associations among data. Information–theoretic methods that do not use entropy include Gini index and chi-square score. Gini index focuses on whether a feature is separative, but does not indicate probabilistic associations. Chi-square score relies on the performance of asymptotic normality on each component, and when there are categories with low frequencies (e.g., less than five), the chi-square score is very unstable. However, under a limited sample, we should expect at least a few, if not many, categories would have relatively low frequencies. For the existing information–theoretic methods that use entropy (we call these entropic methods), all of them estimate entropy with the classical maximum likelihood estimator (the plug-in estimator). The plug-in entropy estimator performs very poorly when the sample size is not sufficiently large [33,34], and we have discussed that the sample size is usually relatively limited in healthcare datasets. As a result, to use entropic methods in healthcare data analytics, the estimation of entropy under small samples must be improved.

In addition to estimation based on small samples, the unhelpful association is another issue with these samples. While the issue of estimation can be addressed by using a better estimator, the problem of unhelpful association is trickier. The unhelpful association is partially a result of sample randomness, and it could be severe when the sample size is small. Suppose there is a healthcare dataset with multiple features and one outcome, and there is a feature in the dataset that could distinguish the values of the outcome based on the sample information, then there are three possible situations for the feature:

Situation One	The feature has abundant real information toward the outcome, and the real information is well preserved by the sample data.
Situation Two	The feature has abundant real information toward the outcome, but the real information is not well preserved by the sample data.
Situation Three	The feature has little real information but seems relevant to the outcome because of randomness in the sample.

The term real information of a feature means the feature-carried information that could indicate the values of the outcome at the population level. All three situations are conceptual classifications. At the population level, situation one and two features are relevant features, and situation three features are irrelevant features. It is clear that situation one features should be selected, while situation three features should be dropped. For situation two, caution should be exercised. Intuitively, situation two features should be kept as they are relevant features at the population level. However, as a result of a limited sample, the information carried by these situation two features are very subtle. There are at least two constitutional problems about the information from situation two features. First, although the feature could distinguish the values of the outcome based on the sample information, the sample-preserved information possibly provides only a meager coverage of all the possible values of the feature. As a result, when there is a new observation (e.g., a new patient), it is very likely that the new observation's corresponding label has not been observed by the preserved information, in which case no outcome information is available to assist prediction based on the information of such a situation two feature. Second, because of the limited sample, the predictability of the situation two features revealed by the sample may not be complete; hence, it could contribute as an error

(noise). For example, based on the sample information, different values of a situation two feature could possibly uniquely determine a corresponding value of the outcome (particularly when a feature space is complex while the sample size is small), but this deterministic relationship revealed by a limited sample is unlikely to be true at the population level. As a result, using this information in further modeling and prediction would be wrong and could further contribute to the issue of generalization. Therefore, we suggest omitting situation two features. In addition, one should note that a relevant feature being categorized as situation two is a consequence of a limited sample. All situation two features would eventually become situation one when the sample size grows (because more real information would be revealed). As a summary, under a limited sample, situation one features should be kept, and situation two and three features should be dropped.

Focusing on the domain-specific challenges from health data, we develop the proposed entropic feature selection method based on the concept of coverage adjusted standardized mutual information (CASMI). The proposed method aims at improving the performance of estimation and addressing the issue of unhelpful association under relatively small samples. The rest of the article is organized as follows. The concept, intuition, and estimation of CASMI are discussed in Section 2. The proposed method is described in detail in Section 3 and evaluated by a simulation study in Section 4. A brief discussion is in Section 5.

2. CASMI and Its Estimation

In this section, we introduce the concept, intuition, and estimation of CASMI. Before we proceed, let us state the notations first.

Let $\mathscr{X} = \{x_i; i = 1, \cdots, K_1\}$ and $\mathscr{Y} = \{y_j; j = 1, \cdots, K_2\}$ be two finite alphabets with cardinalities $K_1 < \infty$ and $K_2 < \infty$, respectively. Consider the Cartesian product $\mathscr{X} \times \mathscr{Y}$ with a joint probability distribution $\mathbf{p} = \{p_{i,j}\}$. Let the two marginal distributions be respectively denoted by $\mathbf{p}_x = \{p_{i,\cdot}\}$ and $\mathbf{p}_y = \{p_{\cdot,j}\}$, where $p_{i,\cdot} = \sum_j p_{i,j}$ and $p_{\cdot,j} = \sum_i p_{i,j}$. Assume that $p_{i,\cdot} > 0$ and $p_{\cdot,j} > 0$ for all $1 \le i \le K_1$ and $1 \le j \le K_2$ and that there are $K = \sum_{i,j} 1[p_{i,j} > 0]$ non-zero entries in $\{p_{i,j}\}$. We re-enumerate these K positive probabilities in one sequence and denote it as $\{p_k; k = 1, \cdots, K\}$. Let X and Y be random variables following distributions $\mathbf{p}_x$ and $\mathbf{p}_y$, respectively. For every pair of i and j, let $f_{i,j}$ be the observed frequency of the random pair (X, Y) taking value (x_i, y_j), where $i = 1, \cdots, K_1$ and $j = 1, \cdots, K_2$, in an independent and identically distributed (*iid*) sample of size n from $\mathscr{X} \times \mathscr{Y}$ under $\mathbf{p}$, and let $\hat{p}_{i,j} = f_{i,j}/n$ be the corresponding relative frequency. Consequently, we write $\hat{\mathbf{p}} = \{\hat{p}_k\}$ (*i.e.*, $\{\hat{p}_{i,j}\}$), $\hat{\mathbf{p}}_x = \{\hat{p}_{i,\cdot}\}$, and $\hat{\mathbf{p}}_y = \{\hat{p}_{\cdot,j}\}$ as the sets of observed joint and marginal relative frequencies. Shannon's mutual information (MI) between X and Y is defined as

$$MI(X, Y) = H(X) + H(Y) - H(X, Y), \tag{1}$$

where

$$H(X) = -\sum_i p_{i,\cdot} \ln p_{i,\cdot},$$

$$H(Y) = -\sum_j p_{\cdot,j} \ln p_{\cdot,j},$$

$$H(X, Y) = -\sum_i \sum_j p_{i,j} \ln p_{i,j} = -\sum_{k=1}^K p_k \ln p_k.$$

We define the CASMI as follows:

Definition 1 (CASMI). *κ^*, the coverage adjusted standardized mutual information (CASMI) of a feature X to an outcome Y, is defined as*

$$\kappa^*(X, Y) = \kappa(X, Y) \cdot (1 - \pi_0(X)), \tag{2}$$

where

$$\kappa(X, Y) = \frac{MI(X, Y)}{H(Y)}, \tag{3}$$

and $(1 - \pi_0)$ *is the sample coverage that was first introduced by Good [35] as "the proportion of the population represented by (the species occurring in) the sample".*

2.1. Intuition of CASMI

Many entropic concepts can measure the associations among non-ordinal data; for example, MI, Kullback–Leibler divergence [36], conditional mutual information [37], and weighted variants [38]. Among them, MI is the fundamental concept as all the other entropic association measurements are developed based on or equivalent to MI. For this reason, we develop the CASMI starting with MI. It is well known that $MI \geq 0$, and $MI(X, Y) = 0$ if and only if X and Y are independent. However, MI is not bounded from above; hence, using the values of MI to compare the degrees of dependence among different pairs of random variables is inconvenient. Therefore, it is necessary to standardize the mutual information, which yields to the so-called standardized mutual information (SMI) or normalized variants. Reference [39] provides several forms of SMI, such as $MI/H(X)$ (also known as information gain ratio if X is a feature and Y is the outcome), $MI/H(Y)$, and $MI/H(X, Y)$. All these forms of SMI can be proven to be bounded by $[0, 1]$, where 0 stands for independence between X and Y, and 1 stands differently for different SMIs. For $MI/H(X)$ (information gain ratio), 1 means that, given the value of Y (outcome), the value of X (feature) is determinate. For $MI/H(Y)$, 1 means that, given the value of X, the value of Y is determinate. For $MI/H(X, Y)$, 1 means a one-to-one correspondence between X and Y.

The goal of feature selection is to separate the predictive features from non-predictive features. In this regard, $MI/H(Y) = 1$ is the most desirable because $MI/H(X) = 1$ does not indicate the predictability of X, and $MI/H(X, Y) = 1$ is too strong and unnecessary. Therefore, we select κ in (3) as the SMI in CASMI.

As we have discussed, detecting unhelpful associations under small samples is important in health data analytics as involving unhelpful associations would bring too much noise or unnecessary dimensions to model-building or prediction. In other words, we would like to detect situation two and three features in a limited sample. The common characteristic among situation two and three features is the information revealed by the limited sample covers little of the total information in the population. For this reason, we use sample coverage $(1 - \pi_0)$, the concept introduced by Good, to detect these features. A feature with high predictability but low sample coverage must belong to either situation two or three. In CASMI, we multiply the SMI by the sample coverage. Under this setting, although features from situations 2 and 3 have high SMI values, their CASMI scores would be low because of their low sample coverages; hence, these features would not be selected in a greedy selection. On the other hand, the CASMI score for a situation one feature would be high because both SMI and the sample coverage are high. As a result, by selecting features greedily, situation one features would be selected, while situation two and three features would be dropped.

The purpose of CASMI is to capture the association between a feature and the outcome, with a penalized term from the sample coverage, so that features under situations 2 and 3 would be eliminated. By selecting features under only situation one, the issue of generalization under small samples is expected to be reduced.

It may be interesting to note that the CASMI is an information–theoretic quantity that is related to both the population and the sample. It is neither a parameter nor a statistic, and it is only observable when both the population and the sample are known. Next, we introduce its estimation.

2.2. Estimation

To estimate κ^* (CASMI), we need to estimate π_0 and κ. $\pi_0(X)$ can be estimated by Turing's formula [35]

$$T_1(X) = N_1(X)/n, \tag{4}$$

where $N_1(X)$ is the number of singletons in the sample. For example, if a sample of English letters consists of $\{a, a, a, b, c, c, d, e, e, f\}$, then the corresponding $N_1 = 3$ (b, d, and f are the three singletons). Discussions on the performance of estimating π_0 by T_1 can be found in [39,40]. In experimental categorical data, singletons could possibly indicate the sample size is small. As the sample size grows, the chance of obtaining a singleton in the sample approaches zero. It may be interesting to note that using (4) to estimate the sample coverage would automatically separate ID-like features. This is because an ID-like feature is naturally all (or almost all) singletons and would result in a zero (or very small) estimated sample coverage that further leads to a zero (or very low) CASMI score; hence, such an ID-like feature would not be selected.

Estimating $\kappa(X, Y)$ is equivalent to estimating $MI(X, Y)$ and $H(Y)$. As we have discussed, thus far, all the existing entropic information–theoretic methods use the plug-in estimator of entropy ($\hat{H}$). However, the plug-in entropy estimator has a huge bias, particularly when the sample size is small. Reference [33] showed that the bias of $\hat{H}$ is

$$\mathrm{E}(\hat{H}) - H = -\frac{K-1}{2n} + \frac{1}{12n^2}\left(1 - \sum_{k=1}^{K}\frac{1}{p_k}\right) + \mathcal{O}\left(n^{-3}\right),$$

(We write $f = \mathcal{O}(g(n))$ to denote $\limsup_{n\to\infty}|f(n)/g(n)| < \infty$.) where n is the sample size and K is the cardinality of the space on which the probability distribution $\{p_k\}$ lives. Based on the expressions of the bias, it is easy to see that the plug-in estimator underestimates the real entropy, and the bias approaches 0 as n (sample size) approaches infinity, with a rate of n^{-1} (power decay). Because of the power decaying rate, the bias is not small when sample size (n) is relatively low.

To improve the estimation under a small sample, we adopt the following $\hat{H}_z$ [41] as the estimator of H:

$$\hat{H}_z = \sum_{v=1}^{n-1}\frac{1}{v}\frac{n^{1+v}[n-(1+v)]!}{n!}\sum_{k}\left[\hat{p}_k\prod_{j=0}^{v-1}\left(1 - \hat{p}_k - \frac{j}{n}\right)\right]. \tag{5}$$

Compared to the power decaying bias of $\hat{H}$, $\hat{H}_z$ has an exponentially decaying bias

$$\mathrm{E}(\hat{H}_z) - H = \mathcal{O}\left(\frac{(1-p_\wedge)^n}{n}\right),$$

where $p_\wedge = \min\{p_k > 0\}$.

To help understand the differences between the power decaying bias and exponentially decaying bias, we conduct a simulation. In the simulation, the real underlying distribution is $p_k = k/2001000$, where $k = 1, 2, \ldots, 2000$ (i.e., a triangle distribution). Under this setting, the true entropy $H = 7.408005$. To compare the two estimators, we independently generate 10,000 samples following the triangle distribution for each of the six sample size settings in Table 1 (i.e., we generate 60,000 random samples in total). The average values of $\hat{H}$ and $\hat{H}_z$ under different sample sizes are summarized in Table 1.

The calculation shows that $\hat{H}$ would consistently underestimate H more than $\hat{H}_z$. The underestimation is more severe when the sample size is smaller. Therefore, from a theoretical perspective, we expect that adopting $\hat{H}_z$ in estimating the entropies in CASMI would provide a better estimation, particularly under small samples. Furthermore, we expect that CASMI would capture the associations among features and the outcome more accurately under small samples because of the improvement in estimation. Interested readers can find additional discussions on comparison among more entropy estimators in [41], and comparison about mutual information estimators using $\hat{H}$ and $\hat{H}_z$ in [42].

Table 1. Estimation comparison between $\hat{H}$ and $\hat{H}_z$.

n	100	300	500	1000	1500	2000
avg. of $\hat{H}$	4.56	5.57	6.00	6.51	6.75	6.89
avg. of $\hat{H}_z$	5.11	6.09	6.49	6.92	7.11	7.21

Consequently, we let

$$\widehat{MI}_z(X, Y) = \hat{H}_z(X) + \hat{H}_z(Y) - \hat{H}_z(X, Y), \tag{6}$$

and we estimate κ as

$$\hat{\kappa}_z(X, Y) = \frac{\widehat{MI}_z(X, Y)}{\hat{H}_z(Y)}. \tag{7}$$

As a summary, we estimate κ^* by the following estimator, which is the scoring function of the selection stage in the proposed method.

$$\hat{\kappa}^*(X, Y) = \hat{\kappa}_z(X, Y) \cdot (1 - T_1(X)), \tag{8}$$

where $\hat{\kappa}_z$ is defined in (7) and T_1 is defined in (4). $\hat{\kappa}^*$ adopts an entropy estimator with an exponentially decaying bias to improve the performance in estimating κ^* and to capture the associations when the sample size is not sufficiently large. Furthermore, we expect that involving the sample coverage would separate and drop situation two and three features under small samples.

3. CASMI-Based Feature Selection Method

In this section, we introduce the proposed feature selection method in detail. The proposed method contains two stages. Before we present the two stages, let us first discuss data preprocessing.

3.1. Data Preprocessing

To use the proposed method, all features and the outcome data must be preprocessed to categorical data. Continuous numerical data must be discretized, and there are numerous discretization methods [43]. While binning continuous features, the estimated sample coverage (4) should be checked to avoid over-discretization, which increases the risk of wrongly shifting a feature from situation one to situation two.

If the data are already categorical, one may need to combine some of the categories to improve the sample coverage, when necessary. When most observations of a feature are singletons, then the coverage is close to 0, in which case it is difficult to draw any reliable and generalizable statistical inference. Therefore, for features that are believed to carry real information but have low sample coverages (below 50%), it is suggested to regroup them to create repeats and improve coverages. Note that not all features are worth regrouping; for example, if a feature is the IDs of patients, regrouping should be avoided as there is no reason to believe an ID can contribute to the outcome. The proposed method does not select features with low sample coverages; hence, ID-like features are eliminated automatically.

When a feature contains missing (or invalid) data that cannot be recovered by the data collector, without deleting the feature, there are several possible remedies, such as deleting the observation, making an educated guess, predicting the missing values, and listing all missing values as NA. While it is the user's preference on how to handle the missing data, one should be advised that manipulating (guessing or predicting) the missing data could create (or enhance) false associations; therefore, one should be cautious. Assigning all the missing values as NA generally would not create false associations, but it may reduce the predictive information of the feature. The performance of each remedy method could vary from situation to situation. Additional discussions on handling

missing data can be found in [44–46]. We suggest dealing with the missing data at the beginning of the data preprocessing.

The processed data should contain only categorical features and outcome(s). A feature with only integer values could be considered as categorical as long as the sample coverage is satisfactory.

3.2. Stage 1: Eliminate Independent Features

In this stage, we eliminate the features that are believed to be independent of the outcome based on a statistical test. This step filters out the features that are very unlikely to be useful; hence, the computation time for feature selection is reduced.

Suppose there are p features, $X_1, X_2, \ldots, X_p$, and one outcome, Y, in a dataset. Note that there could be multiple outcome attributes in a dataset. Because each outcome attribute has its own related features, when making a feature selection, we consider one outcome attribute at a time.

In finding independent features, we adopt a chi-squared test of independence using $\widehat{MI}_z$ as the statistic.

Theorem 1 ([47]). *Provided that $MI = 0$,*

$$\chi^2 = 2n\widehat{MI}_z + (K_1 - 1)(K_2 - 1) \xrightarrow{L} \chi^2\left((K_1 - 1)(K_2 - 1)\right), \tag{9}$$

where $\widehat{MI}_z$ is defined in (6). K_1 and K_2 are the effective cardinalities of the selected feature X and the outcome Y, respectively. We write $\xrightarrow{L}$ to denote convergence in distribution.

Compared to Pearson's chi-squared test of independence, testing independence using Theorem 1 has more statistical power, particularly when the sample size is small [47]. We test hypothesis $H_0 : MI(X, Y) = 0$ against $H_a : MI(X, Y) > 0$ between the outcome and each of the features. At a user-chosen level of significance (α), any feature whose test decision fails to reject H_0 is eliminated at this stage. It is suggested to let $\alpha = 0.10$. A smaller α increases the chance of Type-II error (eliminating useful features); a larger α reduces the ability of the elimination, which results in a longer selection computation time in the next stage.

Let $X_1, X_2, \ldots, X_s$ denote the s features (out of the p features) that have passed the test of independence. The other $(p - s)$ features are eliminated at this stage. Note that the $X_1, \ldots, X_s$ are temporary notations for features. Namely, the X_1 in $\{X_1, \ldots, X_p\} := \{X\}_p$ and the X_1 in $\{X_1, \ldots, X_s\} := \{X\}_s$ are different if the X_1 in $\{X\}_p$ is eliminated in this stage. Note that we do not consider feature redundancy at Stage 1. Redundant features could all pass the test of independence as long as they appear to be relevant to the outcome based on sample data. Feature redundancy would be considered at Stage 2.

3.3. Stage 2: Selection

In this stage, we make a greedy selection among the s remaining features from Stage 1.
The selection algorithm is:

1. $X_{(1)} = \arg_{X_i \in \{X\}_s} \max\left[\hat{\kappa}^*(X_i, Y)\right]$;

2. $X_{(2)} = \arg_{X_i \in \{X\}_s \setminus \{X_{(1)}\}} \max\left[\hat{\kappa}^*(X_{(1)} \times X_i, Y)\right]$;

3. $X_{(3)} = \arg_{X_i \in \{X\}_s \setminus \{X_{(1)}, X_{(2)}\}} \max\left[\hat{\kappa}^*(X_{(1)} \times X_{(2)} \times X_i, Y)\right]$;

    $\ldots$

The algorithm stops at time c when $\hat{\kappa}^*(X_{(1)} \times \cdots \times X_{(c+1)}, Y) < \hat{\kappa}^*(X_{(1)} \times \cdots \times X_{(c)}, Y)$. The features selected by the proposed method are $X_{(1)}, \ldots, X_{(c)}$. The algorithm workflow is provided in Figure 1.
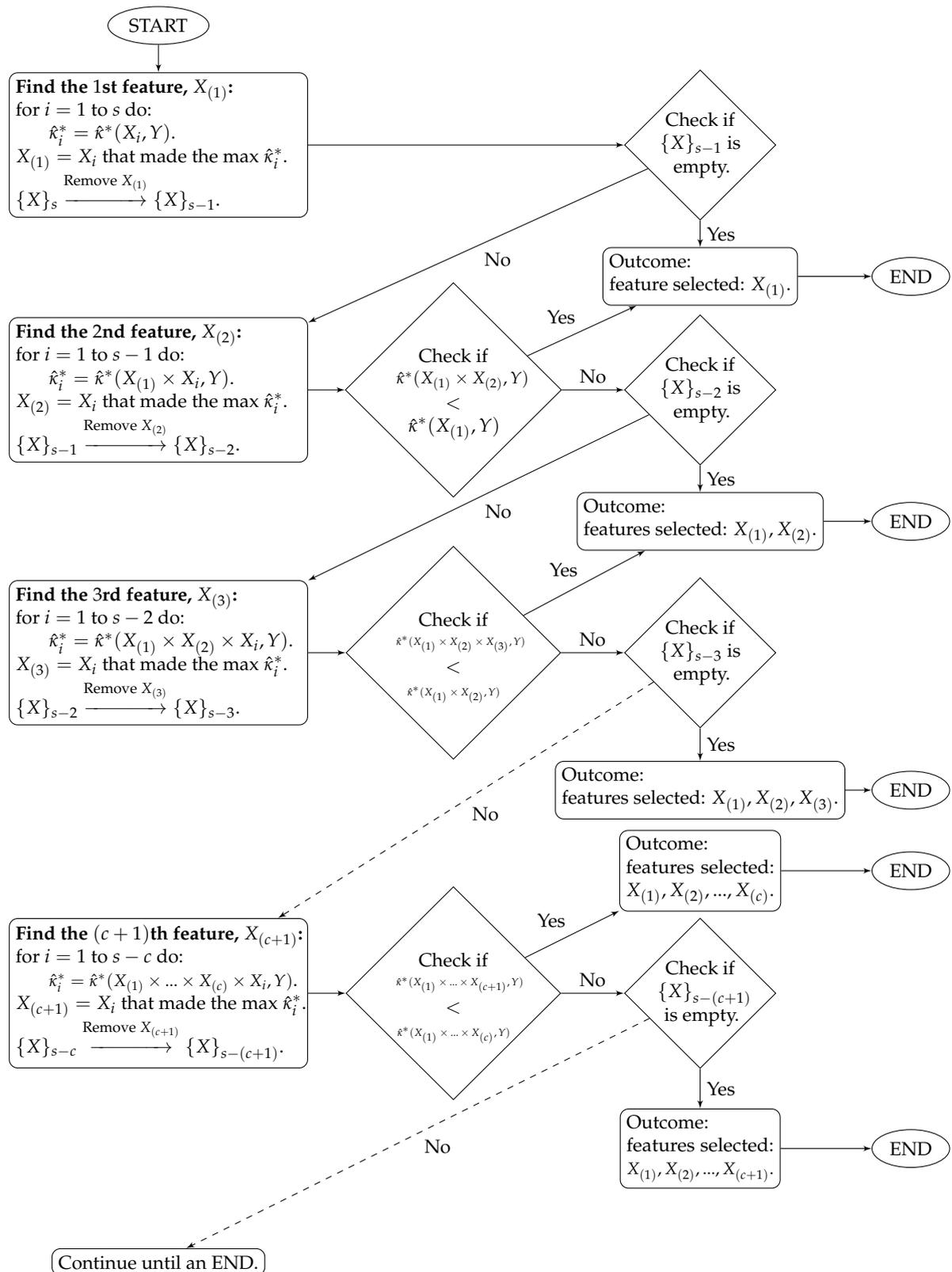
Figure 1. Algorithm workflow for Stage 2.

To clarify the notations, $\hat{\kappa}^*(X_{(1)} \times X_i, Y)$ stands for the estimated CASMI of the joint feature $X_{(1)} \times X_i$ to the outcome Y, and $\{X\}_s$ is the collection of the s remaining features.

The proposed method handles feature redundancy by considering joint-distributions among features. Taking $X_{(1)}$ and $X_{(2)}$ as examples, the first step yields the feature $X_{(1)}$, which is the most relevant feature (measured by the estimated CASMI) to the outcome. In the second step, we join the selected $X_{(1)}$ with each of the remaining $(s-1)$ features, and we evaluate the estimated CASMIs between each of the joint-features and the outcome. The joint-feature with the highest estimated CASMI is selected, which becomes $X_{(2)}$. It should be noted that $X_{(1)}$ and $X_{(2)}$ are neither necessarily independent nor necessarily the least dependent. Selecting $X_{(2)}$ only indicates that based on the information provided from $X_{(1)}$, $X_{(2)}$ provides the most additional information about the outcome among the remaining $(s-1)$ features. In addition, CASMI is an information–theoretic quantity that does not use ordinal information of features; therefore, both linear and nonlinear redundancy are captured, evaluated, and considered.

The proposed method automatically stops at time c when the term $\max[\hat{\kappa}^*(\cdot, Y)]$ starts to decrease. In some situations, a researcher may want to select a desired number of features (d) that is different from c. For example, let $c = 10$, and suppose Researcher A would like to select $d_1 = 6$ features, while Researcher B wants $d_2 = 15$. When $c = 10$ and $d_1 = 6$, because $6 \leq 10$, Researcher A can stop the algorithm at the time 6. When $c = 10$ and $d_2 = 15$, because $15 > 10$, Researcher B needs to select five additional features. We propose two choices on how to select the additional features.

Choice one. Keep running the proposed algorithm until time 15.

Choice two. Use any other user-preferred feature selection methods to select the five additional features.

Choice two could be complicated. If the user-preferred feature selection method has a ranking on the selected features, such as filter methods, then Researcher B can find the additional features by looking for the top five features other than the already-selected 10 features. If the user-preferred feature selection method (e.g., LASSO) does not have a ranking among the selected features, Researcher B can start with selecting 15 features using the preferred method, and then check if there are exactly five new features in the group compared to the 10 features selected by the proposed method. If the number of new features in the group is more than five, then the number of selected features (in the user-preferred method) needs to be reduced, until a point that there are exactly five new features in the group, so that the five additional features can be determined.

After the two stages, the proposed method is completed. The performance of the proposed method is evaluated in the following section.

4. Simulations

In this section, we provide a simulation study to evaluate the performance of the proposed feature selection method. We first discuss the evaluation metric and then introduce the simulation setup and results.

4.1. Evaluation Metric

The proposed feature selection method selects only relevant features but does not provide an associated model or classifier. In evaluating such a feature selection method, there are two possible approaches [48]. The first approach is to embed a classifier and compare the accuracy of the classification process based on a real dataset. The results obtained with this approach are difficult to generalize as they depend on the specific classifier used in the comparison. The second approach is based on a scenario defined by an initial set of features and a relation between these features and the outcome. Under this situation, a feature selection method could be evaluated by the truth. Focusing on the evaluation of the selected features, we adopt the second approach to evaluate the proposed feature selection method based on the truth. Under this approach, there are several

strategies. One can calculate the percentage (success rate) of all relevant features that are selected. For example, let us consider an outcome T that is relevant to three features F_1, F_2, and F_3, where F_1 contributes the most information (variability) of T, F_2 contributes the second most, and F_3 contributes the least. Also, there is an irrelevant feature F_4 in the dataset. Suppose there are four different selection results: $S_1 = \{F_1\}$, $S_2 = \{F_1, F_4\}$, $S_3 = \{F_2, F_4\}$, and $S_4 = \{F_3, F_4\}$. Evaluating their performances using the success rate would achieve the same result (33.3% or 1/3) for all of them as they all identify one correct feature out of the three. The success rate is simple to calculate because the ground truth is known, and it works well when we focus on the number of correctly selected features or if we assume all the relevant features contribute evenly to the outcome. However, under the restriction of a limited sample, it may be more important to select the group of features that could jointly and efficiently provide the most information instead of selecting all relevant features regardless of the degrees of relevance and redundancy. Although ignoring low relevant or vastly redundant features may lose information, dropping them would further reduce the dimensionality and benefit the estimation. This can be considered as a trade-off between estimation (noise) and dimensionality (information): the higher the dimensionality, the harder the estimation; the more the information, the more the noise. When the estimation is overly difficult, or when there is too much noise, the results could be biased and hardly generalizable.

Because the success rate does not take the degrees of relevance and redundancy into consideration, we introduce the following evaluation metric to measure the ratio of the relevant information from the joint of selected features to the total relevant information from the joint of all the relevant features using mutual information.

Definition 2 (Information Recovery Ratio (IRR))**.**

$$IRR = \frac{MI(\mathcal{X}_{selected}, Y)}{MI(\mathcal{X}_{relevant}, Y)}, \tag{10}$$

where $\mathcal{X}_{selected}$ is the random variable that follows the joint-distribution of the selected features, and $\mathcal{X}_{relevant}$ is the random variable that follows the joint-distribution of all the features on which Y depends.

The IRR is not calculable in real datasets because (1) there is no knowledge on which features are relevant to the outcome, and (2) the true underlying distributions and associations (including redundancy) of the features and outcomes in real data are unknown. Given the setup of a simulation, we have all the knowledge; hence, the IRR for any group of selected features is calculable.

The IRR represents the percentage of relevant information in the joint of selected features. It considers feature redundancy by evaluating the mutual information between the joint-feature and the outcome. The range of the IRR is $[0, 1]$. If no relevant features are selected, the IRR is 0. If all the features in the dataset are selected regardless of relevance, the IRR is 1 for certain; therefore, when comparing the performance using the IRR, the number of selected features must be controlled. When the numbers of selected features from different methods are the same, a larger IRR means the joint of the selected features contains more relevant information; hence, the method is more efficient in dimension reduction. The efficiency of a feature selection method is desirable, particularly under small samples.

To make a comparison between the IRR and the success rate, both evaluate the performance of feature selection methods only when the ground truth is known. The success rate focuses on the ratio of the number of relevant features selected to the total number of relevant features, while the IRR focuses on the ratio of the relevant information in the joint of the selected features to the total relevant information.

4.2. Simulation Setup

A good evaluation scenario must include a representative set of features, containing relevant, redundant, and irrelevant ones [48]. In the simulation, we generate 10 X variables ($X_1, \ldots, X_{10}$) and one outcome (Y). Among these variables, X_1, X_2, X_3, X_4 (or X_6), and X_5 are relevant features; X_6 (or X_4) is a redundant feature; X_7, X_8, X_9, and X_{10} are irrelevant features. The detailed settings are as follows.

$$Y = X_1 + X_2 + X_3^3 - 0.5 \cdot X_4^2 + |X_5| + X_6 + \varepsilon, \tag{11}$$

where

$$
\begin{aligned}
X_1 = &- 3.5 \cdot \mathbb{1}[Z_1 < -3] - 1.4 \cdot \mathbb{1}[-3 \leq Z_1 \leq -0.5] \\
&+ \mathbb{1}[0.5 \leq Z_1 \leq 3] + 2.2 \cdot \mathbb{1}[Z_1 > 3], \\
X_2 = &- 5 \cdot \mathbb{1}[Pois_1 = 0] - 3 \cdot \mathbb{1}[Pois_1 = 1] \\
&+ 2.4 \cdot \mathbb{1}[Pois_1 = 3 \text{ or } 4] + 5.4 \cdot \mathbb{1}[Pois_1 \geq 5], \\
X_3 = &- 2 \cdot \mathbb{1}[U_1 \leq -0.6] - \mathbb{1}[-0.6 < U_1 < -0.2] \\
&+ \mathbb{1}[0.2 < U_1 < 0.6] + 2 \cdot \mathbb{1}[U_1 \geq 0.6], \\
X_4 = &(B_1 - 2) \cdot \mathbb{1}[B_1 \neq 4] + 5 \cdot \mathbb{1}[B_1 = 4], \\
X_5 = &- 2.5 \cdot \mathbb{1}[Z_2 < -0.5] - 2 \cdot \mathbb{1}[-0.5 \leq Z_2 \leq -0.2] \\
&+ 1.7 \cdot \mathbb{1}[-0.2 \leq Z_2 \leq 0.2] + 2 \cdot \mathbb{1}[0.2 \leq Z_2 \leq 0.6] \\
&+ 4 \cdot \mathbb{1}[Z_2 > 0.6], \\
X_6 = &X_4, \\
X_7 = &(Pois_2 - 2) \cdot \mathbb{1}[Pois_2 < 2] + 2 \cdot \mathbb{1}[Pois_2 \geq 2], \\
X_8 = &- 2 \cdot \mathbb{1}[U_2 \leq -0.6] - \mathbb{1}[-0.6 < U_2 < -0.2] \\
&+ \mathbb{1}[0.2 < U_2 < 0.6] + 2 \cdot \mathbb{1}[U_2 \geq 0.6], \\
X_9 = &B_2 - 1.2, \\
X_{10} = &- 2 \cdot \mathbb{1}[Z_3 < -1.5] - 1.5 \cdot \mathbb{1}[-1.5 \leq Z_3 \leq -0.7] \\
&+ 1.5 \cdot \mathbb{1}[0.7 \leq Z_3 \leq 1.5] + 2 \cdot \mathbb{1}[Z_3 > 1.5], \\
\varepsilon = &- \mathbb{1}[U_3 \leq \tfrac{1}{3}] + \mathbb{1}[U_3 \geq \tfrac{2}{3}],
\end{aligned}
$$

and

$$
\begin{aligned}
Z_1, Z_2, Z_3 &\sim N(0, 1), \\
Pois_1, Pois_2 &\sim Poisson(2), \\
B_1 &\sim Binomial(4, 0.1), \\
B_2 &\sim Binomial(6, 0.2), \\
U_1, U_2 &\sim Uniform(-1, 1), \\
U_3 &\sim Uniform(0, 1).
\end{aligned}
$$

This simulation setup is fair for all methods compared. The settings were not designed to be more complex is because of the following reasons.

1. No matter whether the settings are simple or complex, it is fair to all methods. The purpose of this simulation is to evaluate the performance of the proposed method, particularly when the sample size is relatively small. The complexity of the feature spaces and the relationships among the features and the outcome would determine the threshold of what constitutes a sufficiently large sample. As they are not complex, we can sample with smaller sizes to evaluate the performances in simulation. This is true to all feature selection methods in comparison as they select features based on the same sample data with the same sample size.

2. Ordinal information in the settings will not affect simulation results. The proposed feature selection method is one of the entropic methods. In the simulation, we would compare the performance of the proposed method to only other entropic methods because of the domain-specific challenges discussed in Section 1. During the simulation, we assign numerical values to the X variables so that we can generate the value of the outcome Y based on a model. However, entropic methods do not use the ordinal information from the numerical data because the inputs of the entropic methods are the frequencies of different classes. Therefore, involving a complicated model (linear or nonlinear) does not affect the entropic methods because they regard the numbers as labels without ordinal information. However, complicating the model could make the outcome variable Y more complex and result in a higher threshold of a sufficiently large sample, which does not affect the comparison and evaluation among different methods, as discussed previously.

3. These settings lead to an efficient calculation for ground truth during the simulation. In calculating IRR, we need the two joint-distributions, $\mathcal{X}_{selected} \times Y$ and $\mathcal{X}_{relevant} \times Y$. To obtain the true joint-distributions, we have to enumerate the combinations among all possible values of the selected relevant features and of all the relevant features with their probabilities, respectively. Complicating the relevant X variables would make the calculation of the joint-distributions unnecessarily complex.

Note that, in real-world data, we do not need such calculations in reason three because the true joint-distributions and the IRR are not calculable. Hence, when applying the proposed method on real-world high dimensional and complex data, the main calculation is just estimating CASMI, which can be calculated in seconds.

With the simulation setup, one can consider that we create a dataset for evaluation. In this case, we know the ground truth that the features X_1, X_2, X_3, X_4 (or X_6), and X_5 should be selected. We would evaluate the performances by calculating the IRRs for features selected by different methods.

4.3. Simulation Results

In the simulation, we compare the IRR of the proposed feature selection method to the IRRs of six widely cited entropic feature selection methods: MIM, JMI, CMIM, MRMR, DISR, and NJMIM.

These six entropic methods all require users to set the number of features to be selected, while the proposed method can automatically decide the most appropriate number of features based on data. As we must control the number of selected features to validate the comparison of IRRs, we use the number of selected features from the proposed method as the number of features to be selected in the six entropic methods in each iteration. It should be noted that we are not claiming the number of features determined by the proposed method is correct. We set them to be the same for only the purpose of validating the comparison. As a matter of fact, the relevant features would not be entirely selected until the sample size is sufficiently large, and the threshold of a sufficiently large sample varies from method to method.

For each sample size N in $\{50, 100, 150, \ldots, 2750, 2800\}$, we re-generate the entire dataset 10,000 times and calculate the average IRRs of each method. The average IRR results are plotted in Figure 2.
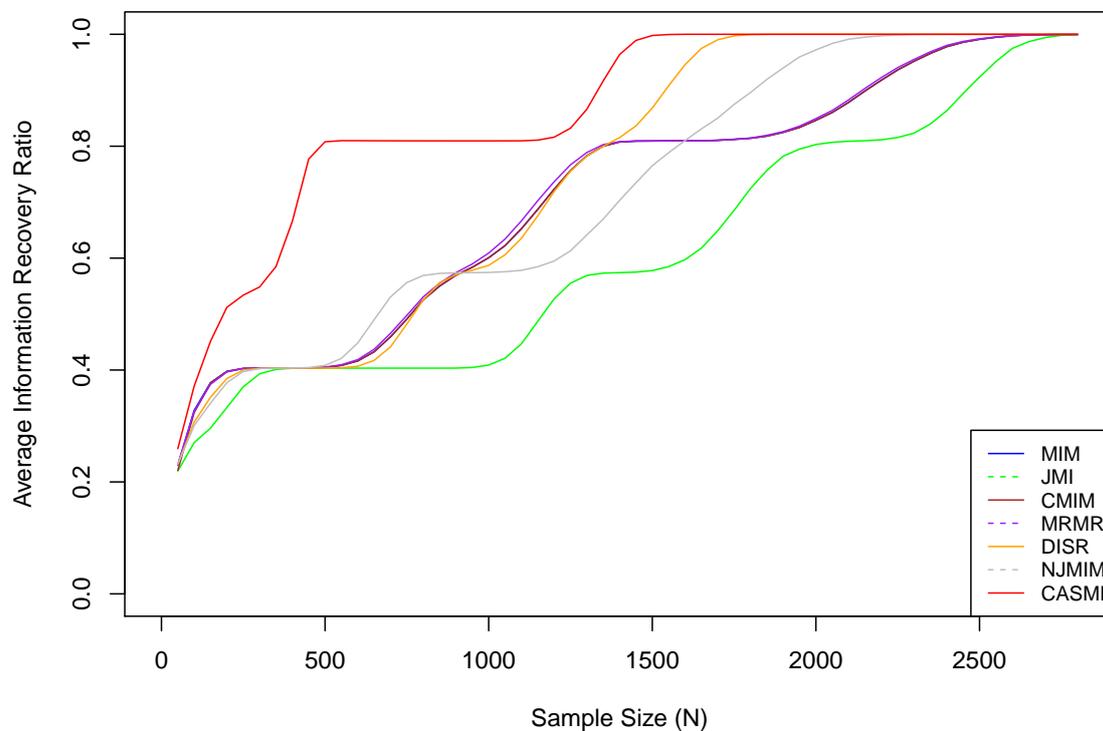
Figure 2. The average IRRs for seven methods, where CASMI refers to the proposed method. The proposed method is the most efficient method when the sample size is limited. In the simulation, the threshold of a sufficiently large sample for the proposed method is approximately $N = 1500$, which is the smallest among all methods. The vertical index is the IRR, not the success rate. An IRR of 0.8 means 80% of the total mutual information has been accounted for by the selected features. It does not mean 80% of relevant features are selected. The proposed method does not select all relevant features when the sample size is small because some relevant features are in situation two under a limited sample (eliminating situation two features is discussed in Section 1). As the sample size grows, all situation two features eventually become situation one features.

Based on the results, we can see that the average IRR of the proposed method is consistently higher than or equivalent to all the other methods. This is because, under the restriction of a limited sample, the proposed method has a much smaller estimation bias so that it captures the associations among features and the outcome more accurately than the existing methods that estimate with the plug-in estimators. Table 2 presents the 95% confidence intervals for IRRs based on features selected by different methods under different sample sizes. Based on the table, we can roughly rank the proposed methods and the six methods as follows: CASMI > DISR > NJMIM > MRMR > MIM ~ CMIM > JMI.

Meanwhile, we recorded the average computation time of the proposed method when implementing feature selection in R. The plot of results is shown in Figure 3. The computation time when $N = 50$ was 0.03 s; the time when $N = 2800$ was 1.97 s; the longest time during the simulation was 3.37 s.
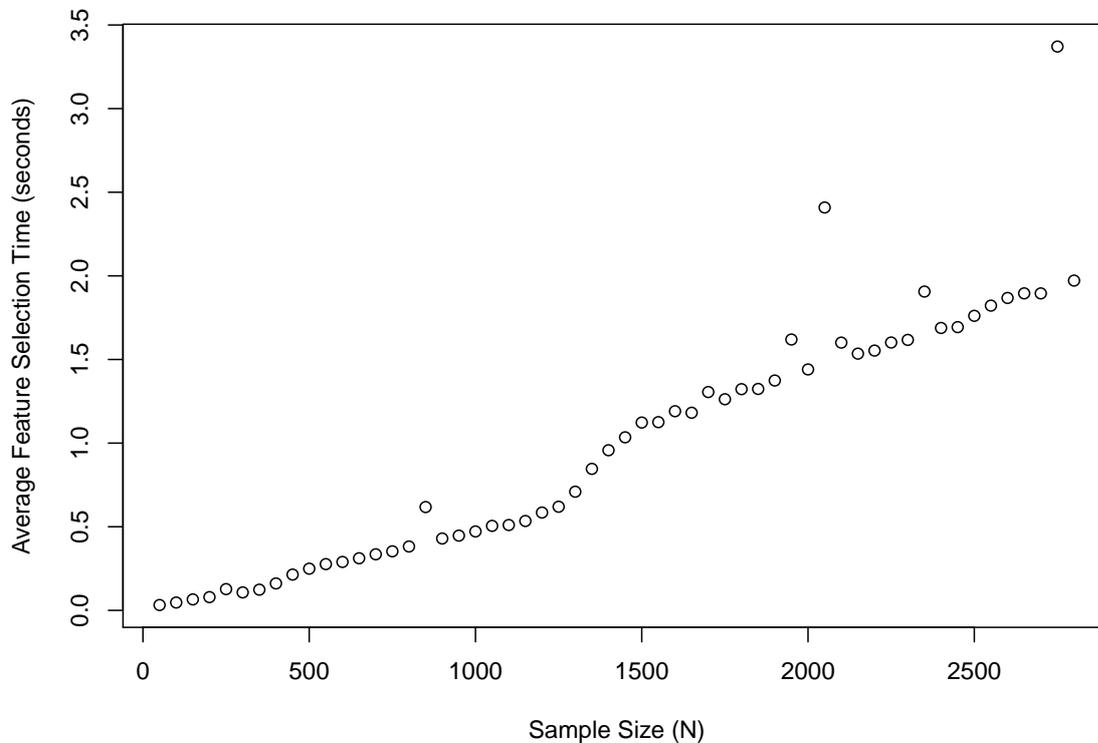
Figure 3. The average computation time of the proposed method when implementing feature selection in R.

Based on the simulation result in Figure 2, different methods achieve 1 (on average IRRs) at different sample sizes. One should realize that the threshold of a sufficiently large sample greatly depends on the probability spaces of the underlying associated features and the outcome. The probability spaces of real datasets are generally significantly more complicated than that of the simulation. Consequently, in reality, particularly in health data, the majority of samples should be considered small; hence, the efficiency of a feature selection method is very important.

The simulation codes are available at [49]. The proposed feature selection method using CASMI is implemented in the R package at [50].

Table 2. The 95% Confidence Intervals for IRRs based on features selected by different methods under different sample sizes.

n	CASMI CI	MIM CI	JMI CI	CMIM CI	MRMR CI	DISR CI	NJMIM CI
50	[0, 0.40]	[0, 0.40]	[0, 0.40]	[0, 0.40]	[0, 0.40]	[0, 0.40]	[0, 0.40]
100	[0.21, 0.57]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]
150	[0.26, 0.62]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]
200	[0.32, 0.62]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]	[0.21, 0.40]
250	[0.37, 0.62]	[0.40, 0.40]	[0.21, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.21, 0.40]
300	[0.37, 0.62]	[0.40, 0.40]	[0.21, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]
350	[0.37, 0.81]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]
400	[0.37, 0.81]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]
450	[0.62, 0.81]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]
500	[0.81, 0.81]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.40]	[0.40, 0.57]
550	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]
600	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]
650	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]
700	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]
750	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]
800	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]
850	[0.81, 0.81]	[0.40, 0.57]	[0.40, 0.40]	[0.40, 0.57]	[0.40, 0.57]	[0.40, 0.57]	[0.57, 0.57]
900	[0.81, 0.81]	[0.40, 0.62]	[0.40, 0.40]	[0.40, 0.62]	[0.40, 0.81]	[0.40, 0.57]	[0.57, 0.57]
950	[0.81, 0.81]	[0.57, 0.81]	[0.40, 0.40]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.57]	[0.57, 0.57]
1000	[0.81, 0.81]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.57]
1050	[0.81, 0.81]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.57]
1100	[0.81, 0.81]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.57]
1150	[0.81, 0.81]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]
1200	[0.81, 1]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]
1250	[0.81, 1]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]
1300	[0.81, 1]	[0.57, 0.81]	[0.40, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]
1350	[0.81, 1]	[0.57, 0.81]	[0.57, 0.57]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]	[0.57, 0.81]
1400	[0.81, 1]	[0.81, 0.81]	[0.57, 0.57]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 0.81]
1450	[0.81, 1]	[0.81, 0.81]	[0.57, 0.57]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 0.81]
1500	[1, 1]	[0.81, 0.81]	[0.57, 0.57]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 0.81]
1550	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 1]
1600	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 1]
1650	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.57, 1]
1700	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 0.81]	[0.81, 1]	[0.81, 1]
1750	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 0.81]	[1, 1]	[0.81, 1]
1800	[1, 1]	[0.81, 0.81]	[0.57, 0.81]	[0.81, 0.81]	[0.81, 1]	[1, 1]	[0.81, 1]
1850	[1, 1]	[0.81, 1]	[0.57, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
1900	[1, 1]	[0.81, 1]	[0.57, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
1950	[1, 1]	[0.81, 1]	[0.57, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
2000	[1, 1]	[0.81, 1]	[0.57, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
2050	[1, 1]	[0.81, 1]	[0.81, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
2100	[1, 1]	[0.81, 1]	[0.81, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
2150	[1, 1]	[0.81, 1]	[0.81, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[0.81, 1]
2200	[1, 1]	[0.81, 1]	[0.81, 0.81]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2250	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2300	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2350	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2400	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2450	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2500	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]
2550	[1, 1]	[0.81, 1]	[0.81, 1]	[0.81, 1]	[1, 1]	[1, 1]	[1, 1]
2600	[1, 1]	[1, 1]	[0.81, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]
2650	[1, 1]	[1, 1]	[0.81, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]
2700	[1, 1]	[1, 1]	[0.81, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]
2750	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]
2800	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]	[1, 1]

5. Discussion

In this article, we have proposed a new entropic feature selection method based on CASMI. Compared to existing methods, the proposed method has two unique advantages: (1) it is very efficient as the joint of selected features provides the most relevant information compared to features selected by other methods, particularly when the sample size is relatively small, and (2) it automatically learns the number of features to be selected from data. The proposed method handles feature redundancy from the perspective of joint-distributions. Although we initially developed the proposed method for the domain-specific challenges in healthcare, the proposed method can be used in many other areas where there is an issue of a limited sample.

The proposed method is an entropic information–theoretic method. It aims at assisting data analytics on non-ordinal spaces. However, the proposed method can also be used on numerical data with an appropriate binning technique. Furthermore, using the proposed method on binned numerical data could discover different information as the entropic method looks at the data from a non-ordinal perspective.

In detecting unhelpful associations (situation two and three features), we implement an adjustment from the sample coverage. The level of this adjustment can be modified by users. For example, users can replace the scoring function of the proposed method by CASMI* with a tuning parameter (u) as follows:

$$\kappa^*(X, Y) = \kappa(X, Y) \cdot (1 - \pi_0(X))^u,$$

and estimate it by

$$\hat{\kappa}^*(X, Y) = \hat{\kappa}_z(X, Y) \cdot (1 - T_1(X))^u,$$

where u is any fixed positive number. The u can be considered as a parameter to determine the requirement for a feature to qualify situation one. A larger u stands for a heavier penalty from the sample coverage; hence, a feature needs to contain more real information to be categorized to situation one. A smaller u stands for a less penalty from the sample coverage; hence, a feature with less real information could be categorized to situation one. However, users should be cautious when using a small u because it may mistakenly classify an irrelevant feature (situation three) to situation one, and further exacerbates the issue of generalization. We suggest beginning the proposed feature selection method with $u = 1$. After completing feature selection, if a user desires to select more or less features, the user could re-run the proposed method with a smaller or larger u, respectively, and keep modifying the value of u until satisfactory.

The proposed method only selects features but does not provide a classifier; however, to draw inferences on outcomes, a classifier is needed. To this end, additional techniques are required, such as machine learning (e.g., regressions and random forest). Into the future, it may be interesting to explore (1) methods that can distinguish features under situation two and three when the sample size is small; and (2) the possibilities of extending the proposed method to tree-based algorithms (e.g., random forest) to help determine which leaves and branches should be omitted. In addition, it may be interesting to investigate the performance of existing entropic methods if we use the $\hat{H}_z$, instead of $\hat{H}$, to estimate the entropies in their score functions.

References

1. Kruse, C.S.; Goswamy, R.; Raval, Y.; Marawi, S. Challenges and opportunities of big data in health care: A systematic review. *JMIR Med. Inform.* **2016**, *4*, e38. [CrossRef] [PubMed]

2. Lee, C.H.; Yoon, H.J. Medical big data: Promise and challenges. *Kidney Res. Clin. Pract.* **2017**, *36*, 3. [CrossRef] [PubMed]

3. Blum, A.L.; Langley, P. Selection of relevant features and examples in machine learning. *Artif. Intell.* **1997**, *97*, 245–271. [CrossRef]

4. Kohavi, R.; John, G.H. Wrappers for feature subset selection. *Artif. Intell.* **1997**, *97*, 273–324. [CrossRef]

5. Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R.P.; Tang, J.; Liu, H. Feature selection: A data perspective. *ACM Comput. Surv. (CSUR)* **2017**, *50*, 94. [CrossRef]

6. Urbanowicz, R.J.; Meeker, M.; La Cava, W.; Olson, R.S.; Moore, J.H. Relief-based feature selection: introduction and review. *J. Biomed. Inform.* **2018**, *85*, 189–203. [CrossRef]

7. Cai, J.; Luo, J.; Wang, S.; Yang, S. Feature selection in machine learning: A new perspective. *Neurocomputing* **2018**, *300*, 70–79. [CrossRef]

8. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]

9. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*; John Wiley & Sons: Hoboken, NJ, USA, 2012.

10. Robnik-Šikonja, M.; Kononenko, I. Theoretical and empirical analysis of ReliefF and RReliefF. *Mach. Learn.* **2003**, *53*, 23–69. [CrossRef]

11. Nie, F.; Xiang, S.; Jia, Y.; Zhang, C.; Yan, S. Trace ratio criterion for feature selection. In Proceedings of the 23rd national conference on Artificial intelligence, Chicago, IL, USA, 13–17 July 2008; pp. 671–676.

12. He, X.; Cai, D.; Niyogi, P. Laplacian score for feature selection. In *Advances in Neural Information Processing Systems*; Jordan, M.I., LeCun, Y., Solla, S.A., Eds.; NIPS: Vancouver, BC, Canda, 2006; pp. 507–514.

13. Zhao, Z.; Liu, H. Spectral feature selection for supervised and unsupervised learning. In Proceedings of the 24th International Conference on Machine Learning, Corvalis, ON, USA, 20–24 June 2007; pp. 1151–1157.

14. Liu, J.; Ji, S.; Ye, J. SLEP: Sparse learning with efficient projections. *Arizona State Univ.* **2009**, *6*, 7.

15. Nie, F.; Huang, H.; Cai, X.; Ding, C.H. Efficient and robust feature selection via joint 2, 1-norms minimization. In *Advances in Neural Information Processing Systems*; NIPS: Vancouver, BC, Canda, 2010; pp. 1813–1821.

16. Cai, D.; Zhang, C.; He, X. Unsupervised feature selection for multi-cluster data. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; pp. 333–342.

17. Yang, F.; Mao, K. Robust feature selection for microarray data based on multicriterion fusion. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2011**, *8*, 1080–1092. [CrossRef] [PubMed]

18. Li, Z.; Yang, Y.; Liu, J.; Zhou, X.; Lu, H. Unsupervised feature selection using nonnegative spectral analysis. In Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, Toronto, ON, Canada, 22–26 July 2012.

19. Davis, J.C.; Sampson, R.J. *Statistics and Data Analysis in Geology*; Wiley: New York, NY, USA, 1986; Volume 646.

20. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B (Methodol.)* **1996**, *58*, 267–288. [CrossRef]

21. Lewis, D.D. Feature selection and feature extraction for text categorization. In Proceedings of the workshop on Speech and Natural Language. Association for Computational Linguistics, Harriman, NY, USA, 23–26 February 1992; pp. 212–217.

22. Battiti, R. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. Neural Netw.* **1994**, *5*, 537–550. [CrossRef] [PubMed]

23. Yang, H.H.; Moody, J. Data visualization and feature selection: New algorithms for nongaussian data. In *Advances in Neural Information Processing Systems*; NIPS: Denver, CO, USA, 2000; pp. 687–693.

24. Vidal-Naquet, M.; Ullman, S. Object Recognition with Informative Features and Linear Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nice, France, 13–16 October 2003; Volume 3, p. 281.

25. Fleuret, F. Fast binary feature selection with conditional mutual information. *J. Mach. Learn. Res.* **2004**, *5*, 1531–1555.

26. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [CrossRef] [PubMed]

27. Lin, D.; Tang, X. Conditional infomax learning: An integrated framework for feature extraction and fusion. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; Springer: Berlin, Germany, 2006; pp. 68–82.

28. Meyer, P.E.; Bontempi, G. On the use of variable complementarity for feature selection in cancer classification. In Proceedings of the Workshops on Applications of Evolutionary Computation, Budapest, Hungary, 10–12 April 2006; Springer: Berlin, Germany, 2006; pp. 91–102.

29. Bennasar, M.; Hicks, Y.; Setchi, R. Feature selection using joint mutual information maximisation. *Expert Syst. Appl.* **2015**, *42*, 8520–8532. [CrossRef]

30. Liu, H.; Setiono, R. Chi2: Feature selection and discretization of numeric attributes. In Proceedings of the 7th IEEE International Conference on Tools with Artificial Intelligence, Herndon, VA, USA, 5–8 November 1995; pp. 388–391.

31. Gini, C. Variabilita e mutabilita, Studi Economico-Giuridici della R. *Univ. Cagliari* **1912**, *3*, 3–159.

32. Hall, M.A.; Smith, L.A. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper. In Proceedings of the FLAIRS Conference, Orlando, FL, USA, 1–5 May 1999; Volume 1999, pp. 235–239.

33. Harris, B. *The Statistical Estimation of Entropy in the Non-Parametric Case*; Technical Report; Wisconsin Univ-Madison Mathematics Research Center: Madison, WI, USA, 1975.

34. Paninski, L. Estimation of entropy and mutual information. *Neural Comput.* **2003**, *15*, 1191–1253. [CrossRef]

35. Good, I.J. The population frequencies of species and the estimation of population parameters. *Biometrika* **1953**, *40*, 237–264. [CrossRef]

36. Kullback, S.; Leibler, R.A. On information and sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [CrossRef]

37. Wyner, A.D. A definition of conditional mutual information for arbitrary ensembles. *Inf. Control* **1978**, *38*, 51–59. [CrossRef]

38. Guiasu, S. *Information Theory with Applications*; McGraw-Hill: New York, NY, USA, 1977; Volume 202.

39. Zhang, Z. *Statistical Implications of Turing's Formula*; John Wiley & Sons: Hoboken, NJ, USA, 2016.

40. Ohannessian, M.I.; Dahleh, M.A. Rare probability estimation under regularly varying heavy tails. In Proceedings of the 25th Conference on Learning Theory (COLT 2012), Edinburgh, Scotland, 25–27 June 2012; pp. 21.1–21.24.

41. Zhang, Z. Entropy estimation in Turing's perspective. *Neural Comput.* **2012**, *24*, 1368–1389. [CrossRef] [PubMed]

42. Zhang, Z.; Zheng, L. A mutual information estimator with exponentially decaying bias. *Stat. Appl. Genet. Mol. Biol.* **2015**, *14*, 243–252. [CrossRef] [PubMed]

43. Dougherty, J.; Kohavi, R.; Sahami, M. Supervised and unsupervised discretization of continuous features. In *Machine Learning Proceedings 1995*; Elsevier: Amsterdam, The Netherlands, 1995; pp. 194–202.

44. Rubin, D.B. Inference and missing data. *Biometrika* **1976**, *63*, 581–592. [CrossRef]

45. Little, R.J.; D'agostino, R.; Cohen, M.L.; Dickersin, K.; Emerson, S.S.; Farrar, J.T.; Frangakis, C.; Hogan, J.W.; Molenberghs, G.; Murphy, S.A.; et al. The prevention and treatment of missing data in clinical trials. *N. Engl. J. Med.* **2012**, *367*, 1355–1360. [CrossRef]

46. Kang, H. The prevention and handling of the missing data. *Korean J. Anesthesiol.* **2013**, *64*, 402–406. [CrossRef]

47. Zhang, J.; Chen, C. On 'A mutual information estimator with exponentially decaying bias' by Zhang and Zheng. *Stat. Appl. Genet. Mol. Biol.* **2018**, *17*. [CrossRef]

48. Pascoal, C.; Oliveira, M.R.; Pacheco, A.; Valadas, R. Theoretical evaluation of feature selection methods based on mutual information. *Neurocomputing* **2017**, *226*, 168–181. [CrossRef]

49. Shi, J. CASMI Simulation R Codes. 2019. Available online: https://github.com/JingyiShi/CASMI/blob/master/SimulationEvaluationUsingGroundTruth.R (accessed on 1 November 2019).

50. Shi, J. CASMI in R. 2019. Available online: https://github.com/JingyiShi/CASMI (accessed on 1 November 2019).