Pattern Recognition 91 (2019) 162–174

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Simultaneous feature selection and discretization based on mutual information

Sadia Sharmin[a,*], Mohammad Shoyaib[a], Amin Ahsan Ali[b], Muhammad Asif Hossain Khan[c], Oksam Chae[d]

[a] Institute of Information Technology, University of Dhaka, Dhaka, Bangladesh
[b] Department of Computer Science and Engineering, Independent University, Bangladesh
[c] Department of Computer Science and Engineering, University of Dhaka, Dhaka, Bangladesh
[d] Department of Computer Engineering, Kyung Hee University, Gyeonggido, South Korea

ARTICLE INFO

Article history:
Received 16 August 2017
Revised 14 February 2019
Accepted 19 February 2019
Available online 19 February 2019

Keywords:
Feature selection
Mutual information
Bias
Dynamic discretization

ABSTRACT

Recently mutual information based feature selection criteria have gained popularity for their superior performances in different applications of pattern recognition and machine learning areas. However, these methods do not consider the correction while computing mutual information for finite samples. Again, finding appropriate discretization of features is often a necessary step prior to feature selection. However, existing researches rarely discuss both discretization and feature selection simultaneously. To solve these issues, Joint Bias corrected Mutual Information (JBMI) is firstly proposed in this paper for feature selection. Secondly, a framework namely modified discretization and feature selection based on mutual information is proposed that incorporates JBMI based feature selection and dynamic discretization, both of which use a χ^2 based searching method. Experimental results on thirty benchmark datasets show that in most of the cases, the proposed methods outperform the state-of-the-art methods.

1. Introduction

Feature selection is the process of selecting the important features and removing the unnecessary ones from the original feature set. It has been explored extensively over the last decades and is considered to be one of the most important topics in the field of machine learning, data mining and pattern recognition. This is because, in a given set of features, not all the features are necessary rather some are noisy or redundant which may even degrade the classification performance. The resulting selected feature set is expected to increase the overall classification accuracy, enable learning algorithms to train faster, reduce the risk of overfitting, reduce the overall computational cost of prediction models, and help unsupervised learning tasks (e.g., clustering) etc. Feature selection also facilitates data visualization as well as data understanding [1] and, therefore, has been applied in several domains such as bio-informatics [2], computer science [3–5], and physics [6,7].

In general, feature selection methods can be divided into three categories – filter, wrapper, and embedded. Filter methods are classifier independent, whereas wrapper and embedded methods are classifier dependent [8]. Wrapper methods utilize a classifier to select the feature subset. Some examples of wrapper methods are GA-KDE-Bayes [9], SAL [10] and [11]. These methods may become biased to a classifier and also, they are prone to over-fitting [12], especially when the amount of data is small. Embedded methods [13] use classifiers during the training phase and perform feature selection as a part of learning procedure. It uses a cost function to guide the feature search. They are faster and less prone to overfitting compared to wrapper methods [14]. Conversely, filter methods are more popular as they assess the goodness of features based on an evaluation criteria. Since filter based methods are not biased to any classification algorithm and computationally less expensive, they are preferable for handling large feature space problems [15]. Among the filter methods, Mutual Information (MI) based methods have gained popularity due to their ability to capture non-linear relationship between variables. Another attractive property of MI is that it can be computed for both categorical and numerical variables and it can deal with multiple classes [8,16].

This paper mainly focuses on MI based filter methods that have the goal to find a subset of features S from a given set F based on some criteria in such a way that it contains only the relevant

* Corresponding author.
E-mail addresses: bit0426@iit.du.ac.bd (S. Sharmin), shoyaib@du.ac.bd (M. Shoyaib), aminali@iub.edu.bd (A.A. Ali), asif@du.ac.bd (M.A.H. Khan), oschae@khu.ac.kr (O. Chae).

https://doi.org/10.1016/j.patcog.2019.02.016
0031-3203/

features for classification and discards the irrelevant and redundant ones. Again, there might exist some features $f_i \in F$ that are redundant, but for a given set of selected features it may contain Complementary Information (CI). Therefore, to select a feature, the following three issues should be considered – maximize relevancy (R) with class variable C, minimize redundancy (r) among the features, and maximize CI (c) [12,17]. In this paper, these properties will be referred as *Rrc* criteria. However, evaluating the *Rrc* criteria efficiently is challenging due to its high computational cost.

Formally, for MI based feature section methods, the main objective is to find a subset of features $S = \{f_1, \ldots, f_k\}$ from a given set F of n features, which have the maximum dependency with the target class C, i.e., find S that maximizes the value of $I(S; C)$ given in (1).

$$I(S;C) = I(f_1, \ldots, f_k; C) = \sum_{f_1,\ldots,f_k} \sum_{C} P(f_1, \ldots, f_k; C) \log \frac{P(f_1, \ldots, f_k; C)}{P(f_1, \ldots, f_k)P(C)} \quad (1)$$

However, the computation of joint MI, $I(S; C)$ in (1) is computationally expensive [18] and therefore, most of the methods propose to add features to S incrementally. Given a set of selected features S, the i^{th} feature f_i is selected that maximizes $I(S \cup f_i; C)$. There exists methods that approximate $I(S \cup f_i; C)$ with only R value (e.g., Mutual Information Maximization (MIM) [19]), or with R and r values (e.g., Mutual Information based Feature Selection (MIFS) [20] and minimal-redundancy-maximal-relevance (mRMR) [18]). The importance of CI has been discussed in several works and one of the earliest such proposals is Joint Mutual Information (JMI) [21]. Following [12], the selection criteria for i^{th} feature f_i used in JMI can be defined as (2)

$$max \quad \left[I(f_i; C) + \frac{1}{|S|} \sum_{f_s \in S} \Big(I(f_i; f_s \mid C) - I(f_i; f_s) \Big) \right] \quad (2)$$

where, $I(f_i; C)$ is relevancy between f_i and C, $I(f_i; f_s)$ represents redundancy between f_i and f_s, and $I(f_i; f_s|C)$ is the CI between f_i and f_s given C. Here, (2) represents the *Rrc* criteria. However, one of the major drawbacks of (2) is that there exists some error (bias) when one calculates the value of MI for finite number of samples [22,23]. Thus, to use the *Rrc* criteria, this bias should be addressed in order to achieve a more accurate estimation of MI.

Another important issue with MI based methods is the use of discretization which is usually applied prior to the feature selection for transforming the continuous valued features into discrete ones. Discretization is a process where one has to discretize a given feature into n discrete intervals $\{ [d_0, d_1], [d_1, d_2], \ldots, [d_{n-1}, d_n] \}$, where d_0 and d_n are the minimum and the maximum values of f_i respectively and d_is are the interval points. It is necessary to add discretization (even for discretized data) to simplify the data, accelerate the learning process, and to reduce noise in data [24]. Discretization methods can be categorized into two types: static and dynamic. In static discretization (SD), discretization is performed considering one feature at a time that maximizes the relevance between f_i and C. However, generally a set of features are usually given and it is necessary to consider all the features in F during the discretization process. This is because, when the discretization levels are changed, considering the previously discretized features, it is possible to incorporate additional information. This type of discretization is called dynamic discretization (DD).

The major goal of feature discretization methods is to discretize a feature in such a way that keeps the number of intervals as few as possible, lowers the inconsistency rate (two identical instances are classified into two different classes) and increases the classification accuracy while minimizing the computation time for finding the best discretization intervals [24]. Few well known SD methods are Chi2 [25], Minimum Description Length Principle [26], Class Attribute Interdependence Maximization [27], Class-Attribute Contingency Coefficient [28], and ur-CAIM [29]. Popular dynamic discretizers are ID3 [30], ITFP [31]. It should be mentioned here that discretizing features dynamically when the feature selection is performed is expected to be advantageous. Apart from feature selection and discretization, a suitable search strategy is also necessary to find the best subset of features that optimizes a selection criteria [16]. An exhaustive search method guarantees to find the optimal solution, but it has to evaluate all possible combinations of all features which is often not feasible. Different search strategies have already been proposed such as forward selection (FS), backward elimination (BE), and Convex based Relaxation Approximation (COBRA) [16]. FS adds features in the selected subset one at a time until the value of the selection criteria cannot be improved. In the same manner, BE starts its selection with all the features and deletes features one at a time. The drawback of both FS and BE is that after selecting/deleting a feature, it cannot be deleted/re-selected later and therefore they might select redundant features [32]. Genetic algorithm can provide optimal feature subsets in many cases, however this method is still computationally impractical for large number of features [32]. On the other hand, greedy search strategies such as FS and BE seem to be particularly computationally advantageous and robust against over-fitting [33]. In [34], the authors propose an optimization framework called Spectral Conditional Mutual Information (SPEC_CMI) which formulates the mRMR feature selection criteria into a semi-definite programming model and solves it via spectral relaxation. It provides a global solution for MI based feature selection by overcoming the problems of greedy methods. However, these methods are not designed to be used for joint feature selection and discretization. Addressing these issues, Discretization and feature Selection based on Mutual information (DSM) [35] is proposed. It satisfies *Rrc* criteria and performs DD. However, the Incremental Gain Strategy proposed in DSM is defined in an adhoc manner. Also, the bias correction is not considered during the calculation of MI in DSM.

To solve these issues, this paper first proposes a method called Joint Bias corrected Mutual Information (JBMI) to reduce the bias and then proposes another method called modified Discretization and feature Selection based on Mutual information (mDSM) that uses JBMI for feature selection. The main contributions of this paper are as follows, first, the amount of error introduced for each part of (2) is derived theoretically (both for continuous and discrete case). It is also shown that relevancy, redundancy, and CI follow χ^2 distribution. Second, it addresses discretization and feature selection jointly with a single criteria (*Rrc*). The proposed discretization is dynamic and both the discretization and feature selection are independent of classification algorithms. Third, a χ^2 based search strategy is introduced to select a small number of features that are highly informative and discretizes these features with small number of intervals.

2. Feature selection criteria

A feature selection method has two major parts: a feature selection criterion and a feature search strategy. Different criteria have been introduced in literature such as distance, dependency, consistency and MI to assess the quality of a feature. Several methods have already been proposed based on these criteria such as RRF-SACO_1 [36] BQIGSA [37], RelaxMRMR [38], MRI [39], and JMI [21]. Among different selection criteria, MI is widely used because it can measure the non-linear dependencies between random variables as well as assess the "information content" of features in complex classification tasks [20]. Furthermore, MI as a formal set measure can assign a score to a feature set, which may not be possible by other measures [16]. Therefore, this paper considers different MI based feature selection criteria which are discussed in the following sub-section.

2.1. Evolution of MI based feature selection methods

Battiti [20] proposed an approximation of $I(S;C)$ (defined in (2)) under the assumption that the features are pairwise class-conditionally independent. This greedy selection algorithm is called MIFS. It adds the first feature to S by considering the highest relevance with C and the next feature is chosen that maximizes $[I(f_i;C) - \beta \sum_{f_s \in S} I(f_i; f_s)]$ where, β is a user defined parameter. However, (for $\beta = 1$) the r term ($I(f_i; f_s)$) grows in magnitude w.r.t. R term ($I(f_i; C)$) with the increase of number of selected features. Thus, R becomes insignificant compared to r and may select irrelevant features [40].

Peng et al. [18] proposed mRMR which partially overcomes the limitation of MIFS by setting the average number of the selected feature as β. This sort of approximation is also validated in [41] under the assumptions of lower-order dependency between the features. Note that, normalization of MI can reduce its bias towards the multivalued features. One such normalization is Normalized Mutual Information Feature Selection (NMIFS) [40] where the average of normalized MI is calculated for measuring the redundancy. The drawback of mRMR and NMIFS is that during the computation of redundancy they do not consider the class variable and thus may lose necessary information for classification.

Brown et al. [12] proposed a parameterized feature selection criteria from which they showed that well known MI based feature selection criteria such JMI and Conditional Mutual Information Maximization (CMIM) [42] can be derived. A unifying theoretical framework has also been established in [17] which along with Brown *et. al.* concludes that inclusion of *Rrc* property shows more satisfactory results.

All of the aforementioned methods use MI for selecting feature subset. Again, calculating MI requires the estimation of entropy of the feature using its underlying probability distributions. This distribution must be obtained from experimental observations. However, insufficient representation of the probability density function by the histogram of the observations may introduces bias in the computation of entropy [22,23]. The bias is a major problem since MI is calculated as the difference of two entropies and so even a very small amount of bias in entropy can have a large effect on MI. Thus, for better estimation of feature subset, this bias should be addressed in feature selection methods.

2.2. Joint bias corrected mutual information

In this section, the amount of bias is first derived using Theorems 1, 2, and Corollaries 1.1, 1.2, and 2.1. JBMI criterion for feature selection is then proposed based on these theorems and corollaries.

Theorem 1. *Bias is* $\frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln(2)}$ *for mutual information* ($I(X;Y)$) *between two continuous random variable X and Y, where* $\mathcal{I}$ *and* $\mathcal{K}$ *are the number of intervals in X and Y, respectively.*

Proof. MI can be expressed in terms of entropy using (3)

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{3}$$

here, $H(X)$ and $H(Y)$ denote the marginal entropies and $H(X, Y)$ denotes the joint entropy. For continuous distributions $H(X)$ can be expressed using $H(X) = -\int_{-\infty}^{\infty} f(x)\log f(x)dx$, where $f(x)$ is the *pdf* of X. In order to estimate it, let us divide X into $\mathcal{I}$ equal length (Δx) intervals. Let the number of samples observed in the i^{th} interval be n_i and the total number of samples be N. Thus the probability (p_i) of observing a sample in the i^{th} interval is $p_i = \int_{cell(i)} f(x)dx \approx f(x_i)\Delta x$. If $f(x)$ is approximately constant within an interval, a reasonable approximation of the entropy (employing discretization for the integral and putting $p_i = \frac{n_i}{N}$ where $i = 1$ to $\mathcal{I}$) is

$$\begin{aligned}\hat{H}(X) &= -\sum_i \left(\frac{n_i}{N}\log\frac{n_i}{N}\right) + \log(\Delta x)\\ &= -\frac{1}{\ln 2}\sum_i \left(\frac{n_i}{N}\ln\frac{n_i}{N}\right) + \log(\Delta x)\end{aligned} \tag{4}$$

Let us define $p = \frac{n_i}{N}$ and $q = \frac{\bar{n}_i}{N}$ and assume $f(p) = \frac{n_i}{N}\ln\frac{n_i}{N} = p\ln p$ as a function of p. Substituting these values in (4) and expanding $f(p)$ through Taylor Series at $p = q$, it is found that

$$\begin{aligned}f(p) &= f(q) + f^{(1)}(q)(p-q) + \frac{f^{(2)}(q)}{2!}(p-q)^2 + \dots\\ &= q\log q + (1+\log q)(p-q) + \frac{(p-q)^2}{2q} + \dots\end{aligned} \tag{5}$$

Substituting the values of p and q into (5) and replacing the formal parameters n_i by the stochastic variables $\underline{n}_i$, the expectation is taken assuming independent samples $\underline{n}_i$ is multinomially distributed.

$$\begin{aligned}E\{\underline{\hat{H}}(X)\} = \frac{1}{\ln 2}\sum_i \Bigg(&-\frac{\bar{n}_i}{N}\ln\frac{\bar{n}_i}{N} - \left(\frac{1}{N} + \frac{1}{N}\ln\frac{\bar{n}_i}{N}\right)E\{(\underline{n}_i - \bar{n}_i)\}\\ &- \frac{E\{(\underline{n}_i - \bar{n}_i)^2\}}{2N\bar{n}_i} + E\{R_i^3(\underline{n}_i)\}\Bigg) + \log(\Delta x)\end{aligned} \tag{6}$$

In (6), the 2nd term vanishes and the 3rd term can be written in the following form, represented in (8), using the value of expectations and variance from (7)

$$E\{\underline{n}_i\} = \bar{n}_i = Np_i; \quad VAR\{\underline{n}_i\} = Np_i(1-p_i) \tag{7}$$

$$\frac{1}{\ln 2}\sum_i \frac{E\{(\underline{n}_i - \bar{n}_i)^2\}}{2N\bar{n}_i} = \frac{1}{2N\ln 2}(\mathcal{I}-1) \tag{8}$$

Considering (6), (8) and ignoring the higher order terms, it can be written as

$$\begin{aligned}E\{\underline{\hat{H}}(X)\} &\approx \sum_i \left(-\frac{\bar{n}_i}{N}\log\frac{\bar{n}_i}{N}\right) - \frac{(\mathcal{I}-1)}{2N\ln 2}\\ &\quad + \log(\Delta x) \approx H(X) - \frac{(\mathcal{I}-1)}{2N\ln 2}\end{aligned} \tag{9}$$

Similarly, dividing Y into $\mathcal{K}$ intervals, $E\{\hat{H}(Y)\}$ can be written as

$$E\{\underline{\hat{H}}(Y)\} \approx H(Y) - \frac{(\mathcal{K}-1)}{2N\ln 2} \tag{10}$$

To calculate $E\{\underline{\hat{H}}(X,Y)\}$, let us consider a rectangular grid in the xy-plane and divide it into $(\mathcal{I}\times\mathcal{K})$ equal sized $(\Delta x \times \Delta y)$ cells with coordinates (i, k). Following the same process described before, the expected value of joint entropy is

$$E\{\underline{\hat{H}}(X,Y)\} = H(X,Y) - \frac{(\mathcal{I}\mathcal{K}-1)}{2N\ln 2} \tag{11}$$

substituting the value of $H(X)$, $H(Y)$ and $H(X, Y)$ into (3), it is found that

$$E\{\underline{\hat{I}}(X;Y)\} = I(X;Y) + \frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln 2} \tag{12}$$

where, $\frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln 2}$ is the bias for $I(X; Y)$ and should be corrected. □

Now let us consider X is a continuous random variable with $\mathcal{I}$ intervals and Y is the discrete class variable C with $\mathcal{K}$ classes. In terms of entropy, MI between X and C can be written as

$$I_1(X;C) = H(X) - H(X \mid C) = H(X) - \sum_{c=1}^{\mathcal{K}} p(c)H(X \mid C = c) \tag{13}$$

Using (9), assuming the entropy is a fixed value for a given X, it is natural to replace $E\{\underline{\hat{H}}(X)\}$ by $\hat{H}(X)$ and estimating $p(c)$ by $\frac{n_c}{N}$, where n_c is the number of elements in a particular class c, (13) can

be written in the following form

$$I_1(X;C) = \hat{H}(X) + \frac{(\mathcal{I}-1)}{2N\ln 2} - \sum_{c=1}^{\mathcal{K}} p(c)\left(\hat{H}(X \mid C=c) + \frac{(\mathcal{I}-1)}{2n_c \ln 2}\right)$$
$$= \hat{I}_1(X;C) + \frac{(\mathcal{I}-1)}{2N\ln 2} - \sum_{c=1}^{\mathcal{K}} \frac{n_c}{N}\frac{(\mathcal{I}-1)}{2n_c\ln 2}$$
$$= \hat{I}_1(X;C) - \frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln 2} \quad (14)$$

It can be observed from (12) and (14), that bias term for both $I_1(X;C)$ and $I(X;Y)$ are same. Hence, it can be concluded that

Corollary 1.1. *Bias is $\frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln(2)}$ for Relevancy ($I(f_i;C)$) between a feature f_i and class variable C, where $\mathcal{I}$ is the number of intervals in feature f_i and $\mathcal{K}$ is the number of classes in C.*

Corollary 1.2. *From Theorem 1 it can be readily observed that Bias is $\frac{(\mathcal{I}-1)(\mathcal{J}-1)}{2N\ln(2)}$ for Redundancy ($I(f_i;f_j)$) between two features f_i and f_j, where $\mathcal{I}$ and $\mathcal{J}$ are the number of intervals in feature f_i and f_j, respectively.*

Theorem 2. *Bias is $\frac{(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}}{2N\ln(2)}$ for Conditional Mutual Information ($I(X;Y|Z)$) where X, Y, and Z are continuous random variables and $\mathcal{I}$, $\mathcal{J}$ and $\mathcal{K}$ are the number of intervals in X, Y, and Z, respectively.*

Proof. Conditional MI, $I(X;Y|Z)$ can be represented as $H(X,Z)+H(Y,Z)-H(X,Y,Z)-H(Z)$. For continuous distribution $H(X,Y,Z)$ can be defined as $H(X,Y,Z) = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y,z)\log f(x,y,z)dxdydz$.

Assume the xyz-space is divided into $(\mathcal{I}\times\mathcal{J}\times\mathcal{K})$ equally sized $(\Delta x\times\Delta y\times\Delta z)$ cells with coordinates (i,j,k). The number of samples observed in cell (i,j,k) is n_{ijk} and the total number of samples is N. Then, the estimator function of entropy $H(X,Y,Z)$ will be

$$\hat{H}(X,Y,Z) = -\frac{1}{\ln 2}\sum_{i,j,k}\left(\frac{n_{ijk}}{N}\ln\frac{n_{ijk}}{N}\right) + \log(\Delta x\Delta y\Delta z) \quad (15)$$

Let us define $p=\frac{n_{ijk}}{N}$ and $q=\frac{\overline{n}_{ijk}}{N}$ and assume $f(p)=\frac{n_{ijk}}{N}\ln\frac{n_{ijk}}{N} = p\ln p$ as a function of p. Substituting these value in (15) and expanding $f(p)$ through Taylor series at $p=q$, it is found

$$\hat{H}(X,Y,Z) = \frac{1}{\ln 2}\sum_{i,j,k}\left(-\frac{\overline{n}_{ijk}}{N}\ln\frac{\overline{n}_{ijk}}{N} - \left(\frac{1}{N}+\frac{1}{N}\ln\frac{\overline{n}_{ijk}}{N}\right)(n_{ijk}-\overline{n}_{ijk}) - \frac{(n_{ijk}-\overline{n}_{ijk})^2}{2N\overline{n}_{ijk}} + R^3_{ijk}(n_{ijk})\right) + \log(\Delta x\Delta y\Delta z) \quad (16)$$

Replacing n_{ijk} by the stochastic variables $\underline{n}_{ijk}$, taking the expectations of $\hat{H}(X,Y,Z)$, and assuming $\underline{n}_{ijk}$ is multinomially distributed, it is found that

$$E\{\hat{H}(X,Y,Z)\} = \frac{1}{\ln 2}\sum_{i,j,k}\left(-\frac{\overline{n}_{ijk}}{N}\ln\frac{\overline{n}_{ijk}}{N} - \left(\frac{1}{N}+\frac{1}{N}\ln\frac{\overline{n}_{ijk}}{N}\right) E\{(\underline{n}_{ijk}-\overline{n}_{ijk})\} - \frac{E\{(\underline{n}_{ijk}-\overline{n}_{ijk})^2\}}{2N\overline{n}_{ijk}} + E\{R^3_{ijk}(\underline{n}_{ijk})\}\right) + \log(\Delta x\Delta y\Delta z) \quad (17)$$

In (17), the 2nd term vanishes and the 3rd term can be written in the form represented in (19) using the value of expectations and variance from (18)

$$E\{\underline{n}_{ijk}\} = \bar{n}_{ijk} = Np_{ijk}; \quad VAR\{\underline{n}_{ijk}\} = Np_{ijk}(1-p_{ijk}) \quad (18)$$

$$\frac{1}{\ln 2}\sum_i\sum_j\sum_k \frac{E\{(\underline{n}_{ijk}-\overline{n}_{ijk})^2\}}{2N\overline{n}_{ijk}} = \frac{(\mathcal{I}\mathcal{J}\mathcal{K}-1)}{2N\ln 2} \quad (19)$$

Considering (17), (19) and ignoring the higher order terms, it is found

$$E\{\hat{\underline{H}}(X,Y,Z)\} \approx \sum_{i,j,k}\left(-\frac{\overline{n}_{ijk}}{N}\log\frac{\overline{n}_{ijk}}{N}\right) - \frac{(\mathcal{I}\mathcal{J}\mathcal{K}-1)}{2N\ln 2} + \log(\Delta x\Delta y\Delta z)$$
$$\approx H(X,Y,Z) - \frac{(\mathcal{I}\mathcal{J}\mathcal{K}-1)}{2N\ln 2} \quad (20)$$

Using (9), (11) and (20) conditional MI can be expressed as

$$I(X;Y\mid Z) = \hat{I}(X;Y\mid Z) - \frac{(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}}{2N\ln 2} \quad (21)$$

□

Let X and Y are continuous and Z is the discrete class variable C with $\mathcal{K}$ classes. Following (14), using (9), (11) and estimating $p(c)$ by $\frac{n_c}{N}$ where n_c is the number of elements in class c, bias of the conditional MI can be derived as

$$I_1(X;Y\mid C) = H(X\mid C)+H(Y\mid C)-H(X,Y\mid C)$$
$$= \hat{I}_1(X;Y\mid C) - \frac{\mathcal{K}(\mathcal{I}-1)(\mathcal{J}-1)}{2N\ln 2} \quad (22)$$

From (22), it can be concluded that

Corollary 2.1. *Bias is $\frac{(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}}{2N\ln(2)}$ for CI ($I(f_i;f_j|C)$) between feature f_i and f_j given class C, where $\mathcal{I}$ and $\mathcal{J}$ are the number of intervals in feature f_i and f_j respectively and $\mathcal{K}$ is the number of classes in C*

Adjusting all the bias terms, (2) can be written as

$$I(f_i;C) - \frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln 2} + \frac{1}{|S|}\sum_{f_s\in S}\left(I(f_i;f_s\mid C) - \frac{(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}}{2N\ln 2} - I(f_i;f_s) + \frac{(\mathcal{I}-1)(\mathcal{J}-1)}{2N\ln 2}\right) \quad (23)$$

Eq. (23) will help to give a more accurate measure for joint MI ($I(S;C)$). Similar proof of Theorem 1 can be found in [22] and [43]. To the best of our knowledge, this is the first work that derives the bias term for CI and computes joint MI using (23) for feature selection (with bias correction for relevancy, redundancy, and CI). Since (23) considers the bias of all the three terms, it is named Joint Bias corrected Mutual Information (JBMI) and the proposed method selects the set of features that maximizes it.

3. Search strategy for feature selection

Apart from the different feature selection criteria, a search strategy is also important to select the feature subset. Several searching mechanism can be found in the literature such as FS, BS, and COBRA. These methods only search for the optimal set of features but the objective of this paper is to search for both feature subset and the appropriate discretization level for each selected feature. For this purpose, it is first shown that MI follows χ^2 distribution and then the χ^2 test is used for both feature selection and discretization.

3.1. Mutual information follows χ^2 distribution

In this subsection, it is first shown that Relevancy, Redundancy, and CI individually follow χ^2 distribution. The proof of Theorem 3 is given in [44]. Similarly, Theorem 4 can also be proved.

Theorem 3. *Relevancy ($I(f_i;C)$) follows χ^2 distribution with $(\mathcal{I}-1)(\mathcal{K}-1)$ degrees of freedom if f_i and C are statistically independent.*

Theorem 4. *Redundancy ($I(f_i;f_j)$) follows χ^2 distribution with $(\mathcal{I}-1)(\mathcal{J}-1)$ degrees of freedom if f_i and f_j are statistically independent.*

Following the proof of Theorems 3 and 4, the critical value for $\chi^2_C(R)$ and $\chi^2_C(r)$ can be calculated using (24), where, N is the total number of samples.

$$\chi^2_C(R) = 2N\ln(2) \times I(f_i;C); \quad \chi^2_C(r) = 2N\ln(2) \times I(f_i;f_j) \tag{24}$$

Theorem 5. *CI ($I(f_i; f_j|C)$) follows χ^2 distribution with $(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}$ degrees of freedom, if f_i and f_j are statistically independent given C.*

Proof. Conditional MI between two independent random variables X and Y for a given Z can be defined as

$$\begin{aligned} I(X;Y \mid Z) &= \sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z} p(x,y,z)\log\frac{p(x,y \mid z)}{p(x \mid z)p(y \mid z)} \\ &= \frac{1}{\ln 2}\sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z} p(x,y,z)\ln\frac{p(x,y,z)}{p(x \mid z)p(y \mid z)p(z)} \end{aligned} \tag{25}$$

here, $p(x, y, z)$ is the joint probability and $p(x|z)$, $p(y|z)$ are the conditional probabilities. Let $q(x, y, z) \equiv p(x|z)p(y|z)p(z)$. For simplicity, assume $p_1 = p(x, y, z), q_1 = q(x, y, z)$ and $f(p_1) = p(x, y, z)\ln\frac{p(x,y,z)}{q(x,y,z)} = p_1\ln\frac{p_1}{q_1}$ as a function of p_1. Expanding $f(p_1)$ into a Taylor series at $p_1 = q_1$ yields

$$f(p_1) = f(q_1) + f^{(1)}(q_1)(p_1-q_1) + \frac{f^{(2)}(q_1)}{2!}(p_1-q_1)^2 + \dots \tag{26}$$

where, $f^r(p_1)$ is the r^{th} derivative of $f(p_1)$. The first term vanishes, the second term is $(p_1 - q_1)$ and the third term is $\frac{(p_1-q_1)^2}{2q_1}$. Ignoring the higher order terms

$$I(X;Y \mid Z) \approx \frac{1}{\ln 2}\sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z}\frac{(p(x,y,z)-q(x,y,z))^2}{2q(x,y,z)} \tag{27}$$

Replacing $p(x,y,z) = \frac{n_{ijk}}{N}$, $p(x \mid z) = \frac{p(x,z)}{p(z)} = \frac{n_{i\cdot k}}{n_{\cdot\cdot k}}$, $p(y \mid z) = \frac{p(y,z)}{p(z)} = \frac{n_{\cdot jk}}{n_{\cdot\cdot k}}$, $p(z) = \frac{n_{\cdot\cdot k}}{N}$, (27) can be written as

$$I(X;Y \mid Z) \approx \frac{1}{2\ln 2}\sum_i\sum_j\sum_k \frac{\left(\frac{n_{ijk}}{N} - \frac{n_{i\cdot k}}{n_{\cdot\cdot k}}\frac{n_{\cdot jk}}{N}\right)^2}{\frac{n_{i\cdot k}}{n_{\cdot\cdot k}}\frac{n_{\cdot jk}}{N}} \tag{28}$$

This expression of conditional MI is related to a standard χ^2 independence test statistic with $(|X|-1)(|Y|-1)|Z|$ degrees of freedom [45]. Hence, $I(f_i; f_j|C)$ follows χ^2 distribution with $(\mathcal{I}-1)(\mathcal{J}-1)K$ degrees of freedom. □

Therefore, CI follows χ^2 distribution and $\chi^2_C(c)$ can be calculated using (29)

$$\chi^2_C(c) = 2N\ln(2) \times I(f_i; f_j \mid C) \tag{29}$$

Theorems 3–5 state that each individual part of (2) follows χ^2 distribution. Comparing (23) and the respective critical values from (24) and (29), one can take decision regarding feature selection and discretization.

4. Modified discretization and feature selection

The modified Discretization and feature Selection based on Mutual information (mDSM) is mainly based on Theorems 1–5 and Corollaries 1.1, 1.2 and 2.1. It has two major parts which are described below.

4.1. Candidate feature selection and discretization

mDSM first finds the discretization level of each feature f_i individually based on its relevancy by calculating its MI with C using (30).

$$J_{rel}(f_i) = I(f_i^{d_i}; C) \tag{30}$$

here, $f_i^{d_i}$ denotes feature f_i with d_i discretization levels. Since for each f_i finding the minimum number of discretization levels is desired, mDSM uses the χ^2 test to select the minimum d_i for which $J_{rel}(f_i)$ is greater than the critical value $\chi^2_C(R)$ with degrees of freedom as mentioned in Theorem 3. Because it implies that f_i is statistically dependent on class C (for discretization levels less than or equal to a maximum value max_d). Here, max_d can be determined by the total number of distinct values in each f_i and therefore, the value of max_d may be different for different f_is. However, instead of setting separate values for each f_i, one can empirically find a single max_d so that desired discretization level of each f_i (if exists) can be achieved within that max_d. In this paper, use of a single max_d for all features is suggested. This process of selecting the features and settings the discretization levels is described in Algorithm 1.

Algorithm 1 Relevance based Candidate Feature Selection and Discretization.

Input: Set of n features, $F = \{f_1, f_2, \dots, f_n\}$, Class C, max_d
Output: Set of r candidate features F_c with discretization levels $D_c = \{d_1, d_2, \dots, d_r\}$, and relevancy $J_c = \{J_1, J_2, \dots, J_r\}$

```
F_c ← ∅
for each f_i ∈ F do
    for all j = 2 to max_d do
        Discretize f_i with j intervals
        if J_rel(f_i) > χ²_C(R) then
            D_c ← D_c ∪ j ; J_c ← J_c ∪ J_rel(f_i) ; F_c ← F_c ∪ f_i
            break
        end if
    end for
end for
Return F_c with its D_c and J_c
```

4.2. Final feature selection and discretization

This step assumes that candidate features are sorted in descending order according to their relevancy. First, the feature with the highest relevancy is selected. Here, the objective is to make the selected feature and new feature (next ranked) jointly more relevant (using J_{mDSM} in (31)) with the class variable C. To achieve this, DD is applied by shifting the discretization level (d_i) of the new feature by a small amount ($\pm\delta$), since large shift may cause the feature to become irrelevant. According to Algorithm 3, the discretization level of the first selected feature (i.e., d_1) never changes. However, it is observed that when the next ranked feature is considered for selection, shifting of d_1 by a small amount may lead to a higher value of J_{mDSM} (31). Therefore, to capture more information the discretization level of the first selected feature is also shifted. This strategy is followed throughout the experiments unless stated otherwise.

$$\begin{aligned} J_{mDSM}(f_i) = I(f_i^{d_j}; C) - \frac{(\mathcal{I}-1)(\mathcal{K}-1)}{2N\ln 2} + \frac{1}{|S|}\sum_{f_s\in S}\Bigg(I(f_i^{d_j}; f_s \mid C) \\ - \frac{(\mathcal{I}-1)(\mathcal{J}-1)\mathcal{K}}{2N\ln 2} - I(f_i^{d_j}; f_s) + \frac{(\mathcal{I}-1)(\mathcal{J}-1)}{2N\ln 2}\Bigg) \end{aligned} \tag{31}$$

here, a feature is selected if its score (J_{mDSM}) is larger than the χ^2 critical value (χ^2_C(Rrc)). Since each of three parts of (31) individually follows χ^2 distribution, mDSM checks whether each of the parts is significant or not. A feature is selected if any part of (31) is significant, except redundancy. If a feature is significantly redundant, it is always discarded. Following this process, Algorithm 3 produces the final subset of features with their appropriate discretization levels.

Table 1
Sample dataset for the example.

f_1	1	17	18	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$f_1(4)$	1	4	4	1	1	1	1	2	2	2	2	2	3	3	3	3	4	4	4	4
$f_1(8)$	1	7	8	1	2	2	2	3	3	4	4	4	5	5	6	6	7	8	8	8
f_2	23	19	10	3	50	24	10	2	30	20	18	25	38	7	20	29	4	3	27	3
f_3	2	4	6	8	28	30	14	16	18	20	12	14	26	28	30	32	34	36	38	40
f_4	19	15	20	17	14	5	1	6	14	18	20	19	14	18	3	8	19	5	4	7
$f_4(2)$	2	2	2	2	2	1	1	1	2	2	2	2	2	2	1	1	2	1	1	1
$f_4(3)$	3	2	3	2	2	1	1	1	2	3	3	3	2	3	1	2	3	1	1	2
f_5	8	18	19	16	30	21	24	23	22	20	19	12	13	16	19	20	10	22	22	25
C	1	1	1	1	2	2	2	2	2	1	1	1	1	1	2	2	2	2	2	2

Algorithm 2 χ^2 based Dynamic Discretization.

Input: Feature f_i with its discretization level d_i
Output: J_i, Threshold T, and d_i
for all $j = d_i - \delta$ to $d_i + \delta$ **do**
  Discretize f_i with j intervals
  if $J_{mDSM}(f_i) > \chi^2_C(Rrc)$ **then**
    $d_i \leftarrow j$; $J_i \leftarrow J_{mDSM}(f_i)$; $T \leftarrow \chi^2_C(Rrc)$
  end if
end for
Return J_i, T, and d_i

Algorithm 3 mDSM.

Input: Set of features F, maximum discretization level max_d
Output: Selected subset of k features, S with discretization levels $D = \{d_1, d_2, \ldots, d_k\}$
Select r candidate feature set F_c, its corresponding discretization level D_c and relevance J_c using Algorithm 1
Sort F_c and D_c in decreasing order based on their corresponding J_c values
Select f_1 and the corresponding d_1 from F_c with max J_c
$S \leftarrow f_1$; $D \leftarrow d_1$; $F_c \leftarrow F_c \setminus f_1$
for all $i = 2$ to r **do**
  Calculate J_{mDSM}, T and d_i for f_i using Algorithm 2
  if $J_{mDSM} > T$ **then**
    $S \leftarrow S \cup f_i$; $D \leftarrow D \cup d_i$
  end if
  $F_c \leftarrow F_c \setminus f_i$
end for
Return S and their respective D

4.3. An illustrative example

Let us consider a dataset having five features ($f_1, \ldots, f_5$ in Table 1) with two classes. The steps followed by the mDSM for this dataset is described below. Here, the $f_i(d)$ is used to denote a feature f_i with d discretization levels.

Firstly, when Algorithm 1 is applied, each feature is discretized and the level of discretization is set in a way so that the relevancy of f_i with C becomes significant with minimum number of intervals. In this dataset it is observed that, when relevancy is computed without bias correction, only f_2 is discarded (fails the χ^2 test) and the rest of the features $f_1(8)$, $f_3(3)$, $f_4(2)$, and $f_5(2)$ are selected as candidates. In this case, $f_1(8)$ contains more information (and thus increased MI) compared to say, $f_1(4)$ and passes the χ^2 test. However, if bias correction is done during relevancy computation, both f_1 and f_2 are discarded. Therefore, the resulting candidate set contains $f_3(3)$, $f_4(2)$, and $f_5(2)$.

In case of SD, Algorithm 2 is not called during the use of Algorithm 3 and the candidate features will have the same discretization level as set by Algorithm 1. Without bias correction, in static case only f_1 is chosen from the candidate set and produces 33% accuracy with 10 fold Cross Validation (10-CV) using Support Vector Machine (SVM). On the other hand, feature $f_3(3)$ and $f_4(2)$ are selected and f_5 is discarded, if bias is incorporated. In this case, the accuracy is improved to 70%.

In case of DD (without bias correction) Algorithm 2 is applied and feature $f_1(8)$ and $f_4(3)$ are selected. Here, the interval of f_4 is changed from 2 to 3 considering its dependency with the other features and thus f_4 provides some extra information about the class. These two selected features now jointly obtain 66% accuracy (10-CV). Finally, if the bias correction and DD are performed, the selected subset includes $f_3(3)$, $f_4(2)$ and $f_5(3)$. Although f_5 is discarded in SD, it gains CI after changing the interval dynamically which is also understandable from (31) (i.e., its J_{mDSM} value increases from 0.135 to 0.291). This increases the accuracy from 70% to 90% and signifies the importance of feature selection with DD and bias correction.

4.4. Complexity analysis

Suppose a dataset has N features and M samples. In mDSM, the computation of MI and conditional MI takes $O(M)$ steps [38]. Also, the Equal Frequency (EF) discretization (or Equal Width (EW) discretization) performed in Algorithm 1 requires $O(M)$ steps. Since the inner loop in Algorithm 1 is iterated a constant number of times, this task of selecting candidate feature set requires a total of $O(NM)$ steps. Similarly, in Algorithm 3, J_{mDSM} value computation and selection of features that pass the χ^2 critical value has complexity of $O(N^2M)$. Moreover, constant time is required to adjust the discretization level of a feature dynamically and therefore, it is omitted from the complexity of mDSM. Hence, total time complexity of mDSM is $O(N^2M)$.

The complexity of mRMR is $O(N^3M)$ [38]. Other similar methods such as JMI and MIFS have the same complexity. Unlike these greedy algorithms, main cost of SPEC_CMI is computing a $N \times N$ matrix of relevancy and pairwise conditional MI. This takes $O(N^2M)$ steps. The ranking of features then takes an additional $O(N^2)$ steps. COBRA sets up a semidefinite programming problem for global feature selection. It computes a $N \times N$ matrix of MI which has similar complexity of $O(N^2M)$ [16]. For selecting a set of k features it solves the semidefinite program using the interior point algorithm that takes $O(N^4)$ time [46]. It should be noted that the complexity of these methods (that use mRMR or similar criteria) can be improved if pairwise MI and conditional MI are cached into a $N \times N$ table for reuse [38]. In that case, their complexity will be $O(N^2M)$. Similar caching mechanism can be applied for mDSM.

Table 2
Dataset Overview and Accuracy of Existing Feature Selection Methods.

Dataset	Features	Instance	Class	MIFS	MIM	mRMR	CMIM	JMI
Low Dimensional (< 100 features) and Large Sample (≥ 100 samples) (10-CV)								
Iris	4	150	3	**0.933**	**0.933**	**0.933**	**0.933**	**0.933**
Liver	7	345	2	**0.571**	**0.571**	**0.571**	**0.571**	**0.571**
Pima	8	768	2	0.727	0.727	0.727	**0.732**	**0.732**
Yeast	8	1484	10	0.465	**0.506**	**0.506**	**0.506**	**0.506**
Glass	9	214	6	0.483	**0.544**	0.526	0.535	0.535
Breast Tissue	10	106	6	**0.586**	**0.586**	0.571	0.571	0.571
Wine	13	178	3	0.937	0.932	0.953	**0.968**	0.931
Heart	13	270	2	0.811	**0.826**	**0.826**	0.822	**0.826**
Australian	14	690	2	**0.876**	0.871	**0.876**	**0.876**	0.871
Segment	19	2310	7	0.878	0.871	**0.891**	0.887	**0.891**
German	20	1000	2	0.726	0.751	0.742	0.748	**0.755**
Cardio	21	2126	10	0.641	0.671	**0.678**	0.669	**0.678**
Waveform	21	5000	3	0.744	0.799	**0.820**	**0.820**	**0.820**
Parkinson	22	197	2	**0.865**	0.840	0.840	0.835	0.845
Steel	27	1941	7	0.672	0.650	0.670	**0.700**	0.653
Breast	30	569	2	0.947	0.960	0.959	**0.966**	0.960
Ionosphere	33	351	2	**0.861**	0.836	0.839	0.817	0.818
Dermatology	34	366	6	0.945	0.865	**0.963**	0.945	**0.963**
Spambase	57	4601	2	0.745	0.746	0.748	**0.767**	0.745
Sonar	60	208	2	0.747	0.723	0.700	0.723	**0.755**
Libras	91	360	15	0.747	0.729	0.753	**0.769**	0.760
Low Dimensional Feature and Small Sample (LOO)								
Lung Cancer	56	32	3	**0.592**	0.444	0.481	0.518	0.519
High Dimensional and Large Sample (10-CV)								
Musk	166	476	2	0.773	0.792	0.756	0.785	**0.794**
Semeion	256	1593	10	**0.855**	0.735	0.751	0.847	0.743
Arrhythmia	279	452	16	0.647	0.746	0.760	**0.770**	0.740
Madelon	500	2600	2	0.551	0.578	0.582	0.583	**0.585**
Lung	3312	203	5	0.895	0.914	0.936	**0.959**	0.941
High Dimensional Feature and Small Sample (LOO)								
Colon	2000	62	2	0.740	0.710	0.774	0.758	**0.806**
Lymphoma	4026	96	9	0.830	0.800	0.860	0.885	**0.920**
DBW(sub)	4702	64	2	0.813	**0.906**	**0.906**	0.875	0.875
Win/Tie/Loss				19/3/8	17/6/7	13/9/8	13/8/9	-

5. Result analysis and discussions

In this section, the experimental setup of different methods along with the proposed ones and their evaluation process are presented. Furthermore, a number of experiments are performed to highlight the effectiveness of the proposed contributions step by step. Finally, mDSM is compared with other state-of-the-art MI based methods.

5.1. Dataset description

In this work, 30 benchmark datasets from UCI Machine Learning Repository [47] and Feature Selection Repository of Arizona State University [48] are used. These datasets are chosen as they are used in [38] and [8] for similar purposes. The characteristics of the datasets are presented in columns 2–4 of Table 2. For the ease of presentation and discussion, all the datasets are categorized into Low Dimensional Datasets and High Dimensional Datasets. Moreover, each of these categories is further divided into two parts: small sample data and large sample data.

5.2. Implementation details

In this paper, SVM (linear kernel) is used to classify the datasets. Beside this, KNN and XGBoost[1] [49] are also used to generate the results of Table 6. 10-CV is performed for the datasets with large number of samples and Leave-One-Out (LOO) cross-validation is conducted otherwise. For fair comparison, the same strategy is followed for all other methods used in this paper. Following the protocol discussed in [38], five (pre-defined) EW discretization is used for all of the existing feature selection methods (unless stated otherwise) and FS is performed to select the subset of features. In order to evaluate the feature selection methods (for Table 2), half of the given features are considered if the number of features in a dataset is less than 50, otherwise the top 50 features are chosen following the experimental setup mentioned in [38]. Apart from FS, χ^2 based selection and COBRA[2] that automatically select features (number of selected features are given within parenthesis in the tables) are used to generate the results.

5.3. Comparison of existing MI based feature selection methods

A feature selection criterion is required to validate the proposed contributions and thus it is preferable to find the best performing existing feature selection criterion and then use it with the three major contributions of this paper. Hence, in this section, the performances of the existing state-of-the-art MI based feature selection methods, namely CMIM[3], MIM[3], MIFS[3], mRMR[3], and JMI[3] are analyzed first. Similar experiments are also performed by researchers [12]. However, in this paper, more rigorous experiments are conducted with large number of datasets. More specifically, high dimensional datasets that have both small and large number of samples are used in this analysis. Table 2 presents the average accuracies of these selection methods for each dataset. In this table,

[1] https://github.com/dmlc/xgboost.

[2] https://github.com/tofigh-/COBRA.

[3] https://www.mathworks.com/matlabcentral/fileexchange/26981-feature-selection-based-on-interaction-information.

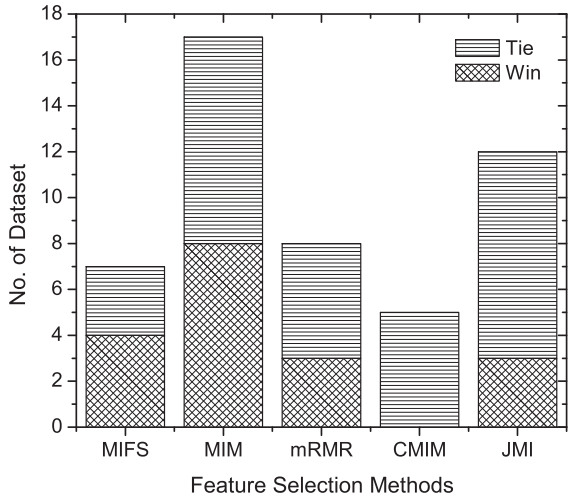

Fig. 1. Performance of existing feature selection methods based on IS.

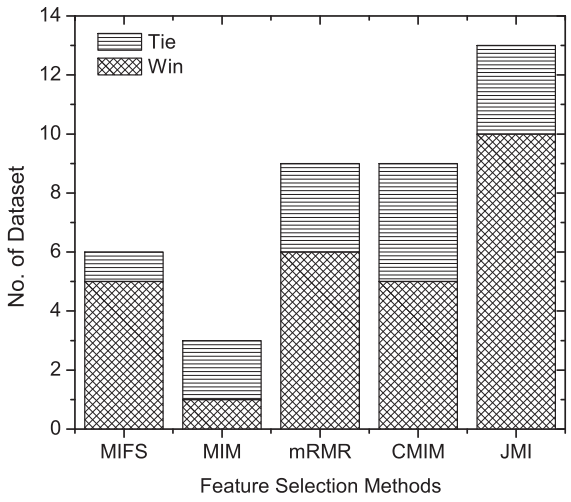

Fig. 2. Performance of existing feature selection methods based on Score.

Table 3
Ranking of existing feature selection criteria.

Frequency in Pareto Optimal set				
JMI (15)	MIM (13)	mRMR (10)	MIFS (10)	CMIM (9)

Win/Tie/Loss indicates the number of datasets for which JMI performs better/equally-well/worse than other aforementioned methods. It reveals that the accuracy of JMI is comparatively better than other methods. MIM performs worst in most of the cases, especially for the datasets with large number of features. For example, the accuracy of MIM is always lower in comparison with other methods except for few datasets. This is because, MIM is capable of performing well by selecting features based on only relevancy for small number of features, where the relationship among the features is not as complicated as in the high-dimensional datasets [38]. On the other hand, for large number of features, in addition to relevancy, JMI considers redundancy and CI and thus provides higher accuracy for classification. Apart from JMI, mRMR and CMIM perform considerably well. In case of MIFS ($\beta = 1$), it does not have good balancing between redundancy and relevancy and thus, the accuracy is lower compared to mRMR and CMIM.

It is observed that, even though JMI performs better in most of the cases, the differences in accuracies are not that high. However, accuracy is not the only evaluation metric to judge the quality of a method; stability is often used along with accuracy [12]. Information stability (IS) [50] is one of such evaluation metric which measures the reliability of a selection method. Fig. 1 compares existing methods based on their IS. Here number of wins ($=w$) for a particular method indicates that this is the best performing method compared to other methods for w datasets and ties indicates the number of datasets for which a particular method does not win independently, but is one of the best performing methods. It is found that MIM is the most stable method as it produces almost same set of features that have similar information in each fold of 10-CV. Note that the stability of MIM is not satisfactory for high dimensional datasets. Although the stability of JMI is not as satisfactory as that of MIM, it is more stable than MIFS, mRMR and CMIM.

To make a trade-off between accuracy and IS, Pareto Optimality (PO) is used here which returns a set of non-dominant candidate solutions. The feature selection methods that appear in this set are superior to other methods that are not included in this set (at least in one criterion). Table 3 presents the ranking of the feature selection methods based on their frequency in the PO set. It shows that JMI holds the top rank which indicates that JMI has better performance considering a trade-off between accuracies and IS. Although, MIM has the second position in ranking (for its higher stability), it cannot be regarded as a good selection method as the selected features contain redundant features which results in poor classification accuracy (see Table 2). Apart from PO computed from the two criteria, a new metric namely Score of the selection methods (which also demonstrates a good balance between stability and accuracy) is suggested here and is defined as the weighted average of the two metrics. In this paper, equal weight is employed for the metrics. Fig. 2 highlights the performance of the selection methods based on their Score values.

Finally, from the aforementioned discussion, it can be easily concluded that JMI (that uses *Rrc* criteria) performs comparatively better. It also provides a good trade-off between stability and accuracy which is asserted by the research community [12,17] as well. This paper, therefore, incorporates JMI as a base feature selection method and uses it to demonstrate the effectiveness of the proposed methods.

5.4. Impact of bias correction, χ^2 based search, and discretization

mDSM has three contributing parts: JBMI, DD, and χ^2 search strategy. To understand their impact separately, each of these three parts is incorporated in Table 4 step by step. So far, PO and Score are calculated based on accuracy and IS. To incorporate the number of features selected by the methods in the computation of PO and Score, another metric namely f_score is defined in (32).

$$f\_score = \frac{N_t - N_s}{N_t} \tag{32}$$

here, N_t is the total number of features and N_s is the number of selected features. The Score is now computed as the equally weighted average of three metrics, namely f_score, accuracy, and IS.

5.4.1. Impact of bias correction

To demonstrate the effect of bias, JMI and JBMI are first compared in terms of accuracies. In Table 4, JC and JBC represent JMI and JBMI with COBRA search respectively. Comparing JC and JBC, it is observed that accuracies are improved and the f_score is decreased due to the incorporation of bias for most of the datasets. JBC wins for 50% of the datasets and for 17% datasets the accuracy remains same compared to JC.

For further validation, JChi and JBChi are examined that represent JMI and JBMI with proposed χ^2 search respectively. JBChi

Table 4
Classification accuracy of different methods.

Dataset	JC	JBC	JChi	JBChi	JBDC	JBDChi
Low Dimensional Feature and Large Sample (10-CV)						
Iris	0.913(2)	0.927(1)	0.940(2)	0.940(2)	0.893(1)	0.920(2)
Liver	0.594(6)	0.571(3)	0.566(3)	0.571(2)	0.563(1)	0.594(2)
Pima	0.736(8)	0.740(3)	0.730(5)	0.719(3)	0.730(3)	0.726(2)
Yeast	0.500(7)	0.495(7)	0.506(6)	0.475(4)	0.497(6)	0.420(2)
Glass	0.517(7)	0.522(5)	0.548(7)	0.539(5)	0.552(6)	0.552(2)
Breast Tissue	0.550(5)	0.571(4)	0.564(6)	0.569(3)	0.700(5)	0.643(2)
Wine	0.916(9)	0.916(5)	0.953(12)	0.942(8)	0.905(8)	0.905(2)
Heart	0.811(10)	0.822(6)	0.811(9)	0.815(7)	0.811(5)	0.804(4)
Australian	0.683(11)	0.876(4)	0.876(6)	0.876(5)	0.876(4)	0.876(4)
Segment	0.888(12)	0.844(11)	0.890(15)	0.890(15)	0.945(8)	0.944(13)
German	0.762(20)	0.746(6)	0.758(10)	0.741(5)	0.739(4)	0.707(5)
Cardio	0.633(13)	0.675(11)	0.702(20)	0.692(18)	0.675(13)	0.789(8)
Waveform	0.808(13)	0.809(9)	0.840(19)	0.840(19)	0.833(9)	0.863(15)
Parkinson	0.845(14)	0.835(9)	0.825(14)	0.845(7)	0.840(9)	0.850(3)
Steel	0.696(20)	0.696(18)	0.709(24)	0.710(23)	0.655(15)	0.602(10)
Breast	0.957(20)	0.945(12)	0.960(25)	0.960(21)	0.934(11)	0.938(2)
Ionosphere	0.850(33)	0.844(28)	0.850(33)	0.850(33)	0.839(16)	0.817(2)
Dermatology	0.953(23)	0.965(18)	0.965(32)	0.965(28)	0.945(21)	0.943(29)
Spambase	0.733(41)	0.727(20)	0.691(6)	0.678(4)	0.910(25)	0.938(48)
Sonar	0.727(60)	0.727(24)	0.741(37)	0.741(17)	0.736(12)	0.759(9)
Libras	0.767(90)	0.773(90)	0.773(90)	0.775(44)	0.751(90)	0.747(79)
Low Dimensional Feature and Small Sample (LOO)						
Lung Cancer	0.444(49)	0.556(26)	0.556(20)	0.481(11)	0.556(6)	0.593(5)
High Dimensional Feature and Large Sample (10-CV)						
Musk	0.817(165)	0.796(91)	0.788(145)	0.798(94)	0.765(70)	0.702(25)
Semeion	0.932(254)	0.932(254)	0.932(256)	0.932(256)	0.939(253)	0.939(255)
Arrhythmia	0.695(255)	0.726(100)	0.709(102)	0.783(36)	0.698(43)	0.677(17)
Madelon	0.543(500)	0.548(176)	0.547(233)	0.597(41)	0.611(10)	0.611(11)
Lung	0.945(3294)	0.959(1466)	0.945(2122)	0.948(437)	0.969(1412)	0.950(1400)
High Dimensional Feature and Small Sample (LOO)						
Colon	0.855(1987)	0.855(722)	0.806(872)	0.806(468)	0.661(64)	0.774(86)
Lymphoma	0.958(4026)	0.969(3905)	0.964(3080)	0.967(967)	0.969(2064)	0.906(930)
DBWorld(sub)	0.844(96)	0.828(73)	0.859(20)	0.797(24)	0.781(6)	0.891(6)
Win/Tie/Loss	15/5/10	-	11/10/9	-	-	-

performs better than JChi for 37% datasets and their performances are equivalent for one third of the datasets (see Table 4). It also demonstrates the superiority of JBMI compared to JMI. Comparison between JC vs JBC and JChi vs JBChi shows that the improvement in accuracy and stability (see Figs. 3 and 4) due to the impact of bias is not that prominent compared to the improvement in Score or the frequency in the PO set (shown in Table 5). The major differences are found in the number of selected features which increases Score and the frequency in PO set of JBC and JBChi compared to JC and JChi respectively. The superiority of JBC and JBChi thus can be attributed to the incorporation of bias which helps to estimate more accurate joint MI and asserts the strength of JBMI.

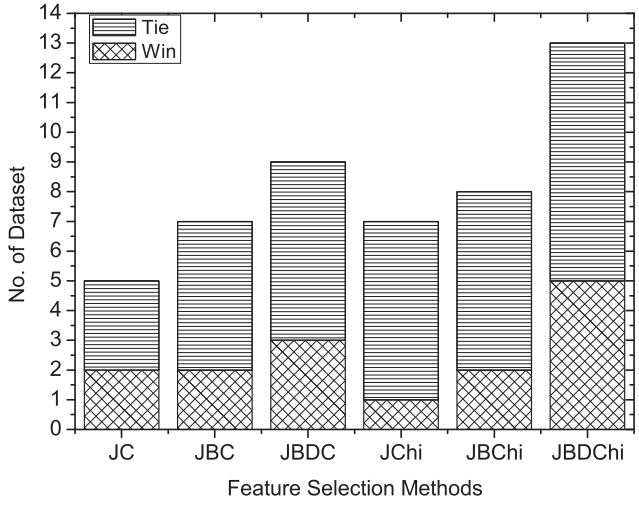

Fig. 3. Performance analysis based on accuracy. Here, the number of wins and ties indicate the number datasets for which the method performs the best and ties with other methods, respectively.

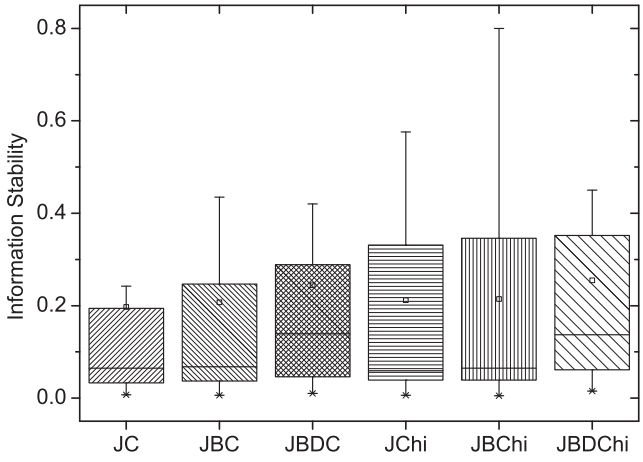

Fig. 4. Performance analysis based on information stability. Here, mean is represented by a smaller rectangle within the bigger one and median is represented by a horizontal bar within the bigger rectangle.

Table 5
Ranking based on Score and frequency in Pareto Optimal Set.

Frequency in Pareto Optimal Set					
JBDChi (26)	JBDC (24)	JBChi(18)	JBC(18)	JC(7)	JChi(6)
Average Score					
JBDChi(0.531)	JBDC(0.526)	JBChi(0.508)	JBC(0.494)	JChi(0.442)	JC(0.388)

Table 6
Comparison among different methods based on its accuracy. (*) and (°) represents that mDSM wins and loses significantly from that method, respectively.

Dataset	SVM				KNN				XGBoost			
	mDSM	JC	JMI	SPEC_CMI	mDSM	JC	JMI	SPEC_CMI	mDSM	JC	JMI	SPEC_CMI
Iris	**0.973(2)**	0.913(2)*	0.927*	0.927*	**0.953**	0.833*	0.780*	0.780*	–	–	–	–
Pima	**0.755(7)**	0.736(8)	0.739	0.738	**0.588**	0.587	0.535*	0.544*	0.706	0.730	0.716	**0.740**
Yeast	**0.562(6)**	0.500(7)*	0.509*	0.512*	**0.401**	0.303*	0.320*	0.335*	**0.569**	0.507*	0.518*	0.522*
Glass	**0.652(7)**	0.517(7)*	0.530*	0.530*	**0.630**	0.513*	0.517*	0.522*	**0.670**	0.561*	0.561*	0.548*
Wine	**0.989(12)**	0.916(9)*	0.958*	0.953*	0.958	0.842*	0.963	**0.968**	0.937	0.905	**0.958**	0.947
Heart	**0.822(9)**	0.811(10)	0.819	0.811	**0.767**	0.719*	0.748	**0.767**	0.807	0.744*	0.815	**0.822**
Australian	**0.876(11)**	0.683(11)*	**0.876**	0.871	**0.817**	0.593*	0.763*	0.780*	**0.897**	0.651*	0.859*	0.883
Segment	**0.921(16)**	0.888(12)*	0.892*	0.892*	**0.919**	0.869*	0.878*	0.879*	**0.929**	0.915*	0.911*	0.913*
Cardio	**0.818(20)**	0.633(13)*	0.700*	0.696*	**0.743**	0.619*	0.683*	0.694*	**0.754**	0.648*	0.705*	0.706*
Waveform	**0.856(19)**	0.808(13)*	0.840*	0.839*	**0.765**	0.703*	0.748*	0.748*	**0.796**	0.768*	0.777*	0.777*
Parkinson	**0.850(18)**	0.845(14)	0.820	0.825	**0.925**	0.920	0.915	0.915	0.870	0.880	**0.905**	**0.905**
Steel	**0.710(25)**	0.696(20)	**0.710**	0.709	**0.704**	0.693	0.696	0.698	**0.723**	0.704	0.719	0.720
Breast	**0.972(28)**	0.957(20)	0.959	0.959	**0.948**	0.922	0.933	0.938	**0.947**	0.933	0.941	**0.947**
Ionosphere	**0.867(32)**	0.850(33)	0.814*	0.825	0.853	0.864	0.858	**0.872**	**0.925**	0.914	0.911	0.903
Dermatology	**0.960(33)**	0.953(23)	0.958	**0.960**	0.948	0.935	0.945	**0.950**	0.943	0.937	**0.945**	**0.945**
Spambase	**0.855(56)**	0.733(41)*	0.746*	0.748*	**0.854**	0.673*	0.684*	0.684*	**0.827**	0.704*	0.717*	0.716*
Sonar	0.741(22)	0.727(60)	**0.750**	0.745	0.850	**0.886**	0.873	0.864	**0.768**	0.750	0.741	0.755
Musk	**0.817(138)**	**0.817(165)**	0.798	0.788	**0.869**	0.852	0.863	0.838	**0.804**	0.800	0.777	0.788
Semeion	**0.935(255)**	0.932(254)	0.932	0.931	**0.915**	0.912	0.912	0.913	0.787	**0.791**	0.786	0.787
Arrhythmia	**0.744(120)**	0.707(253)*	**0.744**	0.728	**0.651**	0.586*	0.621	0.602	**0.737**	0.693*	0.705	0.695*
Madelon	**0.618(26)**	0.543(500)*	0.603	0.591*	**0.783**	0.529*	0.756*	0.728*	**0.806**	0.718*	0.747*	0.748*
Lung	**0.960(2569)**	0.945(3294)	0.955	0.955	0.939	**0.941**	0.929	0.934	**0.905**	0.868	0.904	0.859
Colon	**0.855(148)**	**0.855(1987)**	0.790	0.823	**0.806**	0.694	0.774	0.742	0.661	0.726	0.677	**0.758**
Win/Tie/Loss(EW)	–	21/2/0	19/3/1	21/1/1	–	20/0/3	20/0/3	18/1/4	–	18/0/4	16/0/6	14/2/6
Sig. Win/Loss(EW)	–	11/0	9/0	9/0	–	12/0	10/0	10/0	–	10/0	8/0	8/0
Win/Tie/Loss(EF)	–	21/0/2	20/2/1	18/2/3	–	19/1/3	14/2/7	13/2/8	–	19/0/3	16/1/5	17/0/5
Sig. Win/Loss(EF)	–	9/0	4/0	4/0	–	10/2	6/2	6/3	–	11/0	5/0	5/0

5.4.2. Impact of search strategy

Figs. 3,4 and Table 5 also compare the impact of χ^2 and COBRA based search strategies. Comparing JC vs JChi and JBC vs JBChi, it can be observed that even though χ^2 based search performs somewhat better than COBRA, there is no major difference between these two strategies in terms of the performance metrics used in this paper.

5.4.3. Impact of discretization

Apart from simple pre-defined EW discretization, simultaneous feature selection and discretization can be performed in two ways: using static and dynamic discretization. For the first one, JBMI and the Relevancy based discretization (Algorithm 1) are examined with COBRA (namely JBDC) and with the proposed χ^2 based search technique (namely JBDChi). For the second one, DD using *Rrc* is incorporated (Algorithm 2) which is the proposed mDSM framework.

5.4.3.1. Static discretization. Fig. 3 represents the overall comparison of accuracies among JC, JBC, JChi, JBChi, JBDC and JBDChi. Note that, the inclusion of SD improves the overall performance of JBDC and JBDChi compared to other methods and JBDChi performs the best. In terms of accuracy, JBC is similar to JBDC and JBChi is similar to JBDChi. However, adding SD reduces number of features for JDBC and JDBChi which is further discussed later in this section. Fig. 4 represents the comparison of these methods based on IS. It indicates that stability of the methods has dependency on the discretization scheme. It also shows that both JBDC and JBDChi exhibit better stability compared to the methods that use a fixed pre-defined discretization scheme. Furthermore, JBDChi is the most stable method as its mean (0.255) and median (0.157) values are higher than other methods. Therefore, it can be concluded that adding SD increases the stability of the selected features.

The ranking of these methods based on their frequency in PO set and Score is presented in Table 5. Here, PO and Score are computed by considering accuracy, stability, and *f_score*. This Score is

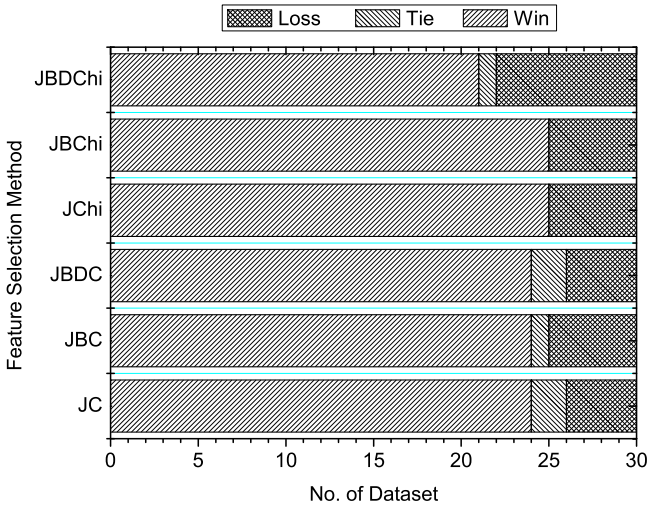

Fig. 5. Performance comparison of mDSM with other methods based on accuracy. Win/Tie/Loss means mDSM performs better/equally-well/ worse than the alternatives.

Table 7
Ranking based on Score and frequency in Pareto Optimal Set.

	EW discretization				EF discretization			
Frequency in Pareto Optimal Set (3 Criteria)								
SVM	mDSM (21)	JC (11)	JMI(6)	SPEC_CMI(4)	mDSM(21)	JC (16)	SPEC_CMI(3)	JMI(2)
KNN	mDSM (22)	JC (13)	JMI(6)	SPEC_CMI(5)	mDSM (20)	JC (17)	JMI(8)	SPEC_CMI(7)
XGBoost	mDSM (20)	JC (11)	JMI(8)	SPEC_CMI(7)	mDSM (19)	JC (14)	JMI(7)	SPEC_CMI(3)
Frequency in Pareto Optimal Set (2 Criteria)								
SVM	mDSM (20)	JMI(4)	SPEC_CMI(2)	JC (1)	mDSM (21)	JC(5)	SPEC_CMI(3)	JMI (1)
KNN	mDSM (22)	JMI(5)	SPEC_CMI(4)	JC (3)	mDSM (20)	JMI(7)	SPEC_CMI(6)	JC (6)
XGBoost	mDSM (18)	SPEC_CMI(6)	JMI(5)	JC (2)	mDSM (19)	JMI(5)	JC(5)	SPEC_CMI(3)
Average Score (3 Criteria)								
SVM	mDSM (0.414)	JMI(0.405)	SPEC_CMI(0.403)	JC (0.366)	mDSM (0.415)	SPEC_CMI(0.403)	JMI(0.403)	JC (0.370)
KNN	mDSM (0.406)	JMI(0.395)	SPEC_CMI(0.394)	JC (0.352)	mDSM (0.408)	SPEC_CMI(0.401)	JMI(0.400)	JC (0.363)
XGBoost	mDSM (0.395)	JMI(0.387)	SPEC_CMI(0.387)	JC (0.351)	mDSM (0.395)	SPEC_CMI(0.387)	JMI(0.387)	JC (0.355)

calculated by averaging the Score of all thirty datasets. In terms of Score and PO, JBDChi also produces a good balance between stability and accuracy with a small set of selected features and thus, possesses the highest rank. From the above discussion and from Tables 4, 5, Figs. 3, and 4, it is evident that the incorporation of bias, χ^2 search, and SD have their individual effect on the performances.

5.4.3.2. Dynamic discretization. Since performance improves when one moves from fixed pre-defined EW discretization to SD, it is expected that further improvements can be obtained using DD. mDSM incorporates DD along with JBMI and χ^2 based search. To examine the effect of DD, mDSM is compared in Fig. 5 with JC, JBC, JBDC, JChi, JBChi and JBDChi in term of accuracy. mDSM performs far better compared to others. This is because, JBDC and JBDChi only discretize the features individually based on their relevancy while mDSM considers the inter-dependency among the features. mDSM, however, selects a larger set of features. This is because, it selects features if they contain CI and uses a relaxed condition to select the features (it can even select a feature if it is not significantly redundant). It increases the accuracy which is a major advantage of mDSM. However, in term of IS, other methods perform slightly better than mDSM. This presumably happens due to its dependency on relevance based ranking. Therefore, it can be concluded that, if the goal of feature selection is only to increase the classification accuracy, one should use mDSM. On the other hand, if better stability with small set of features is desired, then JBDChi will be the better choice.

5.5. Comparison of mDSM with state-of-the-art methods

Table 6 presents the comparison of mDSM with other state-of-the-art MI based methods, namely, JC, JMI, and SPEC_CMI. Among them JC is a feature selection method and SPEC_CMI and JMI are feature ranking methods. Therefore, for JMI and SPEC_CMI, the same number of features that mDSM produces are used to generate the results. Three very different classifiers are used in this experiment to examine whether the methods are classifier

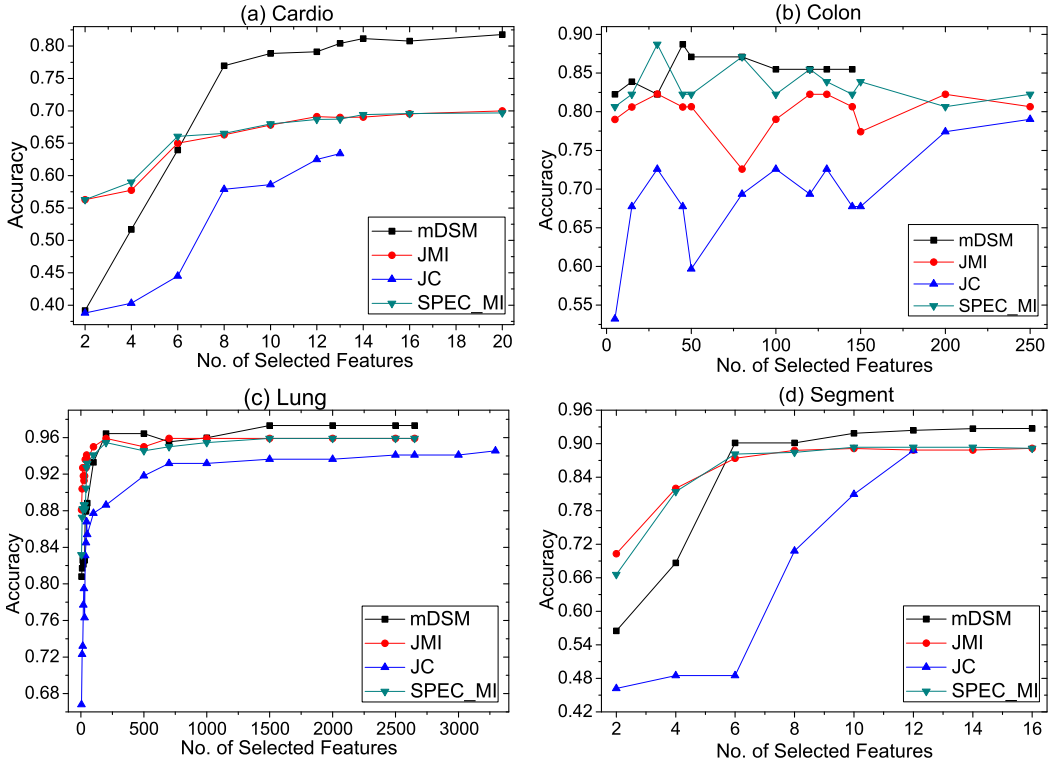

Fig. 6. Comparison for different number of selected features.

independent or not. Moreover, both EW and EF discretizations are applied on these methods to observe their overall impact on the feature selection process. However, only the accuracies for EW discretization is presented in Table 6 and the values in boldface represent the accuracy of the best performing method for a particular classifier. Performance for both EW and EF are summarized in the last four rows of the table. It can be easily observed that mDSM performs best (more wins) irrespective of the classifier and discretization method being used. To evaluate the significance of the improvements in accuracies among different methods, paired t-tests (at 95% significance level) are performed. It can be observed that all methods with EW discretization achieve similar number of wins compared to EF but the number of significant wins is higher for EW. More specifically, no significant losses occur for mDSM in EW discretization irrespective of the classifiers.

In case of EF, it is found that mDSM loses significantly with KNN classifier for three datasets, namely, *Parkinson, Dermatology* and *Sonar*. However, for these datasets mDSM wins for both SVM and XGBoost. Again performance of mDSM with KNN can be improved for *Parkinson* dataset (from 84% to 93.5%), if the discretization level of the first feature is not altered during the selection of the second feature. This is likely due to the fact that mDSM as a greedy method is influenced by the first selected feature and its discretization level on selection of the subsequent features. However, no such improvements for the other two datasets were observed.

On the other hand, from PO and Score (presented in Table 7), it is observed that mDSM wins for both cases: when three (accuracy, stability and *f_score*) or two (accuracy and stability) criteria are considered for calculating PO and average Score using the datasets presented in Table 6. Moreover, mDSM is the top ranked method (both for EW and EF) in terms of PO and Score for all classifiers. The ranking of the other selection methods varies in case of different discretization for SVM and XGBoost classifiers. One interesting observation is that when three criteria are considered for PO, JC ranks better than JMI and SPEC_CMI. However, in case of PO with two criteria and Score, JC is found to be ranked last in most cases. This is because although JC has poorer accuracy and stability, it selects comparatively lower number of features.

Since the number of selected features of mDSM is usually high, it will be interesting to know how it performs compared to other methods with the same number of selected features. This will indicate the quality of the features being selected and is shown here in Fig. 6 using EW discretization (similar analysis is done in [12]). To understand Fig. 6 clearly, let us consider the *Cardio* dataset. Here JC produces 63% accuracy with thirteen selected features and with the same number of selected features mDSM, JMI and SPEC_CMI obtain 80%, 68.9% and 68.7% accuracy respectively.

Analyzing Fig. 6, it is found that in some cases even though the total number selected features exhibit better performances, mDSM performs worse if the number of selected features is lower than that of other methods. This is expected as mDSM follows a greedy search strategy and it is not a ranking method. Hence, the feature selected first is not necessarily the best feature. In contrast, JMI and SPEC_CMI are ranking methods and thus their performances are better when fewer selected features are used. Moreover, for most of the cases, JMI and SPEC_CMI produce similar results. Even though fluctuating behavior is observed when lower number of features are used, the performance of all methods become almost stable for large number of selected features.

6. Conclusion

This paper proposes mDSM, a framework that incorporates feature selection and discretization simultaneously using a bias corrected MI based criteria. The decision about discretization level and selection of features are taken based on the critical value of χ^2 tests. Similar to other filter based methods, mDSM is classifier independent. The improvements due to each of the contributions are shown step by step using an existing feature selection method, namely JMI. Incorporating these three aforementioned contributions with other feature selection criteria (e.g., mRMR, relaxMRMR, and MRI) will also likely to improve their performance. This will be examined in future.

However, in terms of the number of selected features and stability mDSM is not as good as the proposed JBDChi. The only difference between JBDChi and mDSM is that, the former uses static while the latter uses dynamic discretization. mDSM has two limitations. First, it needs to tune the value of δ (Algorithm 2) for each dataset. Second, for some datasets better performance can be achieved using a subset of the features selected by mDSM. For example, in case of *Arrythmia* dataset, the accuracy is 0.744 with 120 selected features (for $\delta = 7$). However, this accuracy can be higher (0.78) with less number of features (60) for the same value of δ. We believe these limitations can be mitigated by addressing the issues related to finding the optimal discretization level and feature set. One possible direction is to add a backward pass in mDSM. As the best the value of δ is related to the number of classes, number of instances, and to the number of features of a dataset, an important research direction could be to define a mechanism relating these issues to obtain the desired δ automatically.

Acknowledgment

This research is supported by ICT Division, Ministry of Posts, Telecommunications and Information Technology, Bangladesh. 56.00.0000.028.33.065.16-747, 21-06-2016.

References

[1] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (26) (2003) 1157–1182.
[2] R. de Matos Simoes, F. Emmert-Streib, Influence of statistical estimators of mutual information and data heterogeneity on the inference of gene regulatory networks, PLoS ONE 6 (12) (2011) e29279.
[3] M. Shoyaib, M. Abdullah-Al-Wadud, S.Z. Ishraque, O. Chae, Facial expression classification based on Dempster–Shafer theory of evidence, in: Belief Functions: Theory and Applications, Springer, 2012, pp. 213–220.
[4] J. Feng, L. Jiao, F. Liu, T. Sun, X. Zhang, Unsupervised feature selection based on maximum information and minimum redundancy for hyperspectral images, Pattern Recognit. 51 (2016) 295–309.
[5] P.M. Rasmussen, L.K. Hansen, K.H. Madsen, N.W. Churchill, S.C. Strother, Model sparsity and brain pattern interpretation of classification models in neuroimaging, Pattern Recognit. 45 (6) (2012) 2085–2100.
[6] J. Ma, H. Luo, W. Zhou, Y. Song, B. Hui, Z. Chang, Discriminative feature selection for visual tracking, J. Phys. Conf. Ser. 844 (1) (2017) 012046.
[7] K. Lee, I. Lee, Bayesian Network and Feature Selection for Rank Deficient Inverse Problem, in: Proceedings of ICIPE, 2017. World Academy of Science, Engineering and Technology
[8] M. Bennasar, Y. Hicks, R. Setchi, Feature selection using joint mutual information maximisation, Expert Syst. Appl. 42 (22) (2015) 8520–8532.
[9] M.F.B. Wanderley, V. Gardeux, R. Natowicz, A. de Pádua Braga, GA-KDE-Bayes: an evolutionary wrapper method based on non-parametric density estimation applied to bioinformatics problems, in: Proceedings of the Twenty First European Symposium on Artificial Neural Networks–ESANN, 2013, pp. 155–160.
[10] S. Sharmin, M.R. Arefin, M. Abdullah-Al Wadud, N. Nower, M. Shoyaib, SAL: an effective method for software defect prediction, in: Proceedings of the Eighteenth ICCIT, IEEE, 2015, pp. 184–189.
[11] J.I. Khan, A.U. Gias, M.S. Siddik, M.H. Rahman, S.M. Khaled, M. Shoyaib, An attribute selection process for software defect prediction, in: Proceedings of the ICIEV, IEEE, 2014, pp. 1–4.
[12] G. Brown, A. Pocock, M.-J. Zhao, M. Luján, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (1) (2012) 27–66.
[13] G.-X. Yuan, K.-W. Chang, C.-J. Hsieh, C.-J. Lin, A comparison of optimization methods and software for large-scale l1-regularized linear classification, J. Mach. Learn. Res. 11 (2010) 3183–3234.
[14] S. Lee, Y.-T. Park, B.J. dAuriol, A novel feature selection method based on normalized mutual information, Appl. Intell. 37 (1) (2012) 100–120.
[15] A. Senawi, H.-L. Wei, S.A. Billings, A new maximum relevance-minimum multicollinearity (MRmMC) method for feature selection and ranking, Pattern Recognit. 67 (2017) 47–61.

[16] T. Naghibi, S. Hoffmann, B. Pfister, A semidefinite programming based search strategy for feature selection with mutual information measure, IEEE Trans. Pattern Anal. Mach. Intell. 37 (8) (2015) 1529–1541.
[17] J.R. Vergara, P.A. Estévez, A review of feature selection methods based on mutual information, Neural Comput. Appl. 24 (1) (2014) 175–186.
[18] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.
[19] D.D. Lewis, Feature selection and feature extraction for text categorization, in: Proceedings of the Workshop on Speech and Natural Language, in: HLT, 1992, pp. 212–217.
[20] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Netw. 5 (4) (1994) 537–550.
[21] H. Yang, J. Moody, Feature selection based on joint mutual information, in: Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis, 1999, pp. 22–25.
[22] S. Panzeri, A. Treves, Analytical estimates of limited sampling biases in different information measures, Netw. Comp. Neural 7 (1995) 87–107.
[23] S. Panzeri, R. Senatore, M.A. Montemurro, R.S. Petersen, Correcting for the sampling bias problem in spike train information measures, J. Neurophysiol. 98 (3) (2007) 1064–1072.
[24] S. Garcia, J. Luengo, J.A. Sáez, V. Lopez, F. Herrera, A survey of discretization techniques: taxonomy and empirical analysis in supervised learning, IEEE Trans. Knowl. Data Eng. 25 (4) (2013) 734–750.
[25] H. Liu, R. Setiono, Chi2: feature selection and discretization of numeric attributes, in: Proceedings of the Seventeenth International Conference on Tools with Artificial Intelligence, IEEE, 1995, pp. 388–391.
[26] U. Fayyad, K. Irani, Multi-interval discretization of continuous-valued attributes for classification learning, in: Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 1993, pp. 1022–1029.
[27] L.A. Kurgan, K.J. Cios, Caim discretization algorithm, IEEE Trans. Knowl. Data Eng. 16 (2) (2004) 145–153.
[28] C.J. Tsai, C.I. Lee, W.P. Yang, A discretization algorithm based on class-attribute contingency coefficient, Inf. Sci. 178 (3) (2008) 714–731.
[29] A. Cano, D.T. Nguyen, S. Ventura, K.J. Cios, Ur-caim: improved caim discretization for unbalanced and balanced data, Soft Comput. 20 (1) (2016) 173–188.
[30] J.R. Quinlan, C4. 5: Programs for Machine Learning, Elsevier, 2014.
[31] W.-H. Au, K.C. Chan, A.K. Wong, A fuzzy approach to partitioning continuous attributes for classification, IEEE Trans. Knowl. Data Eng. 18 (5) (2006) 715–719.
[32] K.Z. Mao, Orthogonal forward selection and backward elimination algorithms for feature subset selection, IEEE Trans. Syst. Man. Cybern. Syst 34 (1) (2004) 629–634.
[33] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (Mar) (2003) 1157–1182.
[34] X.V. Nguyen, J. Chan, S. Romano, J. Bailey, Effective global approaches for mutual information based feature selection, in: Proceedings of the Twentieth ACM SIGKDD, ACM, 2014, pp. 512–521.
[35] S. Sharmin, A.A. Ali, M.A.H. Khan, M. Shoyaib, Feature selection and discretization based on mutual information, in: Proceedings of the icIVPR, IEEE, 2017, pp. 1–6.
[36] S. Tabakhi, P. Moradi, Relevance–redundancy feature selection based on ant colony optimization, Pattern Recognit. 48 (9) (2015) 2798–2811.
[37] F. Barani, M. Mirhosseini, H. Nezamabadi-pour, Application of binary quantum-inspired gravitational search algorithm in feature subset selection, Appl. Intell. 47 (2) (2017) 304–318.
[38] N.X. Vinh, S. Zhou, J. Chan, J. Bailey, Can high-order dependencies improve mutual information based feature selection? Pattern Recognit. 53 (2016) 46–58.
[39] J. Wang, J.-M. Wei, Z. Yang, S.-Q. Wang, Feature selection by maximizing independent classification information, IEEE Trans. Knowl. Data Eng. 29 (4) (2017) 828–841.
[40] P.A. Estévez, M. Tesmer, C.A. Perez, J.M. Zurada, Normalized mutual information feature selection, IEEE Trans. Neural Netw. 20 (2) (2009) 189–201.
[41] K.S. Balagani, V.V. Phoha, On the feature selection criterion based on an approximation of multidimensional mutual information, IEEE Trans. Pattern Anal. Mach. Intell. 32 (7) (2010) 1342–1343.
[42] H. Cheng, Z. Qin, W. Qian, W. Liu, Conditional mutual information based feature selection, in: Proceedings of the International Symposium on Knowledge Acquisition and Modeling, IEEE, 2008, pp. 103–107.
[43] R. Moddemeijer, On estimation of entropy and mutual information of continuous distributions, Signal Process. 16 (3) (1989) 233–248.
[44] B. Goebel, Z. Dawy, J. Hagenauer, J.C. Mueller, An approximation to the distribution of finite sample size mutual information estimates, in: Proceedings of the International Conference on Communications, 2, IEEE, 2005, pp. 1102–1106.
[45] S.N. Roy, S. Roy, Some Aspects of Multivariate Analysis, Wiley NY, 1957.
[46] S. Boyd, L. Vandenberghe, Convex optimization, CUP, 2004.
[47] Uci, (http://archive.ics.uci.edu/ml/), (Accessed on 04/10/2016).
[48] scikit-feature feature selection repository, (http://featureselection.asu.edu/index.php), (Accessed on 04/10/2016).
[49] T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in: Proceedings of the Twenty Second ACM SIGKDD, ACM, 2016, pp. 785–794.
[50] G. Gulgezen, Z. Cataltepe, L. Yu, Stable and accurate feature selection, in: Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2009, pp. 455–468.

Sadia Sharmin received her Bachelor degree in Software Engineering from the Institute of Information Technology, University of Dhaka in 2015. Currently, she is an M.S. student at the same institution. Her research interest includes Software Engineering and Pattern Recognition.

Mohammad Shoyaib received his M.S. degree in computer science from the University of Dhaka, Bangladesh, in 2000 and in 2012, he has completed his Ph.D. degree from the Kyung Hee University, South Korea. His research interests include pattern recognition and machine learning.

Amin Ahsan Ali received his M.S. degree in CSE from the University of Dhaka, Bangladesh, in 2003, and in 2014 he completed his Ph.D. degree from the University of Memphis, USA. Currently he is an Associate Professor in the Department of Computer Science and Engineering at Independent University, Bangladesh. Before joining Independent University, he had been working as an Assistant Professor in the Department of Computer Science and Engineering at the University of Dhaka. His research interests include Machine Learning and Data Science.

Muhammad Asif Hossain Khan received his Ph.D. from the University of Tokyo. He is an Associate Professor in the Department of Computer Science and Engineering at the University of Dhaka. His research interest is natural language processing, information retrieval, microblog analysis and machine learning.

Oksam Chae (M92) received his M.S. and Ph.D. degrees in electrical and computer engineering from the Oklahoma State University, in 1982 and 1986, respectively. Currently he is a Professor in the Kyung Hee University, South Korea. His research interests include multimedia data and image processing.