Efficient multivariate feature filter using conditional mutual information

J. Lee and D.-W. Kim

This reported work focuses on designing an efficient feature filter from a computational point of view. The proposed multivariate filter method using conditional mutual information selects a feature subset that maximises relevance to target class while minimising redundancy between features in $O(N)$ time.

Introduction: The univariate filter methods are the most widely-used feature selection techniques because they are computationally simple and fast. However, univariate filters are limited to take feature dependencies into account. To capture the redundancy between features, a number of multivariate filter methods have been presented. Of the multivariate filters, the minimal-redundancy-maximal-relevance (mRMR) is the most well-known method [1]. Given the input data with N features $X = \{x_1, \ldots, x_N\}$ and the class variable C, the objective of the mRMR is to find a feature subset $S \subset X$ with $n < N$ features that have the largest dependency on the target class C. The selection of the ith feature from the set $\{X - S_{i-1}\}$, where S_{i-1} is a feature set with $i-1$ features, is done by optimising the following condition:

$$\max_{x_i \in X - S_{i-1}} \left[I(x_i; C) - \frac{1}{i-1} \sum_{x_j \in S_{i-1}} I(x_i; x_j) \right]$$

where $I(x; y) = \sum\sum p(x, y) \log (p(x, y)/p(x)p(y))$ is a mutual information between x and y with their probability density functions $p(x)$, $p(y)$, and $p(x, y)$. However, the mRMR-type multivariate filters require an additional computation time to select mutually exclusive features, which takes a worst-case time complexity $O(N^2)$ due to the calculation of redundancy $\sum I(x_i, x_j)$; it is computationally prohibitive to large-scale real-time applications such as wireless sensor networks, electric power prediction, real-time EEG recognition where data editing (adding, deleting) frequently occurs on data sets [2–4]. To reduce the cost of repetitive feature selection, for the first time to our knowledge, we propose an efficient multivariate filter method, taking a worst-case time complexity $O(N)$, which is based on conditional mutual information (CMI) to maximise relevance while minimising redundancy.

Proposed method: The mutual information between the input feature x_i and class C can be represented by the entropy $H(x) = \sum_x p(x) \log p(x)$:

$$I(x_i; C) = H(C) - H(C|x_i) = H(C) + H(x_i) - H(x_i, C)$$

Given a current feature subset S_{i-1}, the proposed method selects the next feature x_i through an iterative process that maximises the CMI $I(x_i; C|S_{i-1})$:

$$\begin{aligned} I(x_i; C|S_{i-1}) &= H(C|S_{i-1}) - H(C|x_i, S_{i-1}) \\ &= H(S_{i-1}, x_i) + H(S_{i-1}, C) \\ &\quad - H(S_{i-1}, x_i, C) - H(S_{i-1}) \end{aligned}$$

A higher value of $I(x_i; C|S_{i-1})$ indicates that x_i is informative and less correlated to the previously selected S_{i-1}. To maximise efficiency, the presented method consists of two steps. 1. For each feature, it calculates the mutual information $I(x_i; C)$, and sorts the features in descending order according to $I(x_i; C)$. Let us denote the sorted features by $F = (f_1, \ldots, f_N)$. The top-ranked feature f_1 is first added to S because it is the most relevant to the class. 2. For the remaining features in F, it computes $I(f_i; C|S_{i-1})$ one-by-one and examines if f_i brings information about C that is not contained in S_{i-1}. Moreover, it should satisfy $I(f_i; C|S_{i-1}) > I(f_{i+1}; C|S_{i-1})$ to be added to the subset. In theory, f_i forms an approximate Markov blanket for f_{i+1} [5]; f_{i+1} can be removed if f_{i+1} and C are conditionally independent given f_i.

The procedural steps of the proposed method are as follows.

Step 1: Set $S \leftarrow \phi$; for $\forall x_i \in X$, calculates $I(x_i; C)$, and sorts the features in descending order accordingly, yielding a sorted list of features $F = (f_1, \ldots, f_N)$.
Step 2: Set $S_1 \leftarrow \{f_1\}$, and $i \leftarrow 2$.
Step 3: If $I(f_i; C|S_{i-1}) > I(f_{i+1}; C|S_{i-1})$, then $S_i \leftarrow \{f_i\}$.
Step 4: Set $i \leftarrow i+1$, and go to step 3 until all features in F are examined.
Step 5: Output the set S containing the selected features.

The CMI calculation of each feature is performed only once because the CMIs of the higher ranked features were already calculated in previous iterations.

Results: To test the proposed method, we applied the mRMR and the proposed method to various data sets and compared the performances of the two methods in terms of both efficiency and dependency; the data sets employed were Secom [6], Madelon [6], Colon Cancer [7], Leukemia [8] (Table 1). Table 2 lists the execution time (seconds) for selecting the top 10 and 30 features ($|S| = 10,30$) by the two methods.

Table 1: Data sets used in experiments

Data sets	Number of patterns	Number of features	Number of classes
Secom [6]	1567	590	2
Madelon [6]	2600	500	2
Colon cancer [7]	62	2000	2
Leukemia [8]	72	7129	2

Table 2: Execution time (seconds) for selecting top 10 and 30 features

Number of features	Secom		Madelon		Colon cancer		Leukemia	
	mRMR	Proposed	mRMR	Proposed	mRMR	Proposed	MRMR	Proposed
$\|S\| = 10$	14.4	0.2	19.7	0.2	10.7	0.1	13.9	0.1
$\|S\| = 30$	44.2	0.7	59.0	0.8	31.3	0.1	35.7	0.1

It is evident that the proposed method gives markedly more efficient performance than the mRMR for all data sets. The speedup by the proposed method becomes obvious when the feature dimension increases. For example, for Leukemia data set, the mRMR takes about 13.9 and 35.7 seconds to select the top 10 and 30 features, respectively. In contrast, the proposed method takes about 0.1 second to select any features. Fig. 1 shows the comparison of dependency, in terms of $I(S; C)$, between the class label and the feature subset selected by each method for the Secom data set. The features selected by the proposed method consistently outperform those selected by the mRMR. It is observed that the dependency by the proposed method is rapidly increased when the feature size is larger than 20. When nearly 50 features are selected, the performances of the two methods become close. Fig. 2 shows the time cost (seconds) by the two methods with increasing the size of the feature subset for the Leukemia data set. The running time for the mRMR is a linear function of the number of selected features, whereas, for the proposed method, it is almost constant. Similar to the Leukemia data, the proposed method showed superior performance to the mRMR method for the other data sets. These results demonstrate the efficiency of the proposed method in a high-dimensional feature space, highlighting potential scalability to real-time applications.

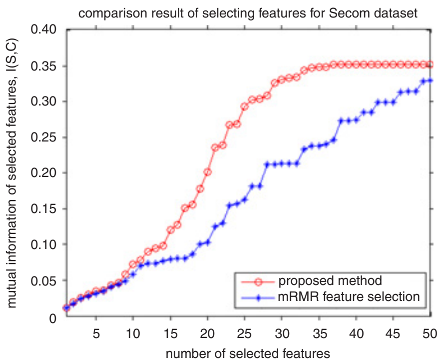

Fig. 1 *Comparison of dependency, I(S;C), between class label and feature subset selected by each method for Secom data set*