Exploration of Heuristic-Based Feature Selection on Classification Problems

Qi Qi[1], Ni Li[2(✉)], and Weimin Li[1]

[1] College of Information Science and Technology, Hainan University, Haikou 570228, China
qqi@hainu.edu.cn
[2] School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China
nl_hainnu@163.com

Abstract. We present two heuristics for feature selection based on entropy and mutual information criteria, respectively. The mutual-information-based selection algorithm exploiting its submodularity retrieves near-optimal solutions guaranteed by a theoretical lower bound. We demonstrate that these heuristic-based methods can reduce the dimensionality of classification problems by filtering out half of its features in the meantime still improving classification accuracy. Experimental results also show that the mutual-information-based heuristic will most likely collaborate well with classifiers when selecting about a half size of features, while the entropy-based heuristic will help most in the early stage of selection when choosing a relatively small percentage of features. We also demonstrate a remarkable case of feature selection being used in classification on a medical dataset, where it can potentially save half of the cost on the diabetes diagnosis.

Keywords: Feature selection · Heuristic · Dimensional complexity · Classification · Machine learning

1 Introduction

Big data often comes with high dimensionality, which makes machine learning tasks difficult. Learning from higher dimensional datasets theoretically needs more samples than from lower ones, which will make learning tasks less efficient. The dimensionality of a problem is usually correlated with the feature size of its dataset. By selecting out a subset of features and using them in a learning task, one can reduce the dimensionality of the problem. The selecting process is often referred to feature selection [1]. In the context of classification problems in machine learning, it can improve the scalability of training and predicting processes, and increase resulted classifiers' accuracy by eliminating irrelevant or noisy attributes.

Methods of feature selection can be briefly divided into two categories, the filter-based and the wrapper-based [2, 3]. The filter-based approach depends on characteristics of training data to select features without any learning process. The wrapper-based approach applies a learning algorithm to evaluate selected features. It could return a better result than the filter-based approach, but it also incurs more

G. Chen et al. (Eds.): PAAP 2017, CCIS 729, pp. 95–107, 2017.
DOI: 10.1007/978-981-10-6442-5_9

computational cost than the latter. Another categorization is based on the size of results, whether it will return all features but with different weights or only a subset of its. Respectively, they are called feature weighting and subset selection [4–6].

Selection problems also exist in other fields. For example, variable or model selection is a typical problem in statistics. The goal is to select a subset of variables from usually a linear regression model to maximize the predictive accuracy with the strongest effects of predictors [7, 8]. In mathematics, given a matrix A and an integer k, the column subset selection problem is to determine a permutation matrix P so that $AP = (A_1 A_2)$, in which A_1 has k columns which should be linearly independent. The matrix P can be seen as a ranking of the column attributes for the matrix A. Rank-Revealing QR (RRQR), a matrix factorization method [9, 10], is one of well-known methods to solve this problem.

In this paper, we introduce two heuristic-based methods for feature selection into the filter-based or subset selection category. The two heuristics are based on entropy and submodular mutual information, respectively. Mutual information were proved to be submodular functions [11]. The Submodularity reflects the intuitive property of diminishing returns, and it can be exploited to develop strongly polynomial time combinatorial algorithms with provable theoretical performance guarantees [12–14]. Authors in [15] demonstrated the advantage of the mutual information criterion over the entropy criterion in sensor placement problems in spatial monitoring applications.

Our contributions are as follows. First we designed two heuristic-based feature selection methods in the scenario of the Gaussian process model. Second we explored these methods under classification problems through carefully designed experiments, and demonstrated its performance and characteristics.

In the following sections, we first present an entropy-based greedy algorithm and a mutual-information-based approximate algorithm that retrieves near-optimal solutions by exploiting mutual information's submodularity. Then, we will explore their feature selection performance under classification problems in machine learning with a variety of datasets.

2 Heuristic-Based Selection Algorithms

In this section, we present two greedy algorithms that employ heuristics of entropy and submodular mutual information, respectively.

The task of feature selection is to select out a subset of features, also known as attributes, of a dataset. Considering each feature as a random variable, we assume that all features in a dataset form a joint multivariate Gaussian distribution. Then, any finite subset of these variables also have a joint Gaussian distribution. This model is also known as Gaussian Process (GP) [16].

A joint multivariate Gaussian distribution is namely:

$$P(\mathcal{X}_\mathcal{V} = \mathrm{x}_\mathcal{V}) = \frac{1}{(2\pi)^{n/2}|\Sigma_{\mathcal{V}\mathcal{V}}|^{1/2}} e^{-\frac{1}{2}(\mathrm{x}_\mathcal{V}-\mu_\mathcal{V})^T \Sigma_{\mathcal{V}\mathcal{V}}^{-1}(\mathrm{x}_\mathcal{V}-\mu_\mathcal{V})}$$

where $\mathcal{V}$ denotes the whole set of feature variable indexes with $|\mathcal{V}| = n$, $\mu_\mathcal{V}$ is the mean vector, and $\Sigma_{\mathcal{V}\mathcal{V}}$ is the covariance matrix. If we take a subset $\mathcal{A}$ from $\mathcal{V}$, then it also

satisfies that the random variable $\mathcal{X}_{\mathcal{A}} \sim \mathcal{N}(\mu_{\mathcal{A}}, \Sigma_{\mathcal{A}\mathcal{A}})$ where $\mu_{\mathcal{A}}$ is a corresponding sub-vector of $\mu_{\mathcal{V}}$, and $\Sigma_{\mathcal{A}\mathcal{A}}$ is a corresponding sub-matrix of $\Sigma_{\mathcal{V}\mathcal{V}}$. This consistency property is also called the marginalization property in GP. It also applies to the conditional probability $P(\mathcal{X}_{\mathcal{U}} |\ \mathcal{X}_{\mathcal{A}} = \mathrm{x}_{\mathcal{A}})$ that is a joint probability distribution of random variables of feature subset $\mathcal{U}$ conditional on the values $\mathrm{x}_{\mathcal{A}}$ at a selected subset $\mathcal{A}$, assuming $\mathcal{U}, \mathcal{A} \subset \mathcal{V}$. The conditional mean $\mu_{\mathcal{U}|\mathcal{A}}$ and variance $\sigma^2_{\mathcal{U}|\mathcal{A}}$ are given by:

$$\mu_{\mathcal{U}|\mathcal{A}} = \mu_{\mathcal{U}} + \Sigma_{\mathcal{U}\mathcal{A}}\Sigma^{-1}_{\mathcal{A}\mathcal{A}}(\mathrm{x}_{\mathcal{A}} - \mu_{\mathcal{A}}) \tag{1}$$

$$\sigma^2_{\mathcal{U}|\mathcal{A}} = \Sigma_{\mathcal{U}\mathcal{U}} - \Sigma_{\mathcal{U}\mathcal{A}}\Sigma^{-1}_{\mathcal{A}\mathcal{A}}\Sigma_{\mathcal{A}\mathcal{U}} \tag{2}$$

where $\mu_{\mathcal{A}}$ is the mean vector of subset variable $\mathcal{X}_{\mathcal{A}}$; $\Sigma_{\mathcal{U}\mathcal{U}}$, $\Sigma_{\mathcal{A}\mathcal{A}}$, $\Sigma_{\mathcal{U}\mathcal{A}}$ and $\Sigma_{\mathcal{A}\mathcal{U}}$ are corresponding sub-matrices of $\Sigma_{\mathcal{V}\mathcal{V}}$. For example, $\Sigma_{\mathcal{U}\mathcal{A}}$ is formed by the $\mathcal{U}$ rows and the $\mathcal{A}$ columns in $\Sigma_{\mathcal{V}\mathcal{V}}$.

The idea here is to select a subset of feature variables that minimizes the uncertainty of probability distribution comprised of the rest of unselected feature variables. In the following sections, we will present two heuristic-based methods of selecting feature variables in the Gaussian Process scenario.

2.1 The Entropy-Based Heuristic

Given a selected subset $\mathcal{A}$, the uncertainty of conditional probability $P(\mathcal{X}_i |\ \mathcal{X}_{\mathcal{A}})$ can be measured by the entropy:

$$\begin{aligned} H(\mathcal{X}_i |\ \mathcal{X}_{\mathcal{A}}) &= -\iint P(x_i, x_{\mathcal{A}}) \log P(x_i | x_{\mathcal{A}}) \mathrm{d}x_i \mathrm{d}x_{\mathcal{A}} \\ &= \frac{1}{2}\log \sigma^2_{i|\mathcal{A}} + \frac{1}{2}(\log \pi + \log 2 + 1), \end{aligned} \tag{3}$$

Note that the entropy is a monotonic function of the variance $\sigma^2_{i|\mathcal{A}}$, which can be evaluated ahead of time by Eq. (2).

Feature selection becomes a subset selection problem, where choosing a subset $\mathcal{A}$ out of the whole feature variable index set $\mathcal{V}$, so that uncertainty of the joint probability distribution of the rest of unselected variables, denoted as $\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}}$, will be minimized. Namely, the selection is made by minimizing the entropy $H(\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}} |\ \mathcal{X}_{\mathcal{A}})$. It is also equivalent to find a subset $\mathcal{A}$ that maximizes $H(\mathcal{X}_{\mathcal{A}})$, as the chain rule for conditional entropy holds that $H(\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}} |\ \mathcal{X}_{\mathcal{A}}) = H(\mathcal{X}_{\mathcal{V}}) - H(\mathcal{X}_{\mathcal{A}})$. The optimization problem turns out to be a NP-hard problem. The heuristic is to greedily select the next feature variable $y^*_{i+1} \in \mathcal{V}\setminus\mathcal{A}_i$ that has the highest conditional entropy given the current selected set $\mathcal{A}_i$:

$$y^*_{i+1} = \mathrm{argmax}_{y_{i+1}} H(\mathcal{X}_{y_{i+1}} |\ \mathcal{X}_{\mathcal{A}_i}), \tag{4}$$

The greedy algorithm is shown as in Algorithm 1.

Algorithm 1: Greedy algorithm for maximizing entropy $H(\mathcal{A})$

Input: covariance matrix $\Sigma_{\mathcal{V}\mathcal{V}}$, selection size k
Output: selected subset $\mathcal{A}(\mathcal{A} \subseteq \mathcal{V}, \text{and } |\mathcal{A}| = k)$

```
begin
    A ← ∅
    for i = 1 to k do
        foreach y ∈ V\A do δ_y ← σ²_{y|A}
        y* ← argmax_{y∈V\A} δ_y
        A ← A ∪ {y*}
    end
end
```

Where k is the selection size, and $\sigma^2_{y|\mathcal{A}}$ is computed by Eq. (2). Because the log function is monotonic, $\sigma^2_{y|\mathcal{A}}$ is proportional to $H(\mathcal{X}_y|\ \mathcal{X}_\mathcal{A})$. That means choosing a variable at y that maximizes $H(\mathcal{X}_y|\ \mathcal{X}_\mathcal{A})$ is equivalent to finding such a y that maximizes $\sigma^2_{y|\mathcal{A}}$.

The calculation of $\sigma^2_{y|\mathcal{A}}$ is expensive. Let $|\mathcal{V}| = n$, there are n times of these computations when $i = 1$, and $(n - k + 1)$ times when $i = k$. Hence, Algorithm 1 has totally $\frac{(2n-k+1)k}{2}$ times of evaluations of $\sigma^2_{y|\mathcal{A}}$.

2.2 The Mutual-Information-Based Heuristic

Another heuristic for optimizing feature subset selection is mutual information, which was originally proposed by Caselton and Zidek in [17]. The mutual information of a subset at $\mathcal{A}$ denoted as $\mathrm{MI}(\mathcal{A})$, which actually is an entropy reduction, is defined as following,

$$\begin{aligned} \mathrm{MI}(\mathcal{A}) &= \mathrm{I}(\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}}; \mathcal{X}_\mathcal{A}) \\ &= H(\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}}) - H(\mathcal{X}_{\mathcal{V}\setminus\mathcal{A}}|\ \mathcal{X}_\mathcal{A}) \\ &= H(\mathcal{X}_\mathcal{A}) - H(\mathcal{X}_\mathcal{A}\ |\ \mathcal{X}_{\mathcal{V}\setminus\mathcal{A}}) \end{aligned} \tag{5}$$

Compared with the entropy-based method, the mutual-information-based heuristic selects a subset $\mathcal{A}$ by maximizing the reduction of the entropy over the rest of the feature space $\mathcal{V}\setminus\mathcal{A}$ before and after selecting out $\mathcal{A}$. Selecting a feature subset such that,

$$\mathcal{A}^* = \mathrm{argmax}_{\mathcal{A}\subset\mathcal{V}}\mathrm{MI}(\mathcal{A}) \tag{6}$$

which is a NP-complete problem. A greedy algorithm developed in [15] selects a feature variable y maximizing the mutual information gain, namely:

$$\Delta_y = \mathrm{MI}(\mathcal{A} \cup y) - \mathrm{MI}(\mathcal{A}), \tag{7}$$

That is, it chooses the next feature variable that provides the maximal increase in the value of mutual information. In the scenario of Gaussian Process, the Δ_y can be further deduced as following:

$$\begin{aligned}\Delta_y &= \mathrm{MI}(\mathcal{A} \cup y) - \mathrm{MI}(\mathcal{A}) \\ &= H(y|\ \mathcal{A}) - H\left(y|\ \bar{\mathcal{A}}\right) \\ &= \frac{1}{2}\log_2\left(\frac{\sigma^2_{y|\mathcal{A}}}{\sigma^2_{y|\bar{\mathcal{A}}}}\right)\end{aligned}$$

where $\bar{\mathcal{A}}$ denotes variable indexes in $\mathcal{V}$ excluding selected $\mathcal{A}$ and y.

A note about the mutual information gain Δ_y is that it is monotonically decreasing as the selected subset $\mathcal{A}$ gets bigger. It inspired an enhanced version of the greedy algorithm with lazy evaluation [15].

Algorithm 2 presents the mutual-information-based heuristic of greedily selecting feature variables in the scenario of GP modelling.

Algorithm 2: Greedy algorithm for maximizing mutual information gain $\mathrm{MI}(\mathcal{A} \cup y) - \mathrm{MI}(\mathcal{A})$ with lazy evaluation

Input: covariance matrix $\Sigma_{\mathcal{V}\mathcal{V}}$, selection size k
Output: selected subset $\mathcal{A}(\mathcal{A} \subseteq \mathcal{V})$, mutual information gains Δ

```
begin
    A ← ∅
    foreach y ∈ V do  Δ_y ← +∞ ; Φ_y ← 0

    for i = 1 to k do
        repeat
            y* ← argmax_{y∈V\A} Δ_y
            if Φ_{y*} == i then
                break
            else
                Ā ← V − (A ∪ y*)
                Δ_{y*} ← (1/2) log2( σ²_{y*|A} / σ²_{y*|Ā} )
                Φ_{y*} ← i
            end
        until 0
        A ← A ∪ {y*}
    end
end
```

Where Φ_{y^*} records in which iteration Δ_{y^*} is updated. The lazy evaluation reduces an amount of computation of Δ_y based on the insight that the sequence of the mutual information gains on a fixed y decreases as the subset $\mathcal{A}$ grows. It will select the y^* if

the maximal Δ_{y^*} is updated in the current iteration, otherwise it will update Δ_{y^*} and Φ_{y^*} and will repeat the selection process.

When $|\mathcal{V}| = n$, Algorithm 2 has $2(n+k-1)$ times of evaluations of either $\sigma^2_{y^*|\mathcal{A}}$ or $\sigma^2_{y^*|\bar{\mathcal{A}}}$ in the best case. This is more efficient and scalable than Algorithm 1 when n becomes very large.

Algorithm 2 is not only efficient, but also provides its solution with a theoretic bound in terms of the optimal solution. Although the mutual information function as in Eq. (5) is not monotonic increasing, it has still been proved to be a partially monotonic submodular function [15]. According to [18], a greedy algorithm, such as the Algorithm 2, optimizing a monotonic submodular function guarantees a solution with a theoretical performance lower bound of $(1-1/e)$OPT, where OPT represents the optimal solution value.

3 Experiments

We explore the heuristic-based selection methods above under classification problems in machine learning. The purpose is to evaluate these methods in feature selection problems, and to check whether they can help classification learning tasks achieve similar or even better predictive accuracy than using full feature sets.

For comparison, we also add two other popular selection methods. One is the rank-revealing QR (RRQR) used for matrix column subset selection, and the other is the ranker method in the data mining software Weka [19]. Our heuristic-based selection methods and the RRQR method belongs to the filter-based category, while the ranker method in Weka is a wrapper-based selection method. In experiments, feature subsets are first chosen by these selection methods, then resulted attribute-filtered datasets will be fed into classifiers in Weka to calculate classification rates.

3.1 Experimental Setting

To compare the feature selection methods as shown in Table 1, we recruited 11 different classifiers (as in Table 3), and 13 different data sets (as in Table 2). Our running experiments systematically sweep feature selection size from 1 to $N-1$, in which N is the total number of features in a dataset, for each of the selection methods, classifiers and datasets.

Table 1. Feature selection methods used in experiments

ID	Method
1	Mutual-information-based heuristic(MI)
2	RRQR
3	Weka's ranker
4	Entropy-based heuristic

We programmed the mutual-information-based and entropy-based heuristics in MatLab, and used a public MatLab implementation of the RRQR provided in [20]. We chose the ranker method with a default attribute evaluation function "ReliefF" in Weka, which weights all features and returns a ranked list of its.

Datasets in Table 2 covers a variety of situations in terms of number of classes and features. For those not providing a test set, we divided the original datasets into two parts, the two-third of which for training and the rest for testing. Most of the datasets are from UCI's machine learning repository [21], the others are from the Libsvm data website [22].

Table 2. Datasets

ID	Dataset name	Class no.	Feature no.	Training no.	Testing no.
1	Australian credit	2	14	460	230
2	Diabetes	2	8	507	261
3	Glass	6	9	142	72
4	Liver disorders	2	6	230	115
5	Satimage	6	36	2217	1000
6	Vehicle	4	18	564	282
7	Breast cancer	2	9	455	227
8	German credit	2	24	667	333
9	Heart	2	13	180	90
10	Pen digits	10	16	2623	1225
11	Sonar	2	60	138	70
12	Wine	3	13	118	60
13	DNA	3	180	2000	1186

Classifiers used in experiments are shown in Table 3. Basically, we picked up one or two representatives in each of classifier categories in Weka, so that it covered a wide spectrum of classification methods. They were employed with their default settings provided in Weka during experiments. For more descriptions of the classifiers, please refer to [23, 24].

Table 3. Classifiers from weka

ID	Classifier
1	RBFNetwork
2	GaussianProcesses
3	SimpleLinearRegression
4	PaceRegression
5	SMOreg
6	KStar
7	AdditiveRegression
8	Bagging
9	RandomSubSpace
10	DecisionTable
11	M5P

Batch experiments were carried out to explore contributions of the listed feature selection methods to the performance of classification methods for each dataset systematically. Each classifier tried out all of the selection methods, and each selection method screened out all of possible selection sizes including a full feature set.

Experimental results are shown next. For convenience, feature selection methods, classifiers and datasets are represented by their ID numbers.

3.2 Experimental Results and Discussion

Table 4 summarizes the best classification rates for each dataset by corresponding combinations of feature selection methods, classifiers, and selected feature sizes in percentage. The mutual-information-based and entropy-based feature selection methods appear multiple times in the table, whereas using full features only appears one time. It shows that feature selection not only reduces the dimensionality of classification problems but also helps to improve classification accuracy.

Table 4. Summary of the combinations for the best classification scores of each dataset

DatasetID	Best rate	Selection method	Selection size (%)	ClassifierID
1	0.90	Entropy-based	35.7	4
2	0.74	Weka-ranker	37.5	9
3	0.72	Weka-ranker	55.6	6
4	0.71	RRQR	83.3	2
4	0.71	Entropy-based	83.3	2
5	0.85	Entropy-based	72.2	6
6	0.76	Weka-ranker	88.9	11
7	0.99	MI-based	55.6	2
7	0.99	RRQR	55.6	1
8	0.69	Entropy-based	54.2	8
9	0.89	MI-based	53.8	4
9	0.88	MI-based	53.8	5
10	0.94	Full-attribute	100.0	6
11	0.93	MI-based	56.7	2
12	0.99	RRQR	76.9	6
13	0.94	Weka-ranker	25.6	11

For a dataset and a classifier, the selection method leading to the best classification score with the smallest selection size was picked out as a winner. We collected winning counts for each of the feature selection methods. We also accommodated co-winning situations of near performances within a 0.5% range of classification accuracy, as well as a shared selection size.

The result of winning counts is summed up in Table 5. It shows that each of the selection methods takes up a variety of seats given a classifier. It can't tell which classifier is favoured by a particular selection method, or vice versa. But it shows clearly that classification with a full feature set wins much less than using a selected subset of features.

Table 5. Summary of winning counts of each selection method given a classifier

Classifier	MI	RRQR	Ranker	Entropy	FullAttrs
RBFNetwork	3	5	5	5	1
GaussianProcesses	4	4	3	5	2
SimpleLinearRegression	3	3	7	6	0
PaceRegression	3	3	3	4	2
SMOreg	4	3	6	7	0
KStar	2	2	7	4	0
AdditiveRegression	2	5	4	5	0
Bagging	4	5	2	4	0
RandomSubSpace	2	4	5	4	0
DecisionTable	2	3	5	7	0
M5P	4	3	3	5	0
Total count	33	40	50	56	5
Percentage %	17.9	21.7	27.2	30.4	2.7

Because the selection size is a key parameter of feature selection methods, Fig. 1 examines the distribution of winnings among different selection sizes given a selection method. There is a distinguishable difference between the mutual-information-based as

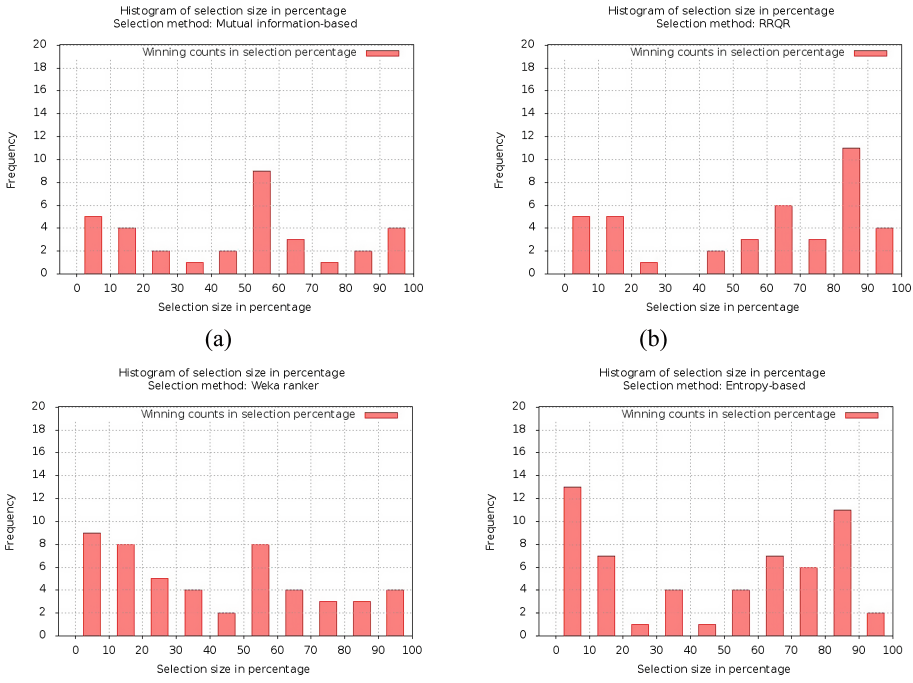

Fig. 1. Distribution of winnings among different selection sizes for each selection method

in Fig. 1a and the entropy-based as in Fig. 1d. The distribution in the first picture spikes at the middle range, while in the latter picture it scatters mostly at the two ends.

This finding reveals us a truth about the mutual information criterion. That is, the value of mutual information goes up first until about half of variables being selected, then it will go downward. Figure 2 draws a few pictures among the datasets to demonstrate the phenomenon. The number of points varies due to different feature numbers available in the datasets. Figure 3 shows for each dataset the selected feature sizes that achieve the maximal mutual information values. The numbers were converted into percentages for comparison convenience. It indicates that resulted selection sizes reside in the half size level.

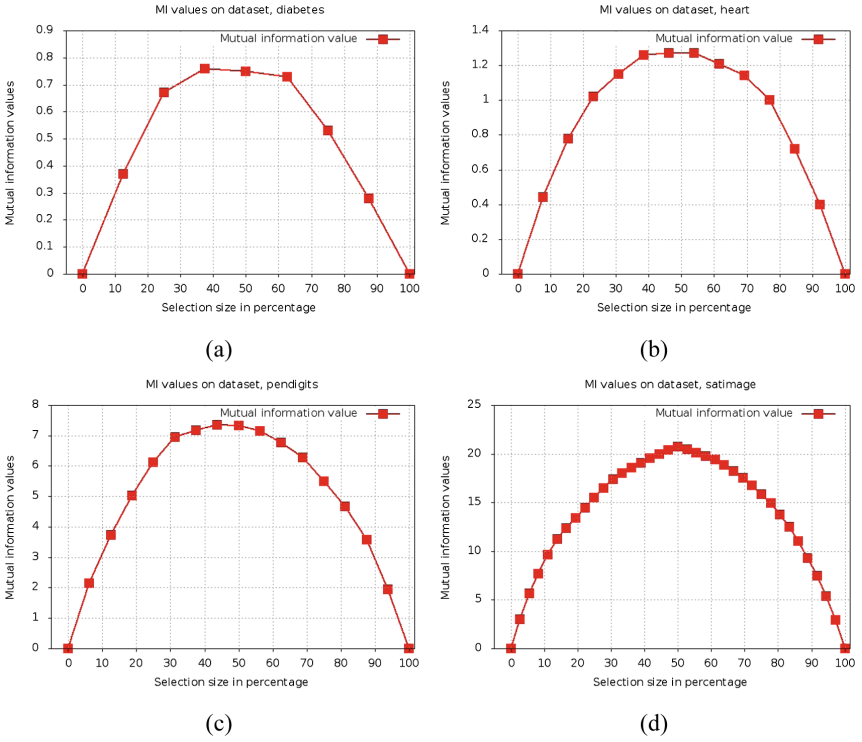

Fig. 2. Values of mutual information change with feature selection sizes

The experimental results disclose some guidelines about using the heuristic-based feature selection methods for classification problems. According to Figs. 1a, 3, and Table 4, the mutual-information-based feature selection heuristic will contribute most to classifiers when selecting out about a half of feature variable size when its selected subset scoring a maximal mutual information value. Whereas, as shown in Fig. 1d and Table 4, the entropy-based heuristic will most likely to help classifiers in the early stage of selection when choosing a relatively small percentage of features.

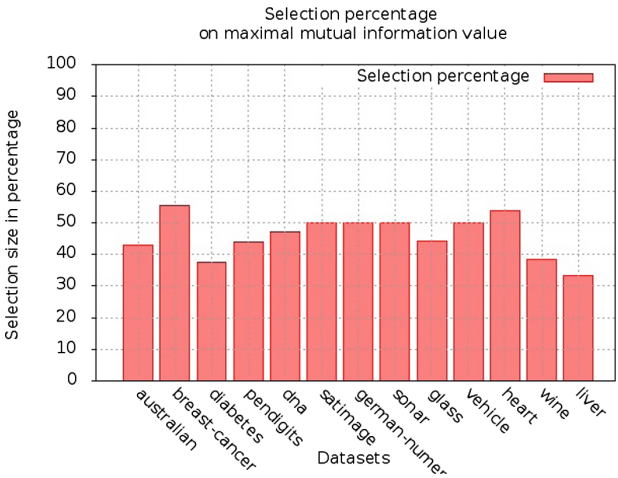

Fig. 3. Summary of feature selection sizes in percentage resulted in maximal mutual information gains for the datasets

Besides the observations above, we will also show a remarkable case for using the selection methods with classification on a medical dataset. It helps patients save their diagnostic costs. In the diabetes dataset for example, there are 8 features representing different diagnostic testings. Its individual costs are listed in Table 6. It looks like testing levels of the glucose and the insulin cost a lot more than the others.

Table 6. Costs of diagnostic testings in the diabetes dataset

Feature ID	Testing	Cost ($)
1	Times_pregnant	1.00
2	Glucose_tol	17.61
3	Diastolic_pb	1.00
4	Triceps	1.00
5	Insulin	22.78
6	Mass_index	1.00
7	Pedigree	1.00
8	Age	1.00

Table 7 sums up the result of how much it can save by selecting features against using full feature set for the classification of diabetes. It shows that the mutual-information-based selection method, which is a filter-based approach, achieves the same classification accuracy as using full features, in the meantime saving more than 50% of the diagnostic costs. It is remarkable because that it not only reduces patients' costs, but also can potentially help doctors improve the diagnostic accuracy.

Table 7. Cost savings by applying selection methods for classification of diabetes

Method	Selection size	Feature index	Accuracy	Classifier ID	Cost($)	Saving
Full-feature	100%	1 2 3 4 5 6 7 8	0.72	8	46.39	0%
Weka-ranker	37.5%	2 8 1	0.74	9	19.61	58%
MI-based	50%	4 8 2 7	0.72	8	20.61	56%
Entropy-based	62.5%	5 2 3 4 8	0.71	2	43.39	6%
RRQR	62.5%	5 2 3 4 8	0.71	2	43.39	6%

4 Summary

We introduced two heuristic-based feature selection methods, and explored their performance under classification problems for a number of datasets. Experimental results showed that feature selection helped reduce the dimensionality of the problems by improving classification accuracies with less number of features. It also showed that the mutual-information-based heuristic would contribute most to classifiers when selecting about a half size of features, while the entropy-based heuristic would most likely help in the early stage of the selection when choosing a relatively small percentage of features. We also demonstrated a remarkable case of feature selection for classification on a medical dataset.

Acknowledgement. This work was generously supported by the following funds: Hainan University's Scientific Research Start-Up Fund; Ministry of Education of China's Scientific Research Fund for the Returned Overseas Chinese Scholars; Hainan Province Natural Science Fund No. 20156243; China's Natural Science Fund Nos. 11401146, 11471135, 61462022, 61562017, 61562018, 61562019; Hainan Province's Major Science and Technology Project Grant No. ZDKJ2016015; Hainan Province's Key Research and Development Program Grant Nos. ZDYF2017010 and ZDYF2017128. This work was also supported by the State Key Laboratory of Marine Resource Utilization in the South China Sea, Hainan University.

References

1. Manning, C.D., Prabhakar Raghavan, H.S.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2009)
2. Das, S.: Filters, wrappers and a boosting-based hybrid for feature selection. In: Proceedings of the eighteenth international conference on machine learning. pp. 74–81. Morgan Kaufmann Publishers Inc., San Francisco (2001)
3. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. **97**, 273–324 (1997)
4. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)
5. Yu, L., Liu, H.: Feature selection for high-dimensional data: a fast correlation-based filter solution. In: Fawcett, T., Mishra, N. (eds.) ICML, pp. 856–863. AAAI Press (2003)

6. Liu, H., Motoda, H.: Computational Methods of Feature Selection (Chapman & Hall/CRC data mining and knowledge discovery series). Chapman & Hall/CRC (2007)
7. George, E.I.: The variable selection problem. J. Amer. Statist. Assoc. **95**, 1304–1308 (1999)
8. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York (2009)
9. Ipsen, I.C.F., Kelley, C.T.: Rank-deficient nonlinear least squares problems and subset selection. SIAM J. Numer. Anal. **49**, 1244–1266 (2011)
10. Gu, M., Eisenstat, S.C.: Efficient algorithms for computing a strong rank-revealing QR factorization. SIAM J. Sci. Comput. **17**, 848–869 (1996)
11. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in gaussian processes: theory, efficient algorithms and empirical studies. J. Mach. Learn. Res. **9**, 235–284 (2008)
12. Iwata, S., Fleischer, L., Fujishige, S.: A combinatorial strongly polynomial algorithm for minimizing submodular functions. J. ACM **48**, 761–777 (2001)
13. Krause, A., Guestrin, C.: Near-optimal nonmyopic value of information in graphical models. In: Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05), pp. 324–331. AUAI Press, Arlington, Virginia (2005)
14. Krause, A., McMahan, B., Guestrin, C., Gupta, A.: Robust submodular observation selection. J. Mach. Learn. Res. **9**, 2761–2801 (2008)
15. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in gaussian processes: theory, efficient algorithms and empirical studies. J. Mach. Learn. Res. **9**, 235–284 (2008)
16. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, Cambridge, Massachusetts (2006)
17. Caselton, W., Zidek, J.: Optimal monitoring network designs. Stat. Prob. Lett. **2**(4), 223–227 (1984)
18. Nemhauser, G., Wolsey, L., Fisher, M.: An analysis of the approximations for maximizing submodular set functions. Math. Program. **14**, 265–294 (1978)
19. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. **11**, 10–18 (2009)
20. Bischof, C.H., Quintana-Ortí, G.: Computing rank-revealing QR factorizations of dense matrices. ACM Trans. Math. Softw. **24**, 226–253 (1998)
21. Frank, A., Asuncion, A.: UCI machine learning repository. http://archive.ics.uci.edu/ml (2010)
22. Chang, C.C., Lin, C.J.: LIBSVM data: classification, regression, and multi-label. http://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/
23. University of Waikato, M.L.G. at: Weka 3: data mining software in java. http://www.cs.waikato.ac.nz/∼ml/weka/index.html
24. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques, Third edn. Morgan Kaufmann, (2011)