ORIGINAL RESEARCH

Supervised feature selection: A tutorial

Samuel H. Huang*

Department of Mechanical and Materials Engineering, University of Cincinnati, Cincinnati, United States

Received: January 6, 2015 **Accepted:** March 9, 2015 **Online Published:** April 9, 2015
DOI: 10.5430/air.v4n2p22 **URL:** http://dx.doi.org/10.5430/air.v4n2p22

Abstract

Supervised feature selection research has a long history. Its popularity exploded in the past 30 years due to the advance of information technology and the need to analyze high-dimensional data sets. Research papers published during these years were mostly from the machine learning and artificial intelligence community. The emphasis was largely on improving model accuracy using empirical methods; whereas the issue of feature relevance was somewhat overlooked. Feature selection methods were loosely classified as filters, wrappers, and embedded methods with little attention paid to their intricate details. This paper provides a tutorial of supervised feature selection, on the basis of reviewing frequently cited papers in this area and a number of classical publications from the statistics community. The objective of feature selection (either to improve model predictive accuracy or to determine relevance for hypothesis generation) is presented and discussed in details. Various supervised feature selection methods are classified using a detailed taxonomy. Guidelines for using feature selection methods in practice are provided based on a comprehensive review of the performance of these methods. Issues that require further attention are also discussed.

Key Words: Feature selection, Relevance, Redundancy, Filters, Wrappers, Embedded methods

1 Introduction

Feature selection, or variable selection, is an important research subject in the general area of learning from data. It arises from the need of determining the "best" subset of variables for prediction. Depending on the type of data, feature selection can be classified as supervised, semi-supervised, and unsupervised. A data instance (e.g., a patient potentially having cancer) is characterized by a number of independent variables (features), e.g., tumor markers (substances found in the blood, urine, stool, other bodily fluids, or tissues of the patient). It may also have a response variable (often called a label), e.g., whether the patient has a benign or a malignant tumor. If all the data instances in the data set have known response values, the process of feature selection is called "supervised". If some data instances have known response values and the others do not, we are facing a semi-supervised feature selection problem. If none of the data instances have response values, the feature selection performed is call "unsupervised".

The majority of research efforts are in the area of supervised feature selection. Although recent activities focus on classification, the problem originated from regression. According to M. Stone when discussing a paper[1] presented by A. J. Miller to the Royal Statistical Society, R. A. Fisher posed the problem of variable selection for regression in 1924. Progress has been made in the 1940s with limited computing power available at that time. The rationale for variable selection can be found in Hotelling.[2] The paper also shed light on earlier approaches to solve this problem. Research in this area gained substantial momentum starting in the early 1960s due to increased computing power. The majority of the early research work is carried out by statis-

*Correspondence: Samuel H. Huang; Email: sam.huang@uc.edu; Address: Department of Mechanical and Materials Engineering, University of Cincinnati, Cincinnati, OH 45221, United States.

ticians and focuses on linear regression. A literature review on variable selection for linear regression was conducted by Hocking.[3] Since then, variable selection research has been expanded to cover classification and clustering problems. It attracted a diverse array of researchers from artificial intelligence, machine learning, and data mining. As a result, the term "variable selection" is replaced over time by the term "feature selection".

Over the past 20 years, a number of well-written review papers on feature selection have been published.[4–9] However, these papers did not systematically discuss some issues that are importance to beginners in this area, especially those who are primarily interested in applications. Some researchers advocated the selection of a feature subset that leads to the highest model accuracy; whereas others argued that the best feature subset is one that included the most relevant and least redundant features. Are these two viewpoints competitive or complementary? In a practical application, should one strive to find a single feature subset that leads to the best model accuracy or trying to find multiple feature subsets for further consideration? Consider the case of identifying factors to assess the risk of patients who are susceptible to a certain disease. This problem is treated by some as selecting a subset of features to classify patients into a high-risk group and a low-risk group. Suppose a feature subset consists of age and gender resulted in a classifier with the highest accuracy, say 74%; whereas another feature subset consists of age and body mass index (BMI) resulted in a classifier with a slightly lower accuracy, say 71%. The two factors in the first feature subset are both non-modifiable, whereas BMI in the second subset is modifiable. Therefore, a clinician may be more interested in the second subset because she can advise a high-risk patient to take action, i.e., change BMI through diet or exercise, in order to reduce the risk of succumbing to the disease.

From the literature review, it appears that the objective of finding a single feature subset that can produce a model with the highest accuracy when evaluated using available data is overly emphasized in current feature selection research. Different applications may call for different objectives, which in term require different approaches for feature selection. In addition, the commonly accepted practice of classifying feature selection methods as belonging to filter, wrapper, and embedded methods did not adequately discern the characteristic of a particular method. This paper aims to provide a detailed description of fundamental issues encountered in feature selection including feature relevance, redundancy, the characteristics and performance of different feature selection methods, and guidelines for selecting appropriate methods for specific applications.

This paper is organized as follows. Section 2 discusses the objective of feature selection. Section 3 investigates the issue of feature relevance and looks into the relationship between feature relevance and model accuracy. Section 4 discusses the notion of feature redundancy. Section 5 explores the issue of identifying an optimal feature subset. Section 6 presents the taxonomy of feature selection methods. Section 7 presents guideline for using feature selection methods in practical applications. Finally, a summary is provided in Section 8. In addition, supplementary information is included in Section 9.

2 Objective of feature selection

It is generally understood that the goal of feature selection is to determine the "best" subsets of features (or variables) for conducting statistical analysis or building a machine learning model. However, the problem of feature selection is not well defined. This fact was acknowledged by Hocking[3] almost 40 years ago, when the author stated that "it is apparent that there is not a single problem, but rather several problems for which different answers might be appropriate." However, Hocking did not attempt to provide specific answers in his paper. To determine if a consensus on the objective of feature selection has been reached over the past decades, we decide to find the most influential feature selection papers and see how they define the problem of feature selection. The method for finding these papers can be found in Section 9.1.

Saeys et al.[4] stated that "the objectives of feature selection are manifold, the most important ones being: (a) to avoid overfitting and improve model performance, . . . (b) to provide faster and more cost-effective models and (c) to gain a deeper insight into the underlying processes that generated the data." Objective (a) focuses on model accuracy; objective (b) is accomplished by selecting a small subset of features; whereas objective (c) can be interpreted as focusing on feature relevance. It appeared that model accuracy is of the utmost importance to the majority of feature selection researchers. In the most influential paper on the wrapper approach for feature selection, Kohavi & John[8] stated that the task of feature selection is to find a subset of features such that ". . . san induction algorithm that is run on data containing only these features generates a classifier with the highest possible accuracy." This is concurred by Jain & Zongker,[10] where the authors stated that the goal is to select a feature subset that ". . . performs the best under some classification system." Pudil et al.,[11] not exclusively focusing on classification problems and aiming to find the best feature subset with a predefined cardinality, stated that the main goal is to select a feature subset ". . . without significantly degrading the performance of the recognition system."

Other researchers have a different emphasis, i.e., "focusing on the most relevant features for use in representing the data".[7] Perhaps realizing that there are different foci in feature selection, Liu & Yu[12] offered a more general definition of feature selection as "a process that selects a subset of original features. The optimality of a feature subset is measured by an evaluation criterion." Peng et al.,[13] being more specific, stated that "the optimal characterization condition often means the minimal classification error." The authors also indicated that if a classifier is not specified, "... minimal error usually requires the maximal statistical dependency..." and the task becomes "... selecting the features with the highest relevance to the target class..." Collins et al.[14] appeared to believe that model accuracy and feature relevance are two sides of the same coin, when they stated that feature selection "... can improve classification performance by discarding irrelevant or redundant features." However, Kohavi & John[8] indicated that these two foci were not equivalent and provided several demonstration examples. In the next section, we will show that this counterintuitive claim is caused by the way feature relevance is defined and the narrow focus on wrappers for feature selection.

3 Feature relevance

The notion of relevance was first studied in the philosophy literature. The focus was on formalizing a concept of relevance that would fit its commonsense notion. Keynes[15] provided a simple definition of *irrelevance* as follows: "h_1 is irrelevant to x on evidence h, if the probability of x on evidence hh_1 is the same as its probability on evidence h." Thus, the notion of relevance can be defined as:

h_1 is relevant to x on evidence h if h_1 is not irrelevant to x on h

Gardenfors,[16] focusing on the relevance between two sentences, proposed six logical conditions to be fulfilled by an appropriate definition of the relevance relation. Other researchers have offered domain specific characterization of relevance. For example, the use of conditional independence to define irrelevance in belief networks.[17] In the domain of feature selection, a number of researchers have provided varying definitions of relevance.[18–20] Kohavi & John[8] discussed these definitions and used an example to show that they gave unexpected results. The authors argued that there was a need to distinguish *strong relevance* and *weak relevance* in feature selection. Strong relevance means a feature cannot be removed without loss of predictive accuracy; whereas weak relevance means a feature can sometimes contribute to predictive accuracy. They went on to provide two additional examples, one intended to show that relevance did not imply optimality and the other optimality did not imply relevance. These examples were used to support the authors' claim that the objectives of maximizing model accuracy and identifying relevant features are not equivalent.

There are two reasons why Kohavi & John made such a counterintuitive claim. First, it has to do with the way feature relevance is defined. Second, it is because the authors focused on model-dependent feature subset selection (wrappers). It is self-evident that the latter two examples rely on the premise that a specific form of learning model has been chosen (a monomial and a limited perceptron, respectively), yet the functional form of the model is incorrect for the concept to be learned. In fact, these two examples are ill-suited to illustrate the relationship between feature relevance and model accuracy. Rather, there are testimonies of the danger of using wrappers for feature selection.

It is believed that if the common sense notion of relevance as defined by Keynes[15] is faithfully adopted in feature selection then there are no conflicts between the objectives of maximizing model accuracy and identifying relevant features. We first extend Keynes' definition of relevance to a set of features. Let $\mathbf{F} = \{X_1, X_2, \cdots X_n\}$ denote the set of features and Y denote the target concept to be learned. Let $\mathbf{S} \subset \mathbf{F}$ (i.e., $\mathbf{S}$ is a subset of $\mathbf{F}$) and $\mathbf{s}$ be a vector of values assignment to all features in $\mathbf{S}$. Then

Feature X_i is relevant to Y given $\mathbf{S}(X_i \notin \mathbf{S})$ iff there exists some x_i, y, and $\mathbf{s}$ for which $P(\mathbf{S} = \mathrm{s}, X_i = x_i) > 0$ such that $P(Y = y|\mathbf{S} = \mathbf{s}, X_i = x_i) \neq P(Y = y|\mathbf{S} = \mathbf{s})$.

Note that the difference between this definition and definition 5 in Kohavi & John (1997) is the elements in $\mathbf{S}$. In Ref.,[8] $\mathbf{S}$ is strictly defined as the set of all features except X_i, i.e., $\mathbf{S} = \mathbf{F} - X_i$. As a result, $\mathbf{S}$ is determined for each feature X_i under consideration. Here, $\mathbf{S}$ could be any subset of $\mathbf{F}$ that does not contain X_i. This is in the spirit of the traditional definition of relevance by Keynes[15] where no relationship was required between hypothesis h_1 (equivalent to feature X_i) and evidence h (equivalent to feature subset $\mathbf{S}$). The above definition is called the *common sense definition of feature relevance*.

Now let us revisit example 1 in Ref.[8] to see how the notion of feature relevance defined here holds up. The example, called *correlated XOR*, has 5 Boolean features, X_1, X_2, X_3, X_4, X_5, where $X_4 = \overline{X_2}, X_5 = \overline{X_3}$. The target concept is $Y = X_1 \bigoplus X_2$. Note that this is equivalent to $Y = X_1 \bigoplus \overline{X_4}$. The truth table for this example problem is shown in Table 1. Kohavi and John showed that all previous definitions of feature relevance cannot produce reasonable results (because they did not follow Keynes's definition of relevance). There are three different outcomes: (1) only X_1 is relevant, (2) no features are relevant, and (3) all features are relevant. This example is used to justify the creation of the definitions of strong relevance and weak relevance. Feature X_1 is strongly relevant, features X_2 and X_4 are weakly relevant, whereas features X_3 and X_5 are irrelevant.

Note that using the common sense definition of feature relevance, one has to determine $\mathbf{S}$. Without prior knowledge there are two nature ways to define $\mathbf{S}$ for initial investigation: (1) $\mathbf{S} = \phi$, and (2) $\mathbf{S} = \mathbf{F} - \{X_i\}$. Note that this corresponds to sequential forward search (SFS) and sequential backward search (SBS) for feature subset selection, respectively. Given $\mathbf{S} = \phi$, none of the features are relevant to Y.

There is a logical interpretation – adding a single feature to a model without any other features will not improve its accuracy to predict the target concept Y. Given $\mathbf{S} = \mathbf{F} - \{X_i\}$, only feature X_1 is relevant to Y. Again, the interpretation is logical – remove X_1 from a model with all the features will lead to a lower predictive accuracy. In other words, feature X_1 is indispensible. Knowing this, we can set $\mathbf{S} = \{X_1\}$. Given $\mathbf{S} = \{X_1\}$, we find that X_2 is relevant and so does X_4. However, given $\mathbf{S} = \{X_1, X_2\}$ none of the remaining features are relevant; this is also true given $\mathbf{S} = \{X_1, X_4\}$. We can thus draw the conclusion that there are two sets of features that can be used to build a model with the highest accuracy; namely $\{X_1, X_2\}, \{X_1, X_4\}$.

Table 1: Truth table for the correlated XOR problem

X_1	X_2	X_3	X_4	X_5	Y
0	1	1	0	0	0
0	1	0	0	1	0
0	0	1	1	0	1
0	0	0	1	1	1
1	1	1	0	0	1
1	1	0	0	1	1
1	0	1	1	0	0
1	0	0	1	1	0

Granted, the notions of strong relevance and weak relevance make sense. However, it is unnecessary to create these two notions for the purpose of reconciling the objectives of maximizing model accuracy and identifying relevant features in feature selection. Rather, it is sufficient to use the common sense definition of feature relevance, where the relevance of a feature is conditioned on the evidence of a feature subset under consideration. This definition also has interesting implications on search strategies for feature subset selection, which will be discussed in Section 5. For now, the notion of feature redundancy will be discussed.

4 Feature redundancy

In addition to feature relevance, feature redundancy is another term that is often encountered in the feature selection literature. There is a popular saying in feature selection that "the m best features are not the best m features" because of feature redundancy, which led to the minimum-redundancy-maximal-relevance framework for feature selection.[13] Feature redundancy is generally understood in terms of feature dependency (or feature correlation). It is widely accepted that two perfectly correlated features are redundant to each other because adding one feature on top of the other will not provide additional information; and hence, will not improve model accuracy. Redundancy may also exist between two independent (uncorrelated) features in the sense that the two best independent features are not the best two features.[21] On the other hand, Guyon & Elisseeff[5] used an example to demonstrate that noise reduction may be obtained by using features that are independent and are presumably redundant. The authors used another example to show that two highly correlated features (with negative correlation), when combined, improved classification accuracy than using any single feature. Therefore, feature correlation cannot be equated to feature redundancy.

Yu & Liu[22] provided a formal definition of feature redundancy using the definition of a feature's Markov blanket by Koller & Sahami[23] and the notion of weakly relevant features. Using the notations in Section 3, a feature subset $\mathbf{S}$ is a Markov blanket for $X_i (X_i \notin \mathbf{S})$ if X_i is conditionally independent of $F - \mathbf{S} - \{X_i\}$. Essentially, $\mathbf{S}$ subsumes not only the information that X_i has about the target concept, but also about all of the other features. A weakly relevant feature is redundant, given a feature subset $\mathbf{S}$, iff it has a Markov blanket within $\mathbf{S}$. Based on this definition, the authors developed a redundancy based filter (RBF), which is an approximate algorithm, for feature selection. All features are heuristically treated as relevant (on the evidence that $\mathbf{S} = $ ‰ in the common sense definition of feature relevance). RBF is then used to remove redundant features.

Yu & Liu's formal definition of feature redundancy is conditioned on a feature subset, meaning a redundant feature cannot be determined in absolute terms. Note that the common sense definition of feature relevance is also conditioned on a feature subset. Therefore, to simplify the problem, it is not necessary to distinguish a redundant feature from an irrelevant feature. After all, the purpose of identifying redundant features is to remove them, which is the same as the purpose of identifying irrelevant features.

5 Optimal feature subset

From the previous discussion, it is clear that the objective of feature selection can be defined as finding a feature subset $\mathbf{S}$ such that no features in $\mathbf{F} - \mathbf{S}$ are relevant to Y (note that the common sense definition of feature relevance is used here). In other words, all the features in the subset are relevant to the target concept, which implies that the feature subset has a causal relationship with the target concept. Based on this definition, it is possible to have multiple optimal feature subsets. Refer to the correlated XOR example, there are two optimal feature subsets; namely, $\{X_1, X_2\}$ and $\{X_1, X_4\}$. It is also possible that there exist multiple optimal feature subsets with different cardinality, in which case one should strive to find the subset with the lowest cardinality (the principle of parsimony). One may also want to identify all optimal feature subsets if the objective is not merely prediction but also include intervention, i.e., changing the values of some features to get a different outcome (the cost of manipulating different features may be very different).

Intuitively, there are two ways of searching for an optimal feature subset; namely, starts from an empty set, or starts from the full set. We have already seen that when starting from an empty set, it is possible that no features are found to be relevant based on the common sense definition of feature relevance. In this case, no information is gained regarding the optimal feature subset. On the other hand, when starting from the full set and if no features are found to be relevant, we will know that the optimal feature subset has a cardinality of less than n. In this case, we can randomly remove a feature and continue the search. The question is whether such a backward elimination process can lead us to the optimal feature subset. Let $\mathbf{S}$ be the feature subset under consideration, and $\mathbf{I}$ be the set of features within $\mathbf{S}(\mathbf{I} \subseteq \mathbf{S})$ that are found to be irrelevant. The question can be restated as "which feature(s) in $\mathbf{I}$ should be removed from $\mathbf{S}$ so that when no irrelevant features are found in $\mathbf{S}$,it is guaranteed that an optimal feature subset is found?"

Koller & Sahami[23] proved that if a feature $X_i(X_i \notin \mathbf{S})$ has a Markov blanket in $\mathbf{S}$, another feature $X_j(X_j \in \mathbf{S})$ also has a Markov blanket in $\mathbf{S}$, then X_i also has a Markov blanket in $\mathbf{S} - \{X_j\}$. This guarantees that a feature removed based on the Markov blanket criterion will not be needed in the optimal feature subset. If all the required probability distribution and conditional probability distribution functions are known, then a backward elimination process based on *Markov blanket filtering* will guarantee that an optimal feature subset can be found. However, there is no guarantee that an optimal feature subset found this way will have the lowest cardinality.

In real-world applications, probability distribution functions are unknown and have to be estimated from data. This problem, commonly known as density estimation, is among the most difficult problems in learning from data.[24] In a high-dimensional space with sparse data, the probability distribution functions cannot be accurately estimated. Therefore, alternative measures have to be used to approximately determine feature relevance. Commonly used measures are correlation coefficient and entropy. The attempt of using feature relevance to identify an optimal feature subset gives rise to the class of feature selection algorithms called *filters*.

Filters select features using a preprocessing step independent of any machine learning models (or induction algorithms). Kohavi & John[8] stated that "the main disadvantage of the filter approach is that it totally ignores the effects of the selected feature subset on the performance of the induction algorithm." The authors argued that the optimal feature subset depends on the specific biases and heuristics of the induction algorithm and advocated the use of wrappers, which rely on evaluating the predictive accuracy of a specific machine learning model for feature selection. As previously mentioned, two examples were given to illustrate the authors' point of view. One example (example 2) is the learning of a target concept $f(X_1, X_2, X_3) = (X_1 \wedge X_2) \vee X_3$ using a monomial, where X_1, X_2, X_3 are Boolean features. In this case, the optimal feature subset is $\{X_3\}$, which achieves a predictive accuracy of 87.5%. The other example (example 3) is the use of a limited perceptron for classification, which requires a dummy (irrelevant) feature that always take the value of 1 (equivalent to the use of an intercept term in regression).

Apparently, the monomial model in example 2 was not an appropriate model. Example 3 is a bit of a stretch. It can be easily argued that a constant-valued feature is not really a feature because it has no effect on the variation of the target concept. In addition, why use a limited perceptron when a regular perceptron (which has a building-in dummy input) can be used? Kohavi & John[8] conceded that "we believe that cases such as those depicted in example 3 are rare in practice and that irrelevant features should generally be removed." In fact, one may use these two examples to argue against wrappers for feature selection when the wrong machine learning model is used. On the other hand, if the goal is to identify an optimal feature subset with respect to a certain machine learning model then it would be logical to conclude that wrappers are a better choice than filters for feature selection. The question is whether a wrapper can guarantee the finding of an optimal feature subset.

Wrappers require a search strategy to explore feature subsets and an evaluation function to measure the goodness of a subset. Cross-validation accuracy is commonly used as the evaluation function. There is consensus that finding the optimal feature subset is a combinatorial problem, which requires a complete search strategy in order to guarantee that an optimal feature subset is found. Note that a complete search does not mean an exhaustive search, for example, the branch and bound algorithm for finding the best subset of m features.[25] However, a brand and bound search requires that the evaluation function is monotonic. In addition, it still has a high computation complexity and is impractical for problems involving more than 30 features.[11] Therefore, in real-world applications other non-complete search strategies must be used. Readers are referred to Ref.[9] for a summary of feature selection methods based on three types of search strategies; namely, complete, heuristic, and random. Heuristic and random search strategies, combined with different evaluation functions, have been shown to produce good results in a reasonable amount of time.[8, 11, 26]

Suppose a complete search is feasible, the ability of a wrapper to find an optimal feature subset still requires the following idealized setting:[3]

"(a) the analyst has data on a large number of potential variables which include all relevant variables and appropriate functions of them plus, possibly, some other extraneous variables and variable functions and (b) the analyst has available "good" data on which to base the eventual conclusions."

J. B. Copas provided a theoretical analysis of the probability of finding the correct subset of p features (with a causal relationship with the response variable) in linear least square regression.[1] The analysis showed that when the population multiple correlation R is 1 (indicating a perfect model fit – the model is correct and the data is noise-free) then the optimal feature subset can always be found. When R is high (e.g., R = 0.8) and the cardinality of the optimal subset is low (e.g., p = 5) there is still a fairly high probability (close to 90%) of finding the optimal feature subset irrespective of the number of total features. However, when p increases (e.g., p = 10) the probability of finding the optimal feature subset decreases quickly as the number of total features increases even when R is high. When R is small, "the situation is hopeless, with the selected subset almost certainly being wrong."

We can infer form the analysis by Copas that the wrapper approach for feature selection can produce good results only when (1) all relevant features are available and their number is small, (2) the correct (or suitable) machine learning model is used, and (3) the data contains little noise. These conditions are not easy to satisfy in real-world applications. Therefore, Copas is skeptical of the practical value of feature subset selection based on model accuracy and professed that *"It has been said: 'If you torture the data for long enough, in the end they will confess.' Errors of grammar apart, what more brutal torture can there be than subset selection? The data will always confess, and the confession will usually be wrong."* The skepticism is echoed by R. L. Plackett, who stated that "If variable elimination has not been sorted out after two decades of work assisted by high-speed computing, then perhaps the time has come to move on to other problems."[1]

6 Feature selection methods

In the mid-1980s, a third class of feature selection algorithms, called embedded methods, emerged. While filters select features independent of any machine learning models and wrappers wrap the feature selection process around a specific model, embedded methods incorporate feature selection as part of the process in building a specific model. The broad classification of feature selection methods into filters, wrappers, and embedded is commonly accepted in the literature. However, this classification does not provide a detailed enough reference framework and the designation of filters and wrappers could cause confusion. For example, filters are often said to be computationally more efficient than wrappers, which is not always true. Saeys et al.[4] distinguishes two types of filters; namely, univariate and multivariate. Univariate filters are basically feature ranking methods that conduct a pair-wise analysis of the dependency between each feature and the target concept. Features with dependency measures (correlation coefficient, information gain, etc.) below a certain threshold are eliminated. This type of filters has $O(n)$ computation complexity and is certainly efficient. On the other hand, multivariate filters such as FOCUS[27] uses an exhaustive search strategy and can hardly be said to be computationally efficient. Another example is that wrappers are often said to be able to find feature subsets that lead to better model accuracy compared to filters. However, multivariate filters that use consistency measure combined with an appropriate search strategy also produced feature subsets that can be used to build models with very high accuracy.[26]

One may notice that the only difference between a consistency measure-based filter and a wrapper is the evaluation function. While inconsistency is a model-independent evaluation function, a wrapper uses cross-validation accuracy of a specific machine learning model as the evaluation function. Model-independent evaluation functions can be combined with any search strategies for feature subset selection. Such filters are more similar to wrappers than to univariate filters. In fact, Dash & Liu[9] classifies feature selection methods into 15 categories based on the combination of 5 types of evaluation measures (distance measure, information measure, dependency measure, consistency measure, and classification error rate) and 3 types of search strategies (heuristic, complete, and random).

A more detailed taxonomy of feature selection methods is depicted in Figure 1. First, one should distinguish feature selection methods based on their outcomes. The outcomes of feature ranking methods are the degrees of dependency of individual features with respect to the target concept. The outcomes of subset selection methods are feature subsets that are relevant to the target concept. The outcomes of embedded methods are predictive models built using certain feature subsets. Note that the objective of subset selection methods is to identify a feature subset. These methods may or may not produce a specific model associated with the feature subset. On the other hand, the objective of embedded methods is to produce a predictive model. Features remain in the model are a byproduct of the modeling process.

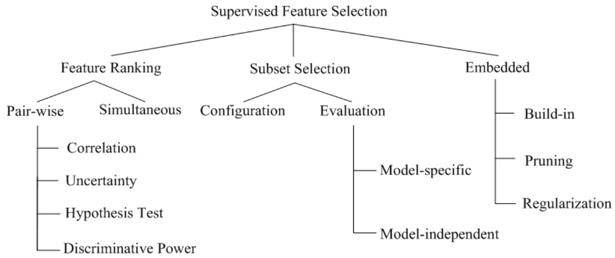

Figure 1: Taxonomy of feature selection methods

Feature ranking methods can be classified into two categories: (1) those that are based on pair-wise dependency analysis of individual features, and (2) those that simultaneously rank all the features. Pair-wise ranking methods evaluate the degree of dependency between each feature and the target concept, one feature at a time. A number of different criteria have been used to conduct pair-wise dependency evaluation. These criteria are classified into four categories: (1) *correlation*, (2) *uncertainty*, (3) *hypothesis test*, and (4) *discriminative power*. Criteria in the first category are based on the measures of correlation between the feature and the target concept, e.g., Pearson's product-moment correlation coefficient.[28] Criteria in the second category are based on uncertainty measures used in information theory, i.e., entropy. The most well-known criterion is probably information gain,[29] which measures the reduction of uncertainty about the target concept when the value of a feature is known. Note that uncertainty-based criteria are applicable only to discrete features and target concepts. Continuous features need to be discretized in order to apply these criteria. Criteria in the third category are based on statistical hypothesis test, in which the p-values of the resulting tests are used to rank features. They are generally applied to discrete target concepts; whereas the features can either be discrete or continuous. A typical example is the use of Chi-squared test to determine if the target class is independent of a discrete feature. Another example is applying t-test to determine if the target class is independent of a continuous feature by comparing the difference in means of samples that are associated with different class labels. The forth category covers criteria that evaluate the discriminative power of a feature. Examples include the use of single-variable models to determine the error or area under the ROC (receiver operating characteristic) curve resulted from each feature.

Pair-wise feature ranking methods have the lowest computational complexity and are suitable for data preprocessing. In certain applications, such as whole-genome SNP (single-nucleotide polymorphisms) association studies that involve several hundred thousand features, these methods are commonly used to reduce the number of features to a more manageable size before other feature selection methods are applied. However, the nature of pair-wise feature ranking methods prevents them from identifying interacting features that as a group can be used to predict the target concept, but individually have no detectable dependency relationship with the target concept, e.g., XOR type problems. Simultaneous feature ranking methods have the potential to overcome this problem. These methods, exemplified by the Relief algorithm,[30] calculate relevance weights for all the features at the same time by looking into their joint relationship with the target concept. The Relief family of algorithms, i.e., ReliefF for dealing with multiclass problems[31] and RReliefF for dealing with regression problems,[32] have been found to be effective in detecting conditional independencies.[33] Note that feature ranking methods based on principal component analysis[34] may be viewed as simultaneous feature ranking methods. However, their ability to identify interacting features has yet to be systematically studied.

Feature subset selection methods can also be classified into two categories; namely, those that exploit correlations among the features and those that do not. Belonging to the first category are subset configuration methods that analyze both feature-target concept correlation and feature-feature correlation in order to configure a feature subset. The result is a feature subset without any indication of achievable model predictive accuracy. Representative methods are Markov blanket filter,[23] correlation-based filter,[35] fast correlation-based filter,[22] and minimum-redundancy-maximal-relevance framework.[13] Note that the max-min method[36] also belongs to this category, but it was found to suffer from serious drawbacks.[37]

The second category covers subset evaluation methods that determine the goodness of feature subsets in terms of their achievable model predictive accuracy in order to identify the best subset. The evaluation criteria can be classified as model-specific and model-independent. Model-specific subset evaluation methods are equivalent to wrappers. These methods use the cross-validation accuracy of a specific machine learning model as the evaluation criterion and produce a predictive model along with the selected feature subset. On the other hand, model-independent subset evaluation methods use criteria derived based on the characteristics of the data and do not produce a predictive model. These criteria include inconsistency rate,[26] inference correlation,[38] and minimum expected cost of misclassification.[39] Note that the classification quality criterion used in rough set based feature selection[40–42] is equivalent to inconsistency rate. Feature subsets selected using model-independent evaluation methods are said to be unbiased toward a particular model, which enables them to be used by a variety of machine learning algorithms. However, it may be more appropriate to use an instance-based learning algorithm (such as k-nearest neighbor) because these methods favor a smaller subset of features with complicated function over a larger subset admitting simple rules.[26]

Both subset configuration and subset evaluation methods require a certain search strategy. Subset configuration methods exploit feature-target concept correlation and feature-feature correlation to guide the search process. The search is typically more efficient than subset evaluation methods that in theory need to explore the entire 2n feature subset space. However, because no indication of achievable model predictive accuracy is obtained, the selected feature subset needs to be passed to a machine learning algorithm for final evaluation.[43] An alternative is to create a number of sequential feature subsets and then use a particular machine learning algorithm to evaluate these subsets in order to identify the smallest subset with the highest prediction accuracy.[13]

For subset evaluation methods, Liu & Motoda[44] provided a comprehensive summary of different search strategies that were classified as complete search, heuristic search, and nondeterministic search. Note that the feature subset space can be arranged as a graph with 2n nodes, where each node represents a subset and an edge represents the containment relationship that adds or subtracts a feature.[7,8] Therefore, standard graph search techniques can be used to explore the feature subset space. To completely search through the entire feature subset space, breadth-first search or depth-first search can be used. If the evaluation function is monotonic, then the brand-and-bound algorithm[25] can be used to avoid exhaustive search while maintaining the ability to find the optimal (with respect to the evaluation function) feature subset. An improved version of the algorithm was developed by Yu & Yuan,[45] which is able to skip some unnecessary searches. It has been demonstrated that when the evaluation function is non-monotonic, branch and bound based feature selection still provided good results.[46] However, as previously mentioned, complete search strategies have a high computational complexity and is impractical for problems involving a large number of features.

Unlike time-consuming complete search strategies, heuristic search strategies trade off optimality for search efficiency. There are many heuristic search strategies. Kohavi & John[8] found that best-first search works much better than hill-climbing in feature selection. Beam search, an extension of best-first search (or a limited version of breadth-first search), has also been used. Simpler heuristic search strategies include SFS (start from the empty set and add one feature at a time) and SBS (start from the full set and remove one feature at a time). These strategies are usually faster but suffer from the "nesting effect", i.e., an added feature can no longer be removed and a removed feature can no longer be added. To overcome this problem, the plus-l-minus-r search was proposed.[47] Pudil et al.[11] argued that there is no theoretical way to determine the values of l and r to find the best feature subset. The authors proposed the floating search strategy that dynamically changes the number of features added or removed at each step.

Heuristic search strategies are prone to get stuck in local minima. Nondeterministic search strategies are used to overcome this problem. Examples include genetic algorithms,[48,49] randomized hill-climbing,[50] Las Vegas algorithm,[51] estimation of distribution,[52] simulated annealing,[53] ant colony optimization,[54] and particle swarm optimization.[40] These search strategies are likely to produce different results in different runs because they use randomized initial conditions and/or randomized search parameters. Note that when using subset evaluation methods, one may also use an evaluation function that includes the minimization of the number of features. For example, Wang et al.[40] used the weighted sum of the classification quality and the number of features as an evaluation function.

Embedded methods can be further divided into three categories. The first are models with a build-in mechanism for feature selection, represented by decision tree models such as classification and regression tree,[55] ID3,[29] and C4.5.[56] The second are pruning methods that train a model with all features and then attempt to remove some of the features by setting the coefficients associated with these features to 0 while maintaining model performance. Examples include optimal brain damage (OBD) for deleting the weights of neural networks,[57] recursive feature elimination using support vector machine (SVM),[58] and nearest shrunken centroids for clustering.[59] The third are regularization models with objective functions that minimize fitting errors and in the meantime force the coefficients to be small. Features with coefficients that are close to 0 are then eliminated. Examples including the lasso (least absolute shrinkage and selection operator) that constrain the sum of the absolute values of the coefficients to be less than a user defined value[60] and models that add the l_p-norm of the coefficients as a term to be minimized simultaneous with the model fitting errors.[61] Note that embedded feature selection attempts to select features and determine model parameters simultaneously. This is a non-convex optimization problem and finding the globally optimal solution is difficult. Several methods aiming to improve the quality of the solution have been proposed recently.[62,63]

The majority of feature selection methods were developed based on heuristics. Recently, attempts have been made to develop theoretical basis to guide feature selection. Yang & Hu[64] established a theoretically optimal feature selection criterion called "discriminative optimal criterion." Song et al.[65] viewed feature selection as a dependency maximization problem and showed that several feature selection methods aimed to solve this problem. On the other hand, Brown et al.[66] viewed feature selection as a conditional likelihood maximization problem and showed that information theoretical feature selection methods are approximate iterative maximizers. Zhao et al.[67] observed that several feature selection methods implicitly measure sample similarity and proposed a unified "similarity preserving" framework for feature selection.

In the era of big data, the size of many data sets increases dynamically. Feature selection methods are typically based on a batch learning mode. When the size of a data set increases, these methods have to be applied repeatedly, which is time-consuming. To overcome this drawback, incremental feature selection methods have been developed.[68] Another characteristic of big data is missing data values. In other words, some data instances may not contain the full set of features. Feature selection methods for dynamically increasing and incomplete data sets have also been proposed.[69,70]

7 Guidelines for applying feature selection methods

For a novice practitioner in feature selection, the decision to be made is usually the selection of a particular feature selection method. The method is then applied to the entire dataset at hand. The features selected are then reported. If the feature selection method also produces a model (when wrappers or embedded methods are used) then the model accuracy is also reported; otherwise (when filters are used), the selected features are then used to build a model using the entire dataset and the model accuracy is reported. Such an approach often leads to inflated model accuracy and the selected features are almost always not optimal, unless the data sample size is large, the total number of features is small, and the model accuracy is close to 100%.

An experienced practitioner will use a cross-validation strategy when selecting features.[8] This strategy is commonly used for the purpose of model validation. It involves the partitioning of the data set at hand into k complementary subsets. Then, k-1 subsets (called training set) are used to select a feature subset. The model built using this selected subset is then tested on the remaining data subset (called testing set). The accuracy is recorded. This process is repeated k times on different testing sets. The averaged accuracy is reported as k-fold validation accuracy. Using this strategy, a more accurate estimation of predictive accuracy can be obtained. However, the problem is that the selected features in each round are not always the same. If the purpose of feature selection is to build a more robust predictor, this strategy is acceptable because one can use an ensemble of models with different features for prediction or use all the features selected in different rounds to build another model.

Some researchers believe that cross-validation may still produce biased estimation of model accuracy. Therefore, a dual-loop cross-validation strategy was proposed.[71] First, the entire data set is partitioned into m complementary subsets. Then, m-1 subsets are used for feature selection using cross-validation. A predictor is then built (as previously mentioned) and tested on the remaining subset (testing set). The accuracy is recorded. This process is repeated m times in different testing sets. Presumably this would give an unbiased estimation of model accuracy to better compare different feature selection methods.

For researchers who want to develop new feature selection methods, cross-validation and dual-loop cross-validation are a must to justify the effectiveness of the developed methods. However, for practitioners whose interest is to distinguish useful features from irrelevant features, the question is two-fold: (1) which method (among those discussed in Section 6) to use? and (2) should cross-validation and dual-loop cross-validation be used?

To answer these questions, a literature review on the performance of different feature selection methods were conducted (see Section 9.2). Apparently, not a single feature selection method is universally superior to the others. To determine which feature selection method to use and how to use it, one should first determine the purpose of feature selection and then study the characteristics of the data set at hand.

If one is more interested in the causality of features without explicit consideration of model predictive accuracy (e.g., studying risk factors for a certain disease) then feature ranking methods should be used. Further, if one is only interested in the main effect of a feature, then pair-wise ranking methods are the best; otherwise, simultaneous feature ranking methods should be used. Cross-validation and dual-loop cross-validation are not applicable to these methods. Under this circumstance, if one is overwhelmed by the number of features selected then feature subset configuration and model-independent feature subset evaluation methods should be used. We do not recommend using cross-validation (to avoid confusion) in conjunction with these methods because model accuracy is not a concern. Of course, the selected features will need to be studied further, either using randomized control trials or justified based on subject matter knowledge.

If one's purpose is to build a robust predictor and does not care much about feature causality, then model-dependent feature subset selection methods and embedded methods should be used. Cross-validation is a must for both methods when accessing model accuracy. For model-dependent feature selection methods, if the number of data samples is large (with respect to the total number of features) then dual-loop cross-validation should be used to provide a more unbiased estimate of model accuracy. Note that dual-loop cross-validation does not apply to embedded methods because no feature subsets were explicitly evaluated. Model-independent feature subset selection methods and feature subset configuration methods can also be used under this circumstance. They should be used on the entire dataset to select a particular feature subset. Different models can then be built using the selected feature subset and the accuracy accessed using cross-validation.

If one is both interested in feature causality and model accuracy, then the problem is more complicated. Refer to the analysis conducted by J. B. Copas in Section 5, unless the underlying data set is nearly noise free, all the causal features are included, and the total number of features is not too large, no feature selection methods would satisfy the need. The only solution is to identify multiple feature subsets (using feature subset selection methods or embedded methods) for subsequent evaluation using additional data sets. How many feature subsets should be identified remains an open research question.

Prior knowledge of the underlying data set can also be used to determine which feature selection method is appropriate. If it is known that the data set contains interacting features (features individually do not provide useful information but jointly can be used to determine the target concept, e.g., the XOR problem), then pair-wise feature ranking and subset feature configuration methods should not be used. If there are redundant features that one wish to exclude, then the one should use feature subset selection methods and embedded methods, because feature ranking methods are incapable of detecting redundant features. If there are correlated features, the situation is more complex. Again, feature ranking methods are not able to detect feature correlation and will always select all of these features if there are relevant to the target concept. One would think that feature subset configuration methods that explicitly consider feature correlation would be the best for such data sets. However, as pointed out by Freeman et al.,[72] such methods (e.g., minimum-redundancy-maximal-relevance) may not work well if the correlation between features is higher than the correlation between the feature and the target concept. Therefore, one should use feature subset evaluation and embedded methods.

8 Summary

The history of feature selection research can be traced back to R. A. Fisher in 1924. Early research works were conducted mostly by statisticians. These works typically involve statistical theories. In the past 30 years, a large number of researchers from machine learning and artificial intelligence communities developed a variety of feature selection methods based on information theory and heuristics. These methods were largely evaluated experimentally. The consensus was that no feature selection method is universally superior to others. This consensus certainly makes sense. However, it was derived based on empirical analysis that focused on model accuracy obtained through cross-validation. Reunanen[73] has shown that cross-validation might not be the best approach to evaluate model accuracy. It was suggested that independent test data sets should be used for evaluation. Although this suggestion was made over 10 years ago, it has yet to gain traction in the feature selection research community. One may speculate that should this very reasonable suggestion be widely adapted, conclusions from most published papers could have been very different. Specifically, it would not be unreasonable to suspect that the problem of selecting features that overfit the data is quite common. In fact, as discussed in Section 5, theoretical analysis by J. B. Copas showed that in most real-world settings, the probability of finding the right feature subset is very low.

One may doubt the conclusion drawn from J. B. Copas' analysis, based on the fact that in many real-world datasets feature selection did produce models with higher accuracy compared to models built using all the features. However, a model with high accuracy does not necessarily require that the features used are the right ones, especially when the accuracy is evaluated using cross-validation instead of using an independent testing data set. In fact, it is common that for the same data set, different researchers selected different feature subsets that achieved similar model accuracy. Because for these real-world datasets one do not know the right feature subsets, it is impossible to verify the ability of feature selection methods to uncover the casual relationship between features and the target concept. One has no choice but to use model accuracy (or generalization ability to be precise) as the performance measure. Under this circumstance, wrappers (model-specific subset evaluation methods) were recommended as the best choice if computation complexity is not an issue. However, few researchers compared the performance of wrappers with that of embedded methods, which use all the features during the model building process but employee various techniques to improve model generalization ability. It is very likely that embedded methods could perform on par with, if not better than wrappers, especially if an independent testing data set is used to evaluate model accuracy.

In many real-world applications, model accuracy is not the only concern. Rather, one may be more interested in identifying the causal relationship between features and the target concept, i.e., finding the right feature subset. Granted, without any subject matter knowledge and without conducting randomized control trials, one can only infer correlation, not causal relationship, from data. In fact, establishing causal relationship cannot be achieved using data analysis alone. Rather, data analysis can only be used for hypothesis testing or hypothesis generation. Feature selection is often used in analyzing data sets for hypothesis generation. Because the probability of finding the right feature subset is low, it would be prudent to identify multiple feature subsets as candidates for follow-up studies utilizing subject matter knowledge. In fact, many medical researchers prefer the use of pair-wise feature ranking methods based on hypothesis test and retain all individual features that are deemed statistically significant. These features are investigated further using a combination of subject matter knowledge and additional data collection. From the literature it is evident that feature selection researchers focus heavily on model accuracy and overlook the issue of hypothesis generation. In terms of feature selection for improved model accuracy, significant research progress has been made. However, there are two critical issues that need further attention:

- One should not overly rely on cross-validation to evaluate model accuracy. A more objective approach is to use an independent testing data set. item More attention should be paid to embedded methods for feature selection. When evaluating the usefulness of a selected feature subset, one should use models built using embedded methods as a benchmark for comparison.

In terms of feature selection for hypothesis generation, further research is needed in the following areas:

- Generation of multiple feature subsets for follow-up studies. It is very likely that a causal feature subset is not the best feature subset in terms of model accuracy. In addition, with multiple feature subsets redundant features can be retained, which may serve as useful alternatives for decision making.

- More emphasis should be placed on feature ranking methods. These methods identify a prioritized list of features that is conducive to follow-up studies. Simultaneous feature ranking is a promising area for further research. Although the Relief family of algorithms can deal with feature interaction, they cannot explicitly identify the set of interacting features. It is necessary to develop a method that can identify sets of interacting features that should be investigated as a group.

9 Supplementary information

9.1 Method for finding the most influential feature selection papers

A SCOPUS (`http://www.scopus.com`) citation search was conducted on March 16, 2014 using the term “feature selection” in article title, which turns up a total of 7,302 papers. Among these papers, 108 have at least 50 citations. The average number of citations per year was calculated for each of these papers as [total citations]/2014 – [year published]. There are 36 papers with at least 20 yearly citations. The yearly citations of these papers are sorted and plotted in Figure 1. The top 4 papers have over 100 yearly citations. The next 8 papers have more than 50 yearly citations. The 13th most frequently cited paper has 47.6 yearly citations; whereas the 14th has only 33.7 yearly citations. Therefore, we decided to focus on the top 13 papers. One of these papers is actually about feature detection, not feature selection; thus, this paper was excluded in our review. The citation information of the remaining 12 papers is summarized in Table 2.

Table 2: The top 12 most frequently cited feature selection papers

Author	Year	Total Citations	Yearly Citations
I. Guyon, A. Elisseeff	2003	3,157	287.0
R. Kohavi, G. H. John	1997	2,652	156.0
H. Peng, F. Long, C. Ding	2005	1,399	155.4
Y. Saeys, I. Inza, P. Larranaga	2007	769	109.9
H. Liu, L. Yu	2005	738	82.0
R. T. Collins, Y. Liu, M. Leordeanu	2005	538	59.8
A. L. Blum, P. Langley	1997	1,016	59.8
G. Forman	2003	649	59.0
P. Pudil, J. Novovicova, J. Kittler	1994	1,145	57.3
A. Jain, D. Zongker	1997	898	52.8
M. Dash, Liu H	1997	878	51.6
C.-L. Huang, C.-J. Wang	2006	381	47.6

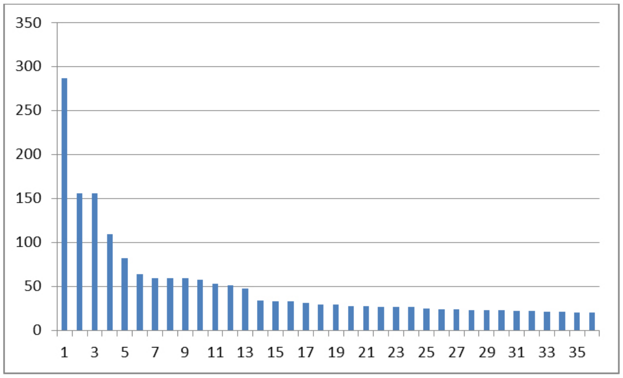

Figure 2: Sorted yearly citations of frequently cited feature selection papers

9.2 Performance analysis of feature selection methods

A number of researchers have evaluated the performance of different feature selection methods. The earliest study is probably that of Mucciardi & Gose[34] where the authors studied 7 techniques for feature selection on an electrocardiograms (EKG) dataset with 157 features and 9 classes. These techniques belong to feature ranking and feature subset configuration methods, summarized as follows:

(1) Probability of error: features are ranked based on the expected probability of error (POE), which is the fraction of patterns that are misclassified using a single feature. This technique belongs to pair-wise feature ranking using discriminative power.

(2) Average correlation coefficient: This is a feature subset configuration method. The first feature chosen is the one with the smallest POE. The second feature chosen is the one that has the smallest correlation coefficient with the first feature. The third feature is chosen such that its average correlation coefficient (ACC) with the first two features is the smallest. Subsequent features are chosen based on the smallest ACC defined above.

(3) Sequential: This is also a feature subset configuration method. Instead of using ACC, the feature to be added to a feature subset is the one that best discriminate the two most confused classes by the current feature subset.

(4) Eigenvector analysis: strictly speaking this is a not a feature selection but a feature transformation technique. Principal component analysis is used to create new features by linearly combining the original features. The new features are ranked based on their eigenvalues.

(5) Incomplete Eigenvectors: original features that make small contributions to the eigenvectors in the above mentioned principal component analysis are dropped from each eigenvectors when computing new fea-

tures. This is a feature transformation technique applied to a selected subset of features.

(6) Property weighting by eigenvector component: the average absolute weight of the original feature over the first 35 eigenvectors is used for feature ranking. This technique can be viewed as a simultaneous feature ranking method.

(7) Weighted sum: features are ranked according to a weighted sum of their POE when used alone and their ACC with the current feature subset. It is slightly more sophisticated than the second technique, but is still a feature subset configuration method.

Nested feature subsets (based on an ordered list of features) were used in a clustering decision rule for classification. The error rates produced were compared with that obtained when the features were ordered randomly. It was found that each of the 7 techniques resulted in lower error rates compared to randomly ordered features. In addition, ACC was found to be a much stronger criterion than POE alone, indicating the usefulness of feature subset configuration.

Kononenko[74] studied various criteria for ranking discrete (multi-valued) features. The author focused on the bias of these criteria caused by the number of values of a feature. The majority of the criteria are uncertainty related including information gain,[29] information gain ratio,[29] distance measure,[75] J-measure,[76] average absolute weight of evidence,[77] Gini-index,[55] and relevance measure.[78] Relief was also used as an individual feature evaluation criterion. It was reformulated as a function of a coefficient highly correlated with the Gini-index called Gini'. Gini' was also used as a ranking criterion. In addition, Kononenko introduced an evaluation measure based on the minimal description length principle[79] called MDL'. Other than these uncertainty related criteria, two hypothesis test criteria were studied; namely, chi-square statistic and G statistic. The findings are summarized as follows:

- Information gain, J-measure, Gini-index, and Gini' have a linear bias in favor of features with higher number of values. The relevance measure has a similar behavior except that its value increases less than linearly with the number of values.
- Information gain ratio, distance measure, and Relief have exponential bias against features with higher number of values. Interestingly, for features that are irrelevant, these three criteria exhibit bias in favor of features with higher number of values. The bias behavior of Relief is logarithmic; whereas that of information gain ratio and distance measure is linear.
- MDL' is biased against features with higher number of values. For irrelevant features, the value of MDL' is always negative.
- The behavior of average absolute weight of evidence is unstable. It seems to be somewhere between the relevance measure (for irrelevant features) and MDL' (for informative features).
- Chi-square statistic was found to be unbiased. However, Kononenko noted that Chi-square statistic cannot distinguish less informative features from more informative features because their p-values are indistinguishable (all take the value of 1) due to computation precision.
- G statistic was found to favor irrelevant features with higher number of values. However, Kononenko stated that this behavior contradicted that observed in an earlier study by White and Liu.[80] The author noted that this contradiction is likely due to the limited scenarios studied. Note that G statistic has the same problem as Chi-square statistic with respect to informative features.

Dash & Liu[9] focused on classification problems and evaluated the performance of a number of feature selection methods using three artificial datasets: (1) CorrAL with 32 instances, binary classes, and 6 Boolean features, (2) Modified parity with 64 instances, binary classes, and 12 Boolean features, and (3) Monk3 with 122 instances, binary classes, and 6 discrete features. These datasets are relatively simple (low dimensionality, discrete features). Nonetheless, it was clear that no single feature selection method is universally superior to the others. The authors recommended that a feature selection method should be chosen based on data set characteristics including data type, data size, and noise. They listed five criteria extracted from data set characteristics: (1) ability to handle different data types, (2) ability to handle multiple (more than two) classes, (3) ability to handle large dataset, (4) ability to handle noise, and (5) ability to produce optimal subset if data is not noisy. A table was presented describing the capability of 16 feature selection methods based on these five criteria.

Freeman et al.[72] studied the performance of 16 commonly used filter measures (including feature ranking, subset configuration, and model-independent subset evaluation methods) using 20 artificial and 20 real-world datasets. For the artificial datasets, the number of features is between 2 to 6. For the real-world datasets, the number of features is as many as 34. The authors concluded that appropriate feature selection measures are data set specific. They discussed the applicability of these measures in terms of the following data set characteristics: (1) whether the functional relationship between the target concept and features is monotonic or non-monotonic, (2) whether there is dependency among the features (feature interaction), (3) whether features are correlated, (4) whether there are redundant features, and (5) whether there are noisy features.

A more comprehensive comparative study was conducted by Hall & Holmes,[81] where the authors used 18 data sets of varying feature size and data instances to evaluate several representative feature selection methods. The methods are:

- information gain, a pair-wise feature ranking method
- ReliefF, a simultaneous feature ranking method
- correlation-based filter, a feature subset configuration method
- inconsistency rate, a model-independent feature subset evaluation method
- wrapper, a model-specific feature subset evaluation method.

In addition, principal component analysis was used, which is essentially a feature transformation method. Naïve Bayes and C4.5 were then used to build classification models. Note that C4.5 can be viewed as an embedded feature selection method. Thus, the study covers the majority of feature selection methods. Again, the conclusion that no single method is universally superior is confirmed. To choose an appropriate feature selection method, the authors suggested that one should (1) understand how each feature selection method works, (2) understand the strength and weakness of the modeling technique to be used, and (3) obtain as much background knowledge of the data set as possible. Other than this general guideline, the authors offered the following specific recommendations:

- If model accuracy is the most important consideration and computation speed is not an issue, then wrappers should be used.
- Otherwise, correlation-based filter, inconsistency rate, and ReliefF are good overall performers.
- Correlation-based filter is faster and chooses fewer features; whereas ReliefF is superb in finding interacting features.

Kohavi & John[8] studied the performance of wrappers (model-specific feature subset selection method) using 14 data sets with varying characteristics. ReliefF was used as a benchmark. It was found that after wrapper feature selection, models built using ID3, C4.5, and naïve-Bayes had significantly improved accuracy on some of the data sets. On real-world datasets, the wrapper approach was found to be superior to ReliefF. This is likely due to the fact that the Relief family of algorithms cannot detect redundant features, as pointed out in Ref.[9] However, ReliefF was found to outperform wrappers on the m-of-n-3-7-10 artificial dataset, which confirmed the unique ability of detecting feature interaction by the Relief family of algorithms.

Note that subset evaluation methods, both model-specific (wrappers) and model-independent, require a search strategy. The study by Kohavi & John[8] indicated that best-first search is preferable to hill-climbing search. Pudil et al.'s[11] showed that sequential floating search is a computationally efficient strategy that produced very good results. Siedlecki and Sklansky[49] advocated the use of genetic algorithms. Jain & Zongker[10] classified search strategies into four categories:

- deterministic single-solution methods, e.g., sequential search
- deterministic multiple-solution methods, e.g., beam search
- stochastic multiple-solution methods, e.g., genetic algorithms
- optimal methods, e.g., branch and bound

The authors applied a total of 14 search strategies, plus neural network node pruning (which is an embedded feature selection method) to a 20-dimensional 2-class dataset. Based on the results, they recommended sequential forward floating search (SFFS) as the best choice due to its excellent performance (comparable to that of optimal search methods) and efficiency (much faster than optimal search methods).

A number of researchers studied different feature selection methods for text classification. Text classification is a very high-dimensional problem and the use of wrappers is time consuming. Therefore, only feature ranking methods were evaluated. Yang & Pedersen[82] studied the performance of 5 feature ranking methods on a single data set. The study by Forman[83] is more comprehensive, involving 12 feature ranking methods on 19 multi-class datasets representing 229 binary text classification problem instances. In Ref.,[82] information gain and chi-square statistic were found to be most effective. This conclusion was partially confirmed by Foreman as the author noted that the outperformance occurs when the number of features used is restricted to less than 100. In addition, information gain outperformed chi-square statistic at every feature size. Foreman also found that a new ranking criterion, bi-normal separation, outperformed others by a substantial margin in most situations (including information gain at high number of features). The outperformance was more apparent in problems with high class skew (imbalanced data sets where the number of one class is significantly larger than that of the other class).

Another very high-dimensional problem is classification based on gene-expression. Li et al.[84] studied the combination of different feature ranking methods with different classification methods using 9 multi-class gene expression datasets. The top 150 genes were selected using 8 pair-wise feature ranking methods implemented in the Rankgene software package.[85] These methods are information gain, towing rule, sum minority, max minority, Gini index, sum of variances, one-dimensional SVM, and t-statistics. Models were then built using J4.8 decision tree, Naïve Bayes, K-nearest neighbor, and 4 variants of SVM (multi-dimensional). It was found that there are complex interactions between feature selection and classification modeling methods. The performance of decision tree was degraded after feature selection. Note that decision tree itself has a build-in mechanism for feature selection (i.e., it is an embedded feature selection method). Using only top ranked features to build a decision tree model can be viewed as taking away potentially useful features. This is likely the cause of performance deterioration. On the other hand, feature selection improved the performance of k-nearest neighbor and Naïve Bayes. Li et al. attributed this outperformance to the reduction in noise and dimensionality. The performance of SVMs was not uniform. In some datasets feature selection improves their performance but in other datasets it degrades their performance. Note that in the study by Kohavi & John,[8] feature selection did not always improve the performance of Naïve Bayes. Therefore, it is unclear under which situation feature selection would improve the performance of a specific classification model.

Note that most of the comparative studies use cross-validation to estimate model accuracy. For feature ranking and feature subset configuration methods, this estimate is appropriate because the features are selected independent of any specific models. However, for feature subset evaluation methods, this estimate might yield biased results due to overfitting. Reunanen[73] argued that for these computationally intensive feature selection methods, one should use an independent data set to evaluate the performance of the selected features. To prove this argument, the author compared two search methods, SFS and SFFS, using 1-nearest neighbor with leave-one-out cross-validation accuracy (LOOCV) on 7 data sets. Each data set was separated into a training set and a testing set. Only training sets were used in feature selection. When using LOOCV on the training sets, feature subsets found by SFFS clearly outperformed those found by SFS. Note that this is consistent with the conclusion by Jain & Zongker[10] that SFFS is the best search strategy. However, when the testing sets were used to evaluate the selected features, the outperformance disappeared. This study indicated that it is necessary to use independent test data sets for performance evaluation when comparing different feature selection methods. However, this performance evaluation method has yet to become a common practice in feature selection research. Therefore, one should be cautious when interpreting results from the comparative studies discussed here, especially with respect to feature subset evaluation methods.

References

[1] Miller AJ. Selection of Subsets of Regression Variables. Journal of the Royal Statistical Society, Series A-G. 1984; 147(3): 389-425. http://dx.doi.org/10.2307/2981576

[2] Hotelling H. The Selection of Variates for Use in Prediction with Some Comments on the General Problem of Nuisance Parameters. Annual of Mathematical Statistics. 1940: 11(3): 271-283. http://dx.doi.org/10.1214/aoms/1177731867

[3] Hocking RR. The Analysis and Selection of Variables in Linear Regression. Biometrics. 1976; 32(1): 1-49. http://dx.doi.org/10.2307/2529336

[4] Saeys Y, Inaki I, Larranaga P. A Review of Feature Selection Techniques in Bioinformatics. Bioinformatics. 2007; 23(19): 2507-2517. PMid:17720704. http://dx.doi.org/10.1093/bioinformatics/btm344

[5] Guyon I, Elisseeff A. An Introduction to Variable and Feature Selection. Journal of Machine Learning Research. 2003; 3(Mar): 1157–1182.

[6] Molina LC, Belanche L, Nebot A. Feature Selection Algorithms: A Survey and Experimental Evaluation. Proceedings of the 2002 IEEE International Conference on Data Mining. IEEE. New York, 2002: 306-313.

[7] Blum AL, Langley P. Selection of Relevant Features and Examples in Machine Learning. Artificial Intelligence. 1997; 97(1/2): 245-271.

[8] Kohavi R, John GH. Wrappers for Feature Subset Selection. Artificial Intelligence. 1997(1/2); 97: 273-324.

[9] Dash M, Liu H. Feature Selection for Classification. Intelligent Data Analysis. 1997; 1(3): 131-156. http://dx.doi.org/10.1016/S1088-467X(97)00008-5

[10] Jain A, Zongker D. Feature Selection: Evaluation, Application, and Small Sample Performance. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997; 19(2): 15-158.

[11] Pudil P, Novovicova J, Kittler J. Floating Search Methods in Feature Selection. Pattern Recognition Letters. 1994; 15(11): 1119-1125. http://dx.doi.org/10.1016/0167-8655(94)90127-9

[12] Liu H, Yu L. Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE Transactions on Knowledge and Data Engineering. 2005; 17(4): 491-502. http://dx.doi.org/10.1109/TKDE.2005.66

[13] Peng H, Long F, Ding C. Feature Selection based on Mutual Information: Criteria of Max-dependency, Max-relevance, and Min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2005; 27(8): 1226-1238. PMid:16119262. http://dx.doi.org/10.1109/TPAMI.2005.159

[14] Collins RT, Liu Y, Leordeanu M. Online Selection of Discriminative Tracking Features. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2005; 27(10): 1631-1643. PMid:16237997. http://dx.doi.org/10.1109/TPAMI.2005.205

[15] Keynes R. A Treatise on Probability. Macmillan, London, 1921.

[16] Gardenfors P. On the Logic of Relevance. Synthese. 1978; 37(3): 351-367. http://dx.doi.org/10.1007/BF00873245

[17] Pearl J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco, 1988.

[18] Gennari JH, Langley P, Fisher D. Models of Incremental Concept Formation. Artificial Intelligence. 1989; 40(1-3): 11-61.

[19] Blum AL. Relevant Examples and Relevant Features: Thoughts from Computational Learning Theory. AAAI Fall Symposium on Relevance. AAAI Press, Palo Alto. 1994: 14-18.

[20] John GH, Kohavi R, Pfleger K. Irrelevant Features and the Subset Selection Problem. Proceedings of the 11th International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 1994: 121-129.

[21] Cover TM. The Best Two Independent Measurements are not the Two Best. IEEE Transactions on Systems, Man, and Cybernetics. 1974; 4(1): 116-117. http://dx.doi.org/10.1109/TSMC.1974.5408535

[22] Yu L, Liu H. Efficient Feature Selection via Analysis of Relevance and Redundancy. Journal of Machine Learning Research. 2004; 5(Oct): 1205-1224.

[23] Koller D, Sahami M. Toward Optimal Feature Selection. Proceedings of the 13th International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 1996: 284-292.

[24] Cherkassky V, Mulier FM. Learning from Data: Concepts, Theory, and Methods. 2nd Edition, John Wiley & Sons, Hoboken, 2007.

[25] Narendra PM, Fukunaga K. A branch and bound algorithm for feature subset selection. IEEE Transactions on Computers. 1977; c26(9): 917-922.

[26] Dash M, Liu H, Motoda H. Consistency based Feature Selection. Proceedings of the Forth Pacific Asia Conference on Knowledge Discovery and Data Mining. Springer-Verlag, London. 2000: 98-109.

[27] Almuallim H, Dietterich TG. Learning Boolean Concepts in the Presence of Many Irrelevant Features. Artificial Intelligence. 1994; 69(1/2): 279-305.

[28] Pearson K. Notes on the History of Correlation. Biometrika, 1920; 13(1): 25-45. http://dx.doi.org/10.1093/biomet/13.1.25

[29] Quinlan R. Induction of Decision Trees. Machine Learning. 1986; 1(1): 81-106. http://dx.doi.org/10.1007/BF00116251

[30] Kira K, Rendell LA. A Practical Approach to Feature Selection. Proceedings of the 9th International Workshop for Machine Learning. Morgan Kaufmann, San Francisco. 1992: 249-259.

[31] Kononenko I. Estimating Attributes: Analysis and Extensions of Relief. Lecture Notes in Computer Science. 1994; 784: 171–182. http://dx.doi.org/10.1007/3-540-57868-4_57

[32] Robnik-Sikonja M, Kononenko I. An Adaptation of Relief for Attribute Estimation in Regression. Proceedings of the Fourteenth International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 1997: 296-304.

[33] Robnik-Sikonja M, Kononenko I. Theoretical and Empirical Analysis of Relief. Machine Learning. 2003; 53(1/2): 23-69. http://dx.doi.org/10.1023/A:1025667309714

[34] Mucciardi AN, Gose EE. A Comparison of Seven Techniques for Choosing Subsets of Pattern. IEEE Transactions on Computers. 1971; c20(9): 1023-1031.

[35] Hall MA. Correlation-based Feature Selection for Discrete and Numeric Class Machine Learning. Proceedings of the Seventeenth International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 2000: 359-366.

[36] Backer E, De Schipper JA. On the Max-min Approach for Feature Ordering and Selection. Seminar on Pattern Recognition. 1977; 2.4.1.

[37] Pudil P, Novovicova J, Choakjarernwanit N, et al. The Max-min Approach to Feature Selection: Its Foundations and Practical Potential. Indian Journal of Pure and Applied Mathematics. 1994; 25(1/2): 71-84.

[38] Mo D, Huang SH. Feature Selection based on Inference Correlation. Intelligent Data Analysis. 2011; 15(3): 375-398.

[39] Huang SH, Mo D, Meller J, et al. Identifying a Small Set of Marker Genes using Minimum Expected Cost of Misclassification. Artificial Intelligence in Medicine. 2012; 55(1): 51-59. PMid:22387186. http://dx.doi.org/10.1016/j.artmed.2012.01.004

[40] Wang X, Yang J, Teng X, et al. Feature Selection based on Rough Sets and Particle Swarm Optimization. Pattern Recognition Letters. 2007; 28(4): 459-471. http://dx.doi.org/10.1016/j.patrec.2006.09.003

[41] Swiniarski RW, Skowron A. Rough Set Methods in Feature Selection and Recognition. Pattern Recognition Letters. 2003; 24(6): 833-849. http://dx.doi.org/10.1016/S0167-8655(02)00196-4

[42] Zhong N, Dong J, Ohsuga S. Using Rough Sets with Heuristics for Feature Selection. Journal of Intelligent Information Systems. 2001; 16(3): 199-214. http://dx.doi.org/10.1023/A:1011219601502

[43] Hall MA, Smith LA. Feature Selection for Machine Learning: Comparing a Correlation-based Filter Approach to the Wrapper. Proceedings of the 12th International Florida Artificial Intelligence Research Society Conference. AAAI Press, Palo Alto. 1999: 235-239.

[44] Liu H, Motoda H. Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic Publishers, Norwell, 2000.

[45] Yu B, Yuan B. A More Efficient Branch and Bound Algorithm for Feature Selection. Pattern Recognition. 1993; 26(6): 883-889. http://dx.doi.org/10.1016/0031-3203(93)90054-Z

[46] Hamamoto Y, Uchimura S, Matsunra Y, et al. Evaluation of the Branch and Bound Algorithm for Feature Selection. Pattern Recognition Letters. 1990; 11(7): 453-456. http://dx.doi.org/10.1016/0167-8655(90)90078-G

[47] Stearns SD. On Selecting Features for Pattern Classifiers. Third International Conference on Pattern Recognition. IEEE, New York. 1976: 71-75.

[48] Huang CL, Wang CJ. A GA-based Feature Selection and Parameters Optimization for Support Vector Machines. Expert Systems with Application. 2006; 31(2): 231-240. http://dx.doi.org/10.1016/j.eswa.2005.09.024

[49] Siedlecki W, Sklansky J. A Note on Genetic Algorithms for Large-scale Feature Selection. Pattern Recognition Letters. 1989; 10(5): 335-347. http://dx.doi.org/10.1016/0167-8655(89)90037-8

[50] Skalak D. Prototype and Feature Selection by Sampling and Random Mutation Hill Climbing Algorithms. Proceedings of the Eleventh International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 1994: 293-301.

[51] Liu H, Setiono R. A Probabilistic Approach to Feature Selection: A Filter Solution. Proceedings of the 13th International Conference on Machine Learning. Morgan Kaufmann, San Francisco. 1996: 319-327.

[52] Inza I, Larranaga P, Etxeberria R, et al. Feature Subset Selection by Bayesian Network-based Optimization. Artificial Intelligence. 2000; 123(1/2): 157-184.

[53] Meiri R, Zahavi J. Using Simulated Annealing to Optimize the Feature Selection Problem in Marketing Applications. European Journal of Operational Research. 2006; 171(3): 842-858. http://dx.doi.org/10.1016/j.ejor.2004.09.010

[54] Sivagaminathan RK, Ramakrishnan S. A Hybrid Approach for Feature Selection using Neural Networks and Ant Colony Optimization. Expert Systems with Application. 2007; 33(1): 49-60. http://dx.doi.org/10.1016/j.eswa.2006.04.010

[55] Breiman L, Friedman JH, Olshen RA, et al. Classification and Regression Trees. Wadsworth and Brooks, Pacific Grove, 1984.

[56] Quinlan R. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco, 1993.

[57] LeCun Y, Denker J, Solla S, et al. Optimal brain damage. In Touretzky DS (ed), Advances in Neural Information Processing Systems II. Morgan Kaufmann, San Francisco. 1990: 598-605.

[58] Guyon I, Weston J, Barnhill S, et al. Gene Selection for Cancer Classification using Support Vector Machine. Machine Learning. 2002; 46(1-3): 389-422.

[59] Tibshirani R, Hastie T, Narasimhan B, et al. Diagnosis of Multiple Cancer Types by Shrunken Centroids of Gene Expression. Proceedings of the National Academy of Science. 2002; 99(10): 6567-6572. PMid:12011421. http://dx.doi.org/10.1073/pnas.082099299

[60] Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society Series B. 1996; 58(1): 267-288.

[61] Vapnik V. Statistical Learning Theory. John Wiley and Son, New York, 1998.

[62] Tayal A, Coleman TF, Li Y. Primal Explicit Max Margin Feature Selection for Nonlinear Support Vector Machines. Pattern Recognition. 2014; 47(6): 2153-2164. http://dx.doi.org/10.1016/j.patcog.2014.01.003

[63] Boubezoul A, Paris S. Application of Global Optimization Methods to Model and Feature Selection. Pattern Recognition. 2012; 45(10): 3676-3686. http://dx.doi.org/10.1016/j.patcog.2012.04.015

[64] Yang SH, Hu BG. Discriminative Feature Selection by Nonparametric Bayes Error Minimization. IEEE Transactions on Knowledge and Data Engineering. 2012; 24(8): 1422-1434. http://dx.doi.org/10.1109/TKDE.2011.92

[65] Song L, Smola A, Gretton A, et al. Feature Selection via Dependence Maximization. Journal of Machine Learning Research. 2012; 13(May): 1393-1434.

[66] Brown G, Pocock A, Zhao MJ, et al. Conditional Likelihood Maximization: A Unifying Framework for Information Theoretic Feature Selection. Journal of Machine Learning Research. 2012; 13(Jan): 27-66.

[67] Zhao Z, Wang L, Liu H, et al. On Similarity Preserving Feature Selection. IEEE Transactions on Knowledge and Data Engineering. 2013; 25(3): 619-632. http://dx.doi.org/10.1109/TKDE.2011.222

[68] Liang J, Wang F, Dang C, et al. A Group Incremental Approach to Feature Selection Applying Rough Set Technique. IEEE Transactions on Knowledge and Data Engineering. 2014; 26(2): 294-307. http://dx.doi.org/10.1109/TKDE.2012.146

[69] Shu W, Shen H. Incremental Feature Selection based on Rough Set in Dynamic Incomplete Data. Pattern Recognition. 2014; 47(12): 3890-3960. http://dx.doi.org/10.1016/j.patcog.2014.06.002

[70] Wang J, Zhao P, Hoi SCH, et al. Online Feature Selection and Its Applications. IEEE Transactions on Knowledge and Data Engineering. 2014; 26 (3): 698-710. http://dx.doi.org/10.1109/TKDE.2013.32

[71] Wessels LFA, Reinders MJT, Hart AAM, et al. A Protocol for Building and Evaluating Predictors of Disease State based on Microarray data. Bioinformatics. 2005; 21(19): 3755-3762. PMid:15817694. http://dx.doi.org/10.1093/bioinformatics/bti429

[72] Freeman C, Kulic D, Basir O. An Evaluation of Classifier-specific Filter Measure Performance for Feature Selection. Pattern Recognition. 2015; 48(5): 1812-1826. http://dx.doi.org/10.1016/j.patcog.2014.11.010

[73] Reunanen J. Overfitting in Making Comparisons between Variable Selection Methods. Journal of Machine Learning Research. 2003; 3(Mar): 1371-1382.

[74] Kononenko I. On Bias in Estimating Multi-valued Attributes. Proceedings of the 14th International Joint Conference on Artificial Intelligence. Morgan Kaufmann, San Francisco. 1995: 1034-1040.

[75] Mantaras RL. ID3 Revisited: A Distance based Criterion for Attribute Selection. Proceedings of the Fourth International Symposium on Methodologies for Intelligent Systems. North Holland, Amsterdam. 1989: 342-350.

[76] Smyth P, Goodman RM. Rule Induction Using Information Theory. In: G. Piatetsky-Shapiro and W. J. Frawley (ed), Knowledge Discovery in Databases. MIT Press, Cambridge, 1990.

[77] Michie D. Personal Models of Rationality. Journal of Statistical Planning and Inference. 1989; 25(3): 381-399. http://dx.doi.org/10.1016/0378-3758(90)90083-7

[78] Baim PW. A Method for Attribute Selection in Inductive Learning Systems. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1988; 10(6): 888-896. http://dx.doi.org/10.1109/34.9110

[79] Li M, Vitanyi P. An Introduction to Kolmogorov Complexity and Its Applications, Springer Verlag, Berlin, 1993.

[80] White AP, Liu WZ. Bias in Information-based Measures in Decision Tree Induction. Machine Learning. 1994; 15(3): 321-329. http://dx.doi.org/10.1007/BF00993349

[81] Hall MA, Holmes G. Benchmarking Attribute Selection Techniques for Discrete Class Data Mining. IEEE Transactions on Knowledge and Data Engineering. 2003; 15(3): 1-16.

[82] Yang Y, Pedersen JO. A Comparative Study on Feature Selection in Text Categorization. Proceedings of the 14th International Conference on Machine Learning. Morgan Kaufmann, San Francisco, 1997, pp. 412-420.

[83] Forman G. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research. 2003; 3(Mar): 1289-1305.

[84] Li T, Zhang C, Ogihara M. A Comparative Study of Feature Selection and Multiclass Classification Methods for Tissue Classification based on Gene Expression. Bioinformatics. 2004; 20(15): 2429-2437. PMid:15087314. http://dx.doi.org/10.1093/bioinformatics/bth267

[85] Su Y, Murali TM, Pavlovic V, et al. RankGene: Identification of Diagnostic Genes based on Expression Data. Bioinformatics. 2003; 19(12): 1578-1579. http://dx.doi.org/10.1093/bioinformatics/btg179