CONTENTS

CHAPTER 1

INFORMATION-THEORETIC GENE SELECTION IN EXPRESSION DATA

PATRICK E. MEYER AND GIANLUCA BONTEMPI
MACHINE LEARNING GROUP,
COMPUTER SCIENCE DEPARTMENT,
UNIVERSITÉ LIBRE DE BRUXELLES, BELGIUM.

{PMEYER,GBONTE}@ULB.AC.BE

1.1 INTRODUCTION

Genome-wide patterns of gene expression represent a snapshot of the state of a cell in a given condition. Using different snapshots, taken under different conditions, it becomes possible to build statistical models that can efficiently classify and predict new snapshots. A typical application of such techniques consists in discovering new molecular signatures of tumor cells for diagnostic and prognostic purposes [52, 53]. However, the detection of functional relationships between genes as well as the design of effective models from expression data is a major statistical challenge, mainly because of the data dimensionality. Expression datasets are typically characterized by a low number of noisy samples together with a high number of variables. As a result, even a simple predictive model such as a linear regression cannot be used without eliminating irrelevant and redundant variables as a first step [30]. A number of experimental studies [28, 6, 46] have shown that the elimination of irrelevant and redundant variables as well as the selection of synergetic variables [35, 24] can dramatically increase the predictive accuracy of models built from data. Moreover, variable selection can decrease future measurements and storage requirements [20] while increasing the intelligibility of a model.

In order to derive efficient methods of variable selection, formal definitions of *relevance, redundancy* and *synergy* of variables have been defined. *Information theory*, a theory introduced for data transmission and signal compression [48] and widely used in areas such as statistics, physics, economics or biology, provides a particularly adapted framework for measuring, quantifying and defining variable interactions [38].

The outline of this chapter is the following. Section 1.2 introduces the curse of dimensionality. Section 1.3 focuses on widely used variable exploration strategies. Section 1.4 intoduces the information-theoretic framework. Section 1.5 recalls variable selection techniques which have been proposed in the literature. Section 1.6 introduces estimation techniques that can be used for implementing the selection strategies on the basis of observed data.

1.2 THE CURSE OF DIMENSIONALITY

A natural question arises when it comes to make experiments or measurements: "how many experiments do I need in order to obtain a clear signal in my data?" or even "how many genes can I select when I have made 100 experiments"?

Although the question cannot be answered directly, it is easy to give an intuition behind "the curse of dimensionality". Most expression datasets provide continuous measurements of gene activities. However, in order to illustrate the problem, we will consider discretized variables, such as binary variables, i.e., genes that can be either "on" or "off". Binary variables have been studied extensively in statistics and the question that was raised at the time was "how many tosses of a coin should I make in order to assess if the coin is fair?" Although genes and coins are very different entities of a physical world, statisticians view them both as *random variables*. The more experiments you have the more confident you are in your estimate of the probability distribution of a random variable. Let us assume that you need an average of 5 samples per possible event ("on" or "off") in order to have a good estimate of the distribution of a random variable, that is to say you require ten samples in order to assess if a coin is fair. Now if you consider a joint distribution of two binary variables you are estimating a distribution that has 4 distinct possibilities (gene1:"on" gene2:"on", gene1:"on" gene2:"off", gene1:"off" gene2:"on", gene1:"off" gene2:"off"), you then require 20 samples. As you are estimating a joint distribution of 4 ternary variables ("on" "off or "in between"), you have $3 \times 3 \times 3 \times 3 = 81$ possibilities. Hence with the requirement of 5 samples by possibilities, you now need 405 samples to have an estimate of a joint distribution of only 4 ternary variables. In general, if you consider estimating a d-variate probability distribution of variables, each one discretized in p bins, and you need an average of 5 samples per bin, then you need $5 * p^d$ samples. You can however reduce the number of samples required by decreasing your estimation accuracy requirements (less than an average of 5 samples by possibilities) or by making more constraining assumptions on the joint distribution of these variables (such as assuming that some events are unlikely).

1.3 VARIABLE SELECTION EXPLORATION STRATEGIES

Let us consider the following problem: given d a number of variables to select, is there an algorithm that can select the optimal subset of variables of a given size?

Unfortunately, various results, like the theorem by Cover and Van Campenhout [14], show that finding the optimal subset of size d among n variables requires to test all $\left(\begin{array}{c} n \\ d \end{array} \right)$ combinations of subsets. However, this is impossible in practice, since it would take too much time to compute all of them. Hence, search heuristics have been used to reach a good predictive subset of variables.

Let us denote A the search space of $\left(\begin{array}{c} n \\ d \end{array} \right)$ subsets of random variables of X having size d. Variable selection can be seen as a combinatorial optimization problem [28] which depends on:

1. a method of exploring the space A (including the starting point and the stop criterion),

2. an evaluation function returning a measure of accuracy.

More formally the problem is: given n input variables X and a performance measure $F : A \rightarrow \mathbb{R}$, find the subset $X_S \subset X$ which maximizes the performance,

$$X_S^{max} = \arg \max_{X_S \in A} F(X_S) \tag{1.1}$$

Exploration strategies can be classified into three main categories of combinatorial optimization algorithms namely *optimal search*, *stochastic search* and *sequential search* (see [21], chapter 4).

1. Optimal search strategies include exhaustive search and branch-and-bound methods [21]. Their high computational complexity makes them impracticable with a high number of inputs and, for this reason, they are not discussed in this work.

2. Stochastic search strategies are also called *randomized* or *non-deterministic* [45] because two runs of these methods (with the same inputs) will not necessarily lead to the same result [16]. These methods explore a smaller portion of the search space A by using rules often inspired by nature. Some examples are: simulated annealing [16], tabu search [16] and genetic algorithms [55, 16].

3. Sequential search strategies are also called *deterministic heuristics* [45]. These methods are widely used for variable selection [17, 43, 18, 9]. Most of them use a neighbor search (two subsets are said neighbors if they differ from one variable) to discover a local optimum. Some examples are: forward selection (see Section 1.3.1), backward elimination (see Section 1.3.2), bi-directional search (see Section 1.3.3).

Because of their simplicity and their wide adoption in the variable selection community, we focus here on the three main sequential strategies namely the forward selection, the backward one and the bi-directional one.

In the following, we denote by X the complete initial set of variables and by $X_i^{METH} \in X,\ i \in A = \{1, 2, ..., n\}$ the variable selected at each step by the method $METH$. X_S and X_R are the set of selected variables and the set of remaining variables respectively. Hence, at each step $X = \{X_S, X_R\}$. X_i or X_j usually denotes a variable in X_R or in X_S, respectively.

1.3.1 Forward Selection search

Forward Selection [9, 28] is a sequential search method that starts with an empty set of variables, $X_S = \phi$. At each step, it selects the variable X_t that brings the best improvement (in terms of a given evaluation criterion $F(\cdot)$. A pseudo-code of the method is given in Algorithm 1.1. As a consequence of the sequential process, each selected variable influences the evaluations of the following steps.

This search has been widely used in variable selection, (see [9, 6, 28]). The forward selection algorithm selects a subset of $d < n$ variables in d steps and explores only $\sum_{i=0}^{d-1}(n-i)$ subsets.

However, this search has some weaknesses:

1. two variables that are synergetic (i.e., highly relevant only once taken together, see 1.4.3) appear as not relevant if taken individually and are as consequence ignored by this procedure,

2. selecting the best variable at each step does not mean selecting the best subset. Indeed, suppose that we have the following situation:

$$Y = f(X_5, X_4, X_3) + N(0, \sigma_1) = f(X_1, X_2) + N(0, \sigma_2) \tag{1.2}$$

where $N(\mu, \sigma)$ denotes a normally distributed noise with mean μ and variance σ. If we have, the following order of univariate relevance:

$$rel(X_5) > rel(X_1) > rel(X_2) > rel(X_4) \geq rel(X_3) \tag{1.3}$$

and

$$\sigma_1 > \sigma_2 \tag{1.4}$$

the best subset should be $X_{1,2} = \{X_1, X_2\}$ because the variance of the noise is smaller. Also, there are less variables in the latter combination, which usually lead to a lower number of parameters to estimate in a model. However, the forward selection algorithm will, in many cases, select the subset $X_{3,4,5} = \{X_5, X_4, X_3\}$. Indeed, it first selects X_5 because X_5 is the most relevant variable. Given X_5, the best improvement can be brought by X_4, and given $\{X_5, X_4\}$, X_3 can be the best variable to select.

Algorithm 1.1

```
   Inputs:    input variables X, the output variable Y, a maximal subset size
d > 0, a performance measure F(·) to maximize

   X_S := φ
   X_R := X
   while (|X_S| < d)
         maxscore := −∞
         for all the inputs X_i in the search space X_R
               Evaluate F(X_{S,i}) for the variable X_i with X_S the subset of
selected variables.
               if ( F(X_{S,i})> maxscore)
                     X_t:= X_i
                     maxscore:= F(X_{S,i})
               end-if
         end-for
          X_S := X_{S,t}
          X_R := X_{R−t}
   end-while
   Output:  the subset X_S
```

1.3.2 Backward Elimination search

Backward elimination [9, 28, 39] is a search method that starts by evaluating a subset containing all the variables $X_S = X$ and progressively discards the least relevant variables. For instance, at the second step, the method compares n subsets of $n-1$ inputs. The variable X_t associated with the least favorable improvement of accuracy is eliminated. The process is repeated until it yields the chosen number of inputs d (see Algorithm 1.2). This method does not suffer from the risk of ignoring a pair of complementary variables as it is the case for forward selection.

Algorithm 1.2

```
   Inputs:  input variables X (the input space), the output variable
   Y, a minimal subset size d > 0, and a performance measure F(·)
   to maximize

   X_S := X
   while (|X_S| > d)
         worstscore := ∞
         for all inputs X_j in the subset X_S
               Evaluate F(X_{S−j}), with all inputs of the subset X_S without
X_j
               if ( F(X_{S−j})<worstscore)
                     X_t:= X_i
                     worstscore:= F(X_{S−j})
               end-if
         end-for
         X_S := X_{S−t}
   end-while
   Output:  the subset X_S
```

1.3.3 Bi-directional search

The strengths of the forward selection and of the backward elimination can be combined in different manners.

As an example, let 26 random variables constitute the search space and be denoted by letters of the alphabet. Let the best subset of four variables be denoted by the letters $\{B, E, S, T\}$. The forward and the backward approaches can be combined in different ways:

- by using a backward elimination on a subset selected with a forward search [9].
  If a forward selection has selected the subset $\{C, B, E, S, T, D, F, G\}$, then, we can use a backward elimination in order to keep the most important variables of the subset, and reach the subset $\{B, E, S, T\}$.

- by performing a stepwise approach [39, 9]: At each step, choose the best action between eliminating a variable or selecting one.
  In our example, we may at some stage have selected the subset $\{E, A, S, T\}$. The stepwise algorithm chooses between adding a variable that brings the best improvement $\{B, E, A, S, T\}$ or eliminating the less important variable $\{E, S, T\}$.

- by using sequential replacement [39, 9]: This procedure consists in replacing $k \geq 1$ variables at each step.
  In our example, we can imagine at some stage having the subset $\{P, E, S, T\}$ that becomes the subset $\{B, E, S, T\}$ after an iteration. The pseudo-code of the algorithm for $k = 1$ is described in Algorithm 1.3.

Algorithm 1.3

```
Inputs:  a selected subset of inputs X_S, the set of remaining
variables X_R, the output variable Y, and a performance measure
F(·) to maximize

do
      for all inputs X_i in the remaining variables X_R
            Evaluate F(X_{S,i})
      end-for
      X_{t1} := arg max_{X_i} F(X_{S,i})
      for all for all inputs X_j in the subset X_S
            Evaluate the F(X_{S-j})
      end-for
      X_{t2} := arg max_{X_j} F(X_{S-j})
      X_S := X_{(S,t1)-t2}
      X_R := X_{(R,t2)-t1}
```

`end-do while` $X_{t1} \neq X_{t2}$
`Output: the subset` X_S

1.4 RELEVANCE, REDUNDANCY AND SYNERGY

In order to improve the resolution of the two variable selection problems stated above, i.e., subset estimation and search of good combination, many variable selection criteria have been developped in the past decade. These criteria focus on 1) select relevant variables without having to estimate accurately the full joint distribution of a subset 2) guide the heuristic search in the space of combinations.

These variable selection criteria have been built around three main notions: relevance, redundancy and synergy. These three notions can be efficiently formulated in an information-theoretic framework that is introduced in the following subsections. For simplicity, we consider here discrete variables though the theory can be extended to the continuous variable case.

1.4.1 Relevance

In this section entropy, conditional entropy and mutual information are defined. These notions will be intensively used in the following in order to define relevance, redundancy and synergy.

The *entropy* [10] of a discrete random variable Y with probability mass function $p(Y)$ is defined by:

$$H(Y) = H(p(Y)) = -\sum_{y \in \mathcal{Y}} p(y) \log p(y) = E_Y \left[\log \frac{1}{p(y)} \right] \tag{1.5}$$

Note that this definition remains valid for a discrete random vector (i.e. a subset of random variables).

The usual unit of the entropy is the *bit*. However, other units are sometimes chosen for this measure. The unit depends on the base taken for the logarithm of Eq 1.5, base 2 for bit, base 10 for *ban*. The *deciban* (one tenth of a ban) is also known as a useful measure of belief since 10 decibans correspond to an odds ratio of 10:1; 20 decibans to 100:1 odds, 30 decibans to 1000:1, etc [25]. The natural logarithm (base e) is increasingly used for computational reasons and in this case the unit is the *nat*.

The *conditional entropy* of Y given X is,

$$H(Y|X) = H(Y, X) - H(X) \tag{1.6}$$

This quantity measures the uncertainty of a variable once another one is known.

The reduction of entropy due to conditioning can be quantified by a symmetric measure called *mutual information* [10]:

$$H(Y) - H(Y|X) = I(Y; X) = I(X; Y) = H(X) - H(X|Y) \tag{1.7}$$

The *mutual information* between X and Y is,

$$I(Y;X) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \tag{1.8}$$

Mutual information can be also viewed as a divergence between the joint distribution $p(X,Y)$ and the product distribution $p(X)p(Y)$ [10] or between the marginal distribution $p(X)$ and a conditional distribution $p(X|Y)$. As a consequence, when two variables are independent, their mutual information is null and the higher the dependency between the variables, the higher the value of the mutual information. When the two variables are identical, this measure reaches its maximum and is equal to the entropy of the variable, i.e., $I(X;X) = H(X)$.

The mutual information is a natural measure of relevance since it quantifies the dependency level between random variables. The use of mutual information as relevance measure traces back to [11]. Later, [50] introduced a selection criterion called the *information bottleneck* which uses also mutual information as a relevance measure. [29] defines the *relevance* of a set X_S to an output variable Y as the mutual information $I(X_S;Y)$.

As a result, the relevance of an input variable X_i knowing a set X_S to an output variable Y is the gain of relevance resulting from using X_i additionally to X_S:

$$I(X_i;Y|X_S) = I(\{X_S, X_i\};Y) - I(X_S;Y) \tag{1.9}$$

This quantity is precisely the conditional mutual information [10]. Its normalized version (i.e. constrained to range between zero and one) has been introduced by [5] in a variable selection procedure.

Note that it is possible to increase the information of a variable with another by appropriate conditioning, as shown in the following example.

Let Y and X be two independent random variables and Z be a random variable defined as a deterministic function of Y and X (see Eq. 1.10).

$$X \rightarrow Z \leftarrow Y \tag{1.10}$$

As X and Y are independent, we have $I(X;Y) = 0$ and since $Z = f(X,Y)$, we obtain $I(X;Y|Z) > 0$. As a result, the conditional mutual information is higher than the mutual information, i.e. $I(X;Y|Z) > I(X;Y)$, which means that conditioning can increase relevance.

1.4.2 Redundancy

According to [54], a redundancy measure should be symmetric, non-negative and non-decreasing with the number of variables. The monotonicity is justified by the fact that, unlike relevance, the amount of redundancy of a variable can never decrease when more variables are added. As a result, [54] proposed to use multiinformation [34]. This measure used in [49] is also called *total correlation* in [23]. The multiinformation

between n sets of random variables $X_1, X_2, ..., X_n$ is,

$$R(X_i; ...; X_n) = \sum_{i=1}^{n} H(X_i) - H(X_1, X_2, ..., X_n) \tag{1.11}$$

Note that in the two-variables case,

$$R(X_i; X_j) = H(X_i) + H(X_j) - H(X_i, X_j) = I(X_i; X_j) \tag{1.12}$$

While the relevance measure concerns the relation between inputs and outputs, the bivariate redundancy measure applies exclusively to input variables.

1.4.3 Synergy

Synergy and redundancy are two sides of a coin that has been called *variable interaction*.The definition of variable interaction in information-theoretic terms can be found in the seminal paper of [34] and, more recently, in [23].

The interaction among n sets of random variables, $X_1, X_2, ..., X_n$ is defined as:

$$C(X_1; X_2; ...; X_n) = \sum_{k=1}^{n} \sum_{S \subseteq \{1,...,n\}:|S|=k} (-1)^{k+1} H(X_S) \tag{1.13}$$

Given a random vector X and a random variable Y the link between mutual information $I(X;Y)$ and interaction information is made explicit by the following formula from [29].

$$\begin{array}{c} I(X;Y) = \sum_{i \in A} C(X_i;Y) - \sum_{i,j \in A} C(X_i; X_j; Y) \\ +... + (-1)^{n+1} C(X_1; X_2; ...; X_n; Y) \end{array} \tag{1.14}$$

with $C(X_i;Y) = I(X_i;Y)$.

In plain words, mutual information can be seen as a series where higher order terms are corrective terms that represent the effect of the multivariate interaction. In most cases, the measure of interaction is positive and it indicates that the n sets of variables share a common information (redundancy). However, interaction can be negative. In the latter case, the variables are said to be complementary [35] or synergetic [2].

The synergy effect was mentioned in several variable selection papers [28, 20, 23, 24] and has been explicitely used in variable selection algorithms only recently [29, 35, 59].

In particular, the synergy between two random features X_i and X_j and the output Y ($n = 3$)

$$\begin{array}{c} -C(X_i; X_j; Y) = -H(X_i) - H(X_j) - H(Y) \\ +H(X_i, X_j) + H(X_i, Y) + H(X_j, Y) - H(X_i, X_j, Y) \\ = I(X_{i,j}; Y) - (I(X_i; Y) + I(X_j; Y)) \end{array} \tag{1.15}$$

measures, the gain resulting from using the joint mutual information of two variables X_i and X_j instead of the sum of the univariate informations. It is well-known,

indeed, that the joint information of two random variables, i.e., $I(X_{i,j}; Y)$ can be higher than the sum of their individual information $I(X_i; Y)$ and $I(X_j; Y)$.

An example of synergy is Example 1.4.1 where conditioning increase relevance. Another known illustration of this phenomenon is the XOR problem as pointed out by [28]:

X_1	X_2	$Y = X_1 \oplus X_2$
1	1	0
1	0	1
0	1	1
0	0	0

One can see that X_1 and X_2 have a null relevance individually, i.e. $I(X_1; Y) = 0$, $I(X_2; Y) = 0$, whereas together $X_{1,2}$ has a maximal relevance, i.e. $I(X_{1,2}; Y) = H(Y) > 0$. Synergy explains why a combination of apparently irrelevant variables can perform efficiently in a learning task. It also gives an intuition behind the Cover and Van Campenhout theorem [14] mentionned earlier, that requires to test all combinations of subset to find the optimal one.

1.5 INFORMATION-THEORETIC FILTERS

Variable selection methods based on mutual information are also called *information-theoretic filters*.

Let us start by stating the objective of an *information-theoretic filter*:

Given a training dataset D_m of m samples, an output variable Y, n input variables X and an integer $d \leq n$, find the subset $X_S \subseteq X$ of size d that maximizes the mutual information $I(X_S; Y)$.

In other words, the objective of *filters variable selection* (for a given d), is to find the subset X_S, with $|X_S| = d$, such that:

$$X_S^{max} = \arg \max_{X_S \subseteq X : |X_S| = d} I(X_S; Y) \tag{1.16}$$

This is a particular case of (1.1), where the evaluation function $F(X_S)$ is the mutual information $I(X_S; Y)$. We assume that the number of variables d has been determined by some a priori knowledge or by some cross-validation techniques. As filters often rank the variables according to their relevance measure, variables can be added one by one in a predictive model, until the cross-validated performances decrease. This procedure allows to reach an adequate number of variables for a given predictive model. Other strategies can be adopted such as the information bottleneck [50], Bayesian confidence on parametric estimations [27] or resampling techniques [19] .

In the following, we review the most important filter selection methods found in the literature which are based on information theory. We present the algorithms by stressing when and where the notion of relevance, redundancy and synergy are used.

1.5.1 Variable Ranking

This method *variable ranking* (RANK) returns a ranking of variables on the basis of their individual mutual informations with the output. This means that, given n input variables, the method first computes n times the quantity $I(X_i, Y)$, $i = 1, \ldots, n$, then ranks the variables according to this quantity and eventually discards the least relevant ones [17, 3].

The main advantage of this method is its low computational cost. Indeed, it requires only n computations of bivariate mutual information. The main drawback derives from the fact that possible redundancies between variables are not taken into account. Indeed, two redundant variables, yet highly relevant taken individually, will be both well-ranked. On the contrary, two variables could be synergetic to the output (i.e., highly relevant together) while being poorly relevant once each taken individually. As a consequence, these variables could be badly ranked, or even eliminated, by a ranking filter.

1.5.2 Fast Correlation Based Filter

Fast Correlation Based Filter (FCBF) is a ranking method combined with a redundancy analysis which has been proposed in [58]. The FCBF starts by selecting the variable (in the remaining variables X_R) with the highest mutual information, denoted by X_i^{FCBF}. Then, all the variables which are less relevant to Y than redundant to X_i^{FCBF} are eliminated from the list. For example, X_i is removed from the remaining variable set X_R if

$$I(X_i; X_i^{FCBF}) > I(X_i; Y)$$

At the next step, the algorithm repeats the selection and the elimination steps. The procedure stops when no more variable remains to be taken into consideration.

In other words, at each step, the set of selected variables X_S is updated with the variable

$$X_i^{FCBF} = \arg \max_{X_i \in X_R} I(X_i; Y) \tag{1.17}$$

and the set of remaining variables X_R is updated by removing the set

$$\{X_i \in X_{R-i} : \ I(X_i; Y) < I(X_i; X_i^{FCBF})\} \tag{1.18}$$

This method is affordable because a few (less than n^2) evaluations of bivariate mutual information are computed. However, although the method addresses redundancy, it presents the risk of eliminating relevant and synergetic variables. Another drawback of this method is that it does not return a complete ranking of the variables of the dataset. In [58], this approach is shown competitive with two filters [31, 1].

Note that in [58], a normalized measure of mutual information called the *symmetrical uncertainty* is used, i.e. $SU(X,Y) = \frac{2I(X;Y)}{H(X)+H(Y)}$. This measure helps to improve the performances of the selection by penalizing inputs with large entropies.

1.5.3 Backward elimination and Relevance Criterion

Let $X_S^{max} \subset X$ be the target subset, i.e., the subset X_S of size d, that achieves the maximal mutual information with the output (1.16). By the chain rule for mutual information [10],

$$I(X;Y) = I(X_S^{max};Y) + I(X_R^{max};Y|X_S^{max}) \tag{1.19}$$

where $X_R = X_{-S}$ is the set difference between the original set of inputs X and the set of variables X_S selected so far.

The backward elimination (using mutual information) [47] starts with $X_S = X$ and, at each step, eliminates from the set of selected variable X_S, the variable X_j^{back} having the lowest relevance on Y,

$$X_j^{back} = \arg\min_{X_j \in X_S} I(X_j;Y|X_{S-j}) \tag{1.20}$$

In other words, X_j^{back} is an approximation of X_R^{max} in (1.19). The approximation is exact for a subset size $d = n-1$ of one variable less than the complete set. The elimination process is then repeated until the desired size is reached. However, this approach is intractable for large variable sets since the beginning of the procedure requires the estimation of a multivariate density that includes the whole set of variables X.

1.5.4 Markov Blanket Elimination

The Markov blanket elimination [30] consists in approximating $I(X_j;Y|X_{S-j})$ in (1.20) by $I(X_j;Y|X_{M_j})$ with $X_{M_j} \subset X_{S-j}$ a subset of variables, i.e. the Markov blanket, having limited fixed size k. The algorithm proceeds in two phases. First, for every variable X_j in the selected set X_S, k variables X_{M_j} are selected among the variables X_{S-j}. Second, the least relevant variable X_j^{MB} (conditioned on the selected subset $X_{M_j} \subseteq X_{S-j}$) is eliminated, i.e., $X_S = X_S \setminus X_j^{MB}$.

$$X_j^{MB} = \arg\min_{X_j \in X_S} I(X_j;Y|X_{M_j}) \tag{1.21}$$

The process is repeated until the selected variable set X_S contains no more irrelevant and redundant variables, or when the desired subset size is reached. The method is named from the fact that X_{M_j} is an approximate Markov blanket. In [30], the Pearson's correlation coefficient [26] is used in order to find the k variables most correlated to the candidate X_j. These k variables are considered as the Markov blanket X_{M_j} of the candidate X_j. In this way, only linear dependencies between variables are considered. However, more complex functions can make the algorithm

very slow. In fact, finding a Markov blanket is itself a variable selection task. As a result, this method is adapted to large dimensionality problems only with very strong assumptions on the structure of X_M.

1.5.5 Forward Selection and Relevance Criterion

A way to sequentially maximize the *relevance* (REL) quantity $I(X_S; Y)$ in (1.16), is provided by the chain rule for mutual information [10]:

$$I(X_{S'}; Y) = I(X_S; Y) + I(X_i; Y|X_S) \tag{1.22}$$

where $X_{S'} = X_{S,i}$ is the updated set of variables. Rather than maximizing the left-hand side term directly, the idea of the forward selection combined with the relevance criterion consists in maximizing sequentially the second term of the right-hand term, $I(X_i; Y|X_S)$. In other words, the approach consists in updating a set of selected variables X_S with the variable X_i^{REL} featuring the maximum relevance.

In analytical terms, the variable X_i^{REL} returned by the relevance criterion at each step is,

$$X_i^{REL} = \arg \max_{X_i \in X_R} \{I(X_i; Y|X_S)\} \tag{1.23}$$

where $X_R = X_{-S}$ is the difference between the original set of inputs X and the set of variables X_S selected so far. This strategy prevents from selecting a variable which, though relevant to Y, is redundant with respect to a previously selected one. This algorithm has been used in [5, 3, 7, 47]. In [5], a normalized version of relevance is used.

Although this method is appealing, it presents some major drawbacks. The estimation of the relevance requires the estimation of large multivariate densities. For instance, at the d-th step of the forward search, the search algorithm requires $n - d$ evaluations, where each evaluation requires in turn the computation of a $(d + 1)$-variate density. It is known that for a large d, the estimations are poorly accurate and/or computationally expensive [44]. In particular in the small sample settings (around one hundred), having an accurate estimation of large ($d > 3$) multivariate densities is difficult (see 1.2). For these reasons, the recent filter literature adopt selection criteria based on bi- and trivariate densities at most.

1.5.6 Forward Selection and Conditional Mutual Information Maximization criterion

The *Conditional Mutual Information Maximization criterion* (CMIM) approach [18] proposes to select the variable $X_i \in X_R$ whose minimal relevance $I(X_i; Y|X_j)$ conditioned to each selected variable taken separately $X_j \in X_S$, is maximal. This requires the computation of the mutual information of X_i and the output Y, conditioned on each variable $X_j \in X_S$ previously selected.

Formally, the variable returned according to the CMIM is

$$X_i^{CMIM} = \arg \max_{X_i \in X_R} \{ \min_{X_j \in X_S} I(X_i; Y|X_j) \} \tag{1.24}$$

A variable X_i can be selected only if its information to the output Y has not been caught by an already selected variable X_j.

The CMIM criterion is an approximation of the relevance criterion,

$$X_i^{REL} = \arg \max_{X_i \in X_R} \{ I(X_i; Y|X_S) \}$$

where $I(X_i; Y|X_S)$ is replaced by $\min_{X_j \in X_S}(I(X_i; Y|X_j))$.

[18] shows experiments where CMIM is competitive with FCBF [58] in selecting binary variables for a pattern recognition task. This criterion selects relevant variables, avoids redundancy, avoids estimating high dimensional multivariate densities and does not ignore complementarity two-by-two. However, it does not necessarily select a variable complementary to the already selected variables. Indeed, a variable that has a high negative interaction to the already selected variable will be characterized by a large conditional mutual information with that variable but not necessarily by a large minimal conditional information. In the XOR problem, for instance, the synergetic variables have a null relevance taken alone. In that case, $\min_{X_j \in X_S} I(X_i; Y|X_j) = 0$ and CMIM would not select those variable.

1.5.7 Forward Selection and Minimum Redundancy - Maximum Relevance criterion

The *Minimum Redundancy-Maximum Relevance* (mRMR) criterion has been proposed in [44, 51, 43] in combination with a forward selection search strategy. Given a set X_S of selected variables, the method updates X_S with the variable $X_i \in X_R$ that maximizes $v_i - z_i$, where v_i is a relevance term and z_i is a redundancy term. More precisely, v_i is the relevance of X_i to the output Y alone, and z_i is the average redundancy of X_i to each selected variables $X_j \in X_S$.

$$v_i = I(X_i; Y) \tag{1.25}$$

$$z_i = \frac{1}{|X_S|} \sum_{X_j \in X_S} I(X_i; X_j) \tag{1.26}$$

$$X_i^{MRMR} = \arg \max_{X_i \in X_R} \{ v_i - z_i \} \tag{1.27}$$

At each step, this method selects the variable which has the best trade-off between relevance and redundancy. This selection criterion is fast and efficient. At step d of the forward search, the search algorithm computes $n - d$ evaluations where each evaluation requires the estimation of $(d + 1)$ bi-variate densities (one for each already selected variables plus one with the output). As a result, MRMR avoids the estimation of multivariate densities by using multiple bivariate densities.

A justification of MRMR given by the authors [44] is that

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{1.28}$$

with

$$R(X_1; X_2; ...; X_n) = \sum_{i=1}^{n} H(X_i) - H(X) \tag{1.29}$$

and

$$R(X_1; X_2; ...; X_n; Y) = \sum_{i=1}^{n} H(X_i) + H(Y) - H(X,Y) \tag{1.30}$$

hence

$$I(X;Y) = R(X_1; X_2; ...; X_n; Y) - R(X_1; X_2; ...; X_n) \tag{1.31}$$

where,

- the minimum of the second term $R(X_1; X_2; ...; X_n)$ is reached for independent variables since, in that case, $H(X) = \sum_i H(X_i)$ and $R(X_1; X_2; ...; X_n) = \sum_i H(X_i) - H(X) = 0$. Hence, if a subset of variables X_S is already selected, a variable X_i should have a minimal redundancy $I(X_i; X_S)$ with the subset. Pairwise independency does not guarantee independency. However, the authors approximate $I(X_i; X_S)$ with $\frac{1}{|S|} \sum_{j \in S} I(X_i; X_j)$.

- the maximum of the first term $R(X_1; X_2; ...; X_n; Y)$ is attained for maximally dependent variables.

Qualitatively, in a sequential setting where a selected subset X_S is given, independence between the variables in X is reached by minimizing $\frac{1}{|X_S|} \sum_{X_j \in X_S} I(X_i; X_j) \simeq I(X_i; X_S)$ and maximizing dependency between the variables of X and of Y, i.e., by maximizing $I(X_i; Y)$.

Although the method addresses the issue of bivariate redundancy through the term z_i, it does not capture synergy between variables. This can be ineffective in situations like Example 1.4.1 where, although the set $\{X, Z\}$ is very relevant for predicting Y, once X has been selected Z will not since

1. the redundancy term z_i is large due to the redundancy of X and Z,

2. the relevance term v_i is small since Z alone is not relevant to Y.

Nonetheless MRMR has been sucessfully used for network inference tasks when coupled with forward, backward and bi-directional searches [36, 37].

1.5.8 Forward Selection and Minimum Interaction - Maximum Relevance criterion

The *Minimum Interaction-Maximum Relevance* (mIMR) criterion, also called *Double Input Symmetrical Relevance* (DISR) in a previous version, has been proposed in [35, 8] in combination with a forward selection search strategy. Given a set X_S of selected variables, the method updates X_S with the variable $X_i \in X_R$ that maximizes $v_i - w_i$, where v_i is a relevance term and w_i is an interaction term. As in mRMR, v_i is the relevance of X_i to the output Y alone, however in mIMR the second term w_i is an average interaction of X_i to each selected variables $X_j \in X_S$.

$$v_i = I(X_i; Y) \tag{1.32}$$

$$w_i = \frac{1}{|X_S|} \sum_{X_j \in X_S} C(X_i; X_j; Y) \tag{1.33}$$

$$X_i^{mIMR} = \arg \max_{X_i \in X_R} \{v_i - w_i\} \tag{1.34}$$

At each step, this method selects the variable which has the best trade-off between relevance and interaction. At step d of the forward search, the search algorithm computes $n - d$ evaluations where each evaluation requires the estimation of a bivariate density plus d trivariate ones. Assuming normally distributed variables, trivariate densities can even be computed with bivariate terms, leading to much faster selection of variables. This criterion eliminates redundant variable (because they are penalized by a positive w_i) but also tends to select synergetic variable (because of their negative w_i) while avoiding the estimation of multivariate densities.

A theoretical justification of mIMR given by the authors is that

$$I(X_S; Y) \geq \frac{1}{\begin{pmatrix} d \\ 2 \end{pmatrix}} \sum_{X_i \in X_S} \sum_{X_j \in X_S} I(X_{i,j}; Y) \tag{1.35}$$

with

$$\arg \max_{X_i \in X_S} \sum_{X_j \in X_S} I(X_{i,j}; Y) = \arg \max_{X_i \in X_S} \left\{ I(X_i; Y) - \frac{1}{|X_S|} \sum_{X_j \in X_S} C(X_i; X_j; Y) \right\} \tag{1.36}$$

1.5.9 A theoretical comparison of filters

The presented criteria can be analyzed under different perspectives. We stress in Table 1.1,

1. which issues, among relevance, redundancy and synergy, are taken into account,

2. the ability of a criterion to avoid the estimation of large multivariate densities and

3. whether it returns a ranking of variables.

Table 1.2 reports a comparative analysis of the different techniques in terms of computational complexity of the evaluation step.

methods:	RANK	FCBF	REL	CMIM	mRMR	mIMR
Select Relevance	Yes	Yes	Yes	Yes	Yes	Yes
Eliminate Redundancy	No	Yes	Yes	Yes	Yes	Yes
2-variables synergy	No	No	Yes	No	No	Yes
Avoid Multivariate Density	Yes	Yes	No	Yes	Yes	Yes
Return Ranking	Yes	No	Yes	Yes	Yes	Yes

Table 1.1 Comparison of the properties (relevance, redundancy and synergy, ability to avoid estimation of large multivariate densities, ability to rank the variables) that are taken into account in each selection criterion.

variable evaluation:	RANK	REL	CMIM	mRMR	mIMR
calls of mutual information	1	1	d	$d+1$	$d+1$
k-variate density	2	$d+1$	3	2	3
computational cost	$O(C)$	$O(d \times C)$	$O(d \times C)$	$O(d \times C)$	$O(d \times C)$

Table 1.2 The computational cost of a variable evaluation using rel, cmim, mrmr, mimr with C being the cost of a mutual information estimation.

We observe from Tables 1.2 and 1.1 that the mIMR criterion avoids redundant variables, multivariate density estimation, but selects synergetic variables (up to the second order), at the same computational cost than CMIM and mRMR. Numerous experimental studies have shown the performances of mRMR and mIMR on expression datasets [38, 8].

1.6 FAST MUTUAL INFORMATION ESTIMATION

In the previous section, Table 1.2 shows that the computational complexity of most filters strongly depends on the cost of estimation of bi- and trivariate mutual information. We present here two very simple estimators that have shown good trade-off

between computational load and accuracy in a variable selection purposes [38, 41]. However, it should be noted that most filters exposed above could be used with other estimators or even with other evaluation functions/ similarity measures than mutual information.

Mutual information computation requires the determination of three entropy terms (see Section 1.4.1):

$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

An effective entropy estimation is then essential for computing mutual information. Entropy estimation has gained much interests over the last decade[13] and most approaches focus on reducing the bias inherent to entropy estimation.

For microarray datasets, bias reduction should not be the only criterion to choose an estimator. Bias reduction should be traded with speed/computational complexity of the estimator. Indeed, in variable selection and network inference, mutual information estimation routines are expected to be called a huge number of times and used for estimating tasks with the same number of variables and the same amount of samples. Since filters consists mainly in comparing information-theoretic quantities, the correct ranking of the estimated quantities is much more important than the correct estimation of the quantities. A similar conclusion has been reached in [33, 41]. We refer the reader to [42, 13, 4, 40, 12] for alternative approaches of entropy and mutual information estimation.

1.6.1 Discretizing variables and empirical estimation

We describe here the equal frequency discretization since it has been reported in [57] Meyer as one of the most efficient discretization method.

The equal frequency discretization scheme consists in partitioning the interval $[a, b]$ into $|\mathcal{X}_i|$ intervals, each having the same number, $m/|\mathcal{X}_i|$, of data points [15, 56, 32]. As a result, the intervals can have different sizes. If the $|\mathcal{X}_i|$ intervals have equal frequency, then the computation of entropy is straightforward: $\log \frac{1}{|\mathcal{X}_i|}$.

The value of the number of bins $|\mathcal{X}_i|$ controls a trade-off. With a too high $|\mathcal{X}_i|$, each bin will contain a few number of points, hence the variance is increased, whereas a too low $|\mathcal{X}_i|$ will introduce a too high loss of information [10]. A classical choice of $|\mathcal{X}_i|$ is given by the samples square root $\sqrt{m}$ [56]. One justification given for that choice is that the ratio $m/|\mathcal{X}_i|$ becomes $m/\sqrt{m} = \sqrt{m}$, hence there are as many bins as the average number of points per bin. Note also, that when estimating the entropy of a bivariate distribution where each variable has $\sqrt{m}$ bins, the number of bins of the joint distribution is upper-bounded by $|\mathcal{X}_i| \leq \sqrt{m} \times \sqrt{m} = m$. As a result, the empirical entropy estimator should not be too biased when combined with this choice of $|\mathcal{X}_i|$.

In order to compute the joint entropy of two discrete variables, the empirical estimator can be used. The empirical entropy estimator (also called "plug-in", "maximum likelihood" or "naive", [42]) is simply the entropy of the empirical distribution.

$$\hat{H}^{emp}(X) = -\sum_{x\in\mathcal{X}} \frac{\#(x)}{m} \log \frac{\#(x)}{m} \tag{1.37}$$

where $\#(x)$ is the number of data points having value x. Because of the convexity of the logarithmic function, underestimates of $p(x)$ cause errors on $E[\frac{1}{\log p(x)}]$ that are larger than errors due to overestimations. As a result, entropy estimators are biased downwards, that is,

$$E\big[\hat{H}^{emp}(X)\big] \leq H(X). \tag{1.38}$$

It has been shown in [42] that

1. the variance of the empirical estimator is upper-bounded by a term $\left(\frac{(\log m)^2}{m}\right)$ which depends only on the number of samples
2. the asymptotic bias is $-\frac{|\mathcal{X}|-1}{2m}$ and depends on the number of bins $|\mathcal{X}|$ [42]. As $|\mathcal{X}| \gg m$, this estimator can still have a low variance but the bias can become very large [42].

The computation of $\hat{H}^{emp}(X)$ has an $O(m)$ complexity cost.

The Miller-Madow correction is given by the following formula which is the empirical entropy corrected for the asymptotic bias,

$$\hat{H}^{mm}(X) = \hat{H}^{emp}(X) + \frac{|\mathcal{X}|-1}{2m} \tag{1.39}$$

where $|\mathcal{X}|$ is the number of bins with non-zero probability. This correction, while adding no computational cost, reduces the bias without changing variance. As a result, the Miller-Madow estimator is often preferred to the naive empirical entropy estimator.

1.6.2 Assuming Normally Distributed Variables

Another way to deal with continuous variables without discretizing variables and decreasing the computational cost, is to assume that variables follow a well known probability distribution, such as the Gaussian one.

Let X be a multivariate Gaussian, having a density function,

$$f(X) = \frac{1}{\sqrt{(2\pi)^n |V|}} exp^{(-\frac{1}{2}(x-\mu)^T V^{-1}(x-\mu))} \tag{1.40}$$

with mean μ and covariance matrix V.

The (differential) entropy of this distribution is [10]

$$H(X) = \frac{1}{2} \ln\{(2\pi e)^n |V|\} \tag{1.41}$$

where $|V|$ is the determinant of the covariance matrix [10].

As a result, the mutual information between two normal distributions is given by [22]

$$I(X_i, X_j) = \frac{1}{2} \log \left(\frac{\sigma_{ii}\sigma_{jj}}{|V|} \right) \tag{1.42}$$

where σ_{ii} and σ_{jj} are the standard deviations of X_i and X_j respectively. Hence

$$I(X_i, X_j) = -\frac{1}{2} \log(1 - \rho^2) \tag{1.43}$$

with ρ being the Pearson's correlation [26] between X_i and X_j. Note that the complexity of estimating $\hat{\rho}^2$ is $O(m)$, whith m is the number of samples.

1.7 CONCLUSIONS

Variable selection algorithms are mostly composed of two parts: a search strategy and an evaluation function. In the case of information-theoretic filters a third component is given by the mutual information estimator. In this chapter we have introduced eight different information-theoretic evaluation functions together with three heuristics searches and two mutual information estimators. The three sequential heuristics searches introduced, namely the forward, the backward and the bi-directional selection, share with the two mutual information estimators, the empirical and the gaussian, a low computational cost coupled with a growing literature of good empirical results. Having a low computational cost is critical in large datasets such as microarray data where the number of subset combinations is very high. An additional requirement brought by typical expression datasets is the ability to deal with a low number of samples. Most of the selection criteria presented here use combinations of only bi- and trivariate probability distributions in order to reduce the effect of the curse of dimensionality. Indeed, the latter require exponentially more samples for estimating larger joint distribution. Finally we a have introduced the notion of relevance, redundancy and synergy out of an information-theoretic framework in order to understand and compare each method's ability to combine those bi- and trivariate distributions in an efficient setting.

REFERENCES

1. H. Almuallim and T. G. Dietterich. Learning with many irrelevant features. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI91)*, pages 547–552. AAAI Press, 1991.
2. Dimitris Anastassiou. Computational analysis of the synergy among multiple interacting genes. *Molecular Systems Biology*, 2007.
3. R. Battiti. Using mutual information for selecting features in supervised neural net learning. In *IEEE Transactions on Neural Networks*, 1994.
4. J. Beirlant, E. J. Dudewica, L. Gyofi, and E. van der Meulen. Nonparametric entropy estimation: An overview. *Journal of Statistics*, 1997.

5. D. A. Bell and H. Wang. A formalism for relevance and its application in feature subset selection. *Machine Learning*, 41(2):175–195, 2000.

6. A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97:245–271, 1997.

7. B. V. Bonnlander and A. S. Weigend. Selecting input variables using mutual information and nonparametric density estimation. In *Proceedings of the 1994 International Symposium on Artificial Neural Networks (ISANN94)*, 1994.

8. G. Bontempi and P. E. Meyer. Causal filter selection in microarray data. In *International Conference On Machine Learning (ICML)*, 2010.

9. R. Caruana and D. Freitag. Greedy attribute selection. In *International Conference on Machine Learning*, pages 28–36, 1994.

10. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley, New York, 1990.

11. R. T. Cox. *Algebra of Probable Inference*. Oxford University Press, 1961.

12. G. Darbellay and I. Vajda. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, 1999.

13. C. O. Daub, R. Steuer, J. Selbig, and S. Kloska. Estimating mutual information using b-spline functions - an improved similarity measure for analysing gene expression data. *BMC Bioinformatics*, 5, 2004.

14. L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, 1996.

15. J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *International Conference on Machine Learning*, pages 194–202, 1995.

16. J. Dreo, A. Petrowski, P. Siarry, and E. Taillard. *Métaheuristiques pour l'Optimisation Difficile*. Eyrolles, 2003.

17. W. Duch, T. Winiarski, J. Biesiada, and A. Kachel. Feature selection and ranking filters. In *International Conference on Artificial Neural Networks (ICANN) and International Conference on Neural Information Processing (ICONIP)*, pages 251–254, June 2003.

18. F. Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, 2004.

19. D. François, F. Rossi, V. Wertz, and M. Verleysen. Resampling methods for parameter-free and robust feature selection with mutual information. *Neurocomputing*, 70(7-9):1276–1288, 2007.

20. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

21. Isabelle Guyon, Steve Gunn, Masoud Nikravesh, and Lotfi A. Zadeh. *Feature Extraction: Foundations and Applications*. Springer-Verlag New York, Inc., 2006.

22. S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall International, 1999.

23. A. Jakulin and I. Bratko. Quantifying and visualizing attribute interactions, 2003.

24. A. Jakulin and I. Bratko. Testing the significance of attribute interactions. In *Proc. of 21st International Conference on Machine Learning (ICML)*, pages 409–416, 2004.

25. E. T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

26. M. G. Kendall, A. Stuart, and J. K. Ord. *Kendall's advanced theory of statistics*. Oxford University Press, Inc., 1987.

27. M. B. Kennel, J. B. Shlens, H. D. I. Abarbanel, and E. J. Chichilnisky. Estimating entropy rates with bayesian confidence intervals. *Neural Computation*, 17(7), 2005.

28. R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

29. I. Kojadinovic. Relevance measures for subset variable selection in regression problems based on k-additive mutual information. *Computational Statistics and Data Analysis*, 49, 2005.

30. D. Koller and M. Sahami. Toward optimal feature selection. In *International Conference on Machine Learning*, pages 284–292, 1996.

31. I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. In *European Conference on Machine Learning*, pages 171–182, 1994.

32. H. Liu, F. Hussain, C. Lim Tan, and M. Dash. Discretization: An enabling technique. *Data Mining and Knowledge Discovery*, 6, 2002.

33. A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Dalla Favera, and A. Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7, 2006.

34. W. J. McGill. Multivariate information transmission. *Psychometrika*, 19, 1954.

35. P. E. Meyer and G. Bontempi. On the use of variable complementarity for feature selection in cancer classification. In F. Rothlauf et al., editor, *Applications of Evolutionary Computing: EvoWorkshops*, volume 3907 of *Lecture Notes in Computer Science*, pages 91–102. Springer, 2006.

36. P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi. Information-theoretic inference of large transcriptional regulatory networks. *EURASIP Journal on Bioinformatics and Systems Biology*, Special Issue on Information-Theoretic Methods for Bioinformatics, 2007.

37. P. E. Meyer, D. Marbach, S. Roy, and M. Kellis. Information-theoretic inference of gene networks using backward elimination. In *International Conference on Bioinformatics and Computational Biology (Biocomp)*, 2010.

38. P. E. Meyer, C. Schretter, and G. Bontempi. Information-theoretic feature selection using variable complementarity. *IEEE Journal of Special Topics in Signal Processing*, 2(3), 2008.

39. A.J. Miller. *Subset Selection in Regression Second Edition*. Chapman and Hall, 2002.

40. I. Nemenman, W. Bialek, and R. de Ruyter van Steveninck. Entropy and information in neural spike trains: Progress on the sampling problem. *Physical Review Letters*, 69, 2004.

41. C. Olsen, P. E. Meyer, and G. Bontempi. On the impact of entropy estimation on transcriptional regulatory network inference based on mutual information. *EURASIP Journal on Bioinformatics and Systems Biology*, 2009.

42. L. Paninski. Estimation of entropy and mutual information. *Neural Computation*, 15(6):1191–1253, 2003.

43. H. Peng and F. Long. An efficient max-dependency algorithm for gene selection. In *36th Symposium on the Interface: Computational Biology and Bioinformatics*, may 2004.
44. H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.
45. L. Portinale and L. Saitta. Feature selection. *Applied Intelligence*, 9(3):217–230, 1998.
46. G. Provan and M. Singh. Learning bayesian networks using feature selection. In *in Fifth International Workshop on Artificial Intelligence and Statistics*, pages 450–456, 1995.
47. F. Rossi, A. Lendasse, D. François, V. Wertz, and M. Verleysen. Mutual information for the selection of relevant variables in spectrometric nonlinear modelling. *Chemometrics and Intelligent Laboratory Systems*, 2:215–226, 2006.
48. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 1948.
49. M. Studený and J. Vejnarová. The multiinformation function as a tool for measuring stochastic dependence. In *Proceedings of the NATO Advanced Study Institute on Learning in graphical models*, pages 261–297, 1998.
50. N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. In *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing*, 1999.
51. G. D. Tourassi, E. D. Frederick, M. K. Markey, and Jr. C. E. Floyd. Application of the mutual information criterion for feature selection in computer-aided diagnosis. *Medical Physics*, 28(12):2394–2402, 2001.
52. M. J. van de Vijver, Y. D. He, L. J. van't Veer, H. Dai, A. A. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts, M. J. Marton, M. Parrish, D. Atsma, A. Witteveen, A. Glas, L. Delahaye, T. van der Velde T, H. Bartelink H, S. Rodenhuis, E. T. Rutgers, S. H. Friend, and R. Bernards. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medecine*, 347, 2002.
53. L. J. van 't Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 406, 2002.
54. W. Wienholt and B. Sendhoff. How to determine the redundancy of noisy chaotic time series. *International Journal of Bifurcation and Chaos*, 6(1):101–117, 1996.
55. J. Yang and V. Honavar. Feature subset selection using A genetic algorithm. In *Genetic Programming 1997: Proceedings of the Second Annual Conference*, page 380. Morgan Kaufmann, 1997.
56. Y. Yang and G. I. Webb. Discretization for naive-bayes learning: managing discretization bias and variance. Technical Report 2003/131 School of Computer Science and Software Engineering, Monash University, 2003.
57. Y. Yang and G. I. Webb. On why discretization works for naive-bayes classifiers. In *Proceedings of the 16th Australian Joint Conference on Artificial Intelligence*, 2003.
58. L. Yu and H. Liu. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5:1205–1224, 2004.
59. Z. Zhao and H. Liu. Searching for interacting features. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, 2007.