Sharper Bounds for the Hardness of Prototype and Feature Selection

Richard Nock[1] and Marc Sebban[2]

[1] Université des Antilles-Guyane, Dépt Scientifique Interfacultaire, Campus de Schoelcher
97233 Schoelcher, France
Richard.Nock@martinique.univ-ag.fr
[2] Université des Antilles-Guyane, Dépt de Sciences Juridiques, Campus de Fouillole
97159 Pointe-à-Pitre, France
Marc.Sebban@univ-ag.fr

Abstract. As pointed out by Blum [Blu94], "nearly all results in Machine Learning [...] deal with problems of separating relevant from irrelevant information in some way". This paper is concerned with structural complexity issues regarding the selection of relevant Prototypes or Features. We give the first results proving that both problems can be much harder than expected in the literature for various notions of relevance. In particular, the worst-case bounds achievable by any efficient algorithm are proven to be very large, most of the time not so far from trivial bounds. We think these results give a theoretical justification for the numerous heuristic approaches found in the literature to cope with these problems.

1 Introduction

With the development and the popularization of new data acquisition technologies such as the World Wide Web (WWW), computer scientists have to analyze potentially huge data sets. The available technology to analyze data has been developed over the last decades, and covers a broad spectrum of techniques and algorithms. The overwhelming quantities of such easy data represent however a noisy material for learning systems, and filtering it to reveal its most informative content has become an important issue in the fields of Machine Learning (ML) and Data Mining.

In this paper, we are interested in two important aspects of this issue: the problem of selecting the most relevant examples (named prototypes), a problem to which we refer as "Prototype selection" (PS), and the problem of selecting the most relevant variables, a problem to which we refer as "Feature selection" (FS). Numerous works have addressed empirical results about efficient algorithms for PS and FS [Koh94, KS95, KS96, SN00a, SN00b, Ska94, WM97] and many others. However, in comparison, very few results have addressed the theoretical issues of

H. Arimura, S. Jain and A. Sharma (Eds.): ALT 2000, LNAI 1968, pp. 224–238, 2000.

both PS and FS, and more particularly have given insight into the hardness of FS and PS. This is an important problem because almost all efficient algorithms presented so far for PS or FS are heuristics, and no theoretical results are given for the guarantees they give on the selection process. The question of their behavior in the worst case is therefore of particular importance. Structural complexity theory can be helpful to prove lowerbounds valid for any time-efficient algorithm, and negative results for approximating optimization problems are important in that they may indicate we can stop looking for better algorithms [Bel96]. On some problems [KKLP97], they have even ruled out the existence of efficient approximation algorithms in the worst case.

In this paper, we are interested in PS and FS as optimization problems. So far, one theoretical result exists [BL97], which links the hardness of approximating FS and the hardness of approximating the Min-Set-Cover problem. We are going to prove in that paper that PS and FS are very hard problems for various notions of what is ”relevance”, and our results go far beyond the negative results of [BL97]. The main difficulty in our approach is to capture the essential notions of relevance for PS and FS. As underlined in [BL97], there are many definitions for relevance, principally motivated by the question ”relevant to what?”, and addressing them separately would require large room space. However, these notions can be clustered according to different criteria, two of which seem to be of particular interest. Roughly speaking, relevance is generally to be understood with respect to a *distribution*, or with respect to a *concept*. While the former encompasses *information* measures, the latter can be concerned with the *target* concept (governing the labeling of the examples) or the *hypothesis* concept built by a further induction algorithm. In this work, we have chosen to address two notions of relevance, each representative of one cluster, for each of the PS and FS problems.

We prove for each of the four problems, that any time-efficient algorithm shall obtain very bad results in the worst case, much closer than expected to the ”performances” of approaches consisting in *not* (or randomly) filtering the data ! From a practical point of view, we think our results give a theoretical justification to heuristic approaches of FS and PS. While these hardness results have the advantage of covering the basic notions of relevance found throughout the literature (of course by investigating four *particular* definitions of relevance), they have two technical commonpoints. First, the results are obtained by reduction from the same problem (Min-Set-Cover), *but* they do *not* stem from a simple coding of the instance of Min-Set-Cover. Second, the proofs are standardized: they all use the same reduction tool but in a different way. From a technical point of view, the reduction technique makes use of *blow-up reductions*, a class of reductions between optimization problems previously sparsely used in Computational Learning Theory [HJLT94, NJ98a, NJS98]. Informally, blow-up reductions (also related to *self-improving reductions*, [Aro94]) are reductions which can be made from a problem onto itself: the transformation is such that it depends on an integer d which is used to tune the hardness result: the higher d, the larger the inapproximability ratio obtained. Of course, there is a price to pay : the reduction time is also an increasing function of d; however, sometimes, it is possible to show that the inapproximability ratio can be blown-up *e.g.* up to *exponent* d, whereas the reduction time increases reasonably as a function of d [NJS98].

The remaining of this paper is organized as follows. After a short preliminary, the two remaining parts of the paper address separately PS and FS. Since all our results use reductions from the same problem, we detail one proof to explain the nature of *self-improving reductions*, and give proof sketches for the remaining results.

2 Preliminary

Let LS be some learning sample. Each element of LS is an example consisting of an observation and a class. We suppose that the observations are described using a set V of n Boolean (0/1) variables, and there are only two classes, named "positive" (1) and "negative" (0) respectively. The basis for all our reductions is the minimization problem Min-Set-Cover:

Name: Min-Set-Cover.
Instance: a collection $C = \{c_1, c_2, ..., c_{|C|}\}$ of subsets of a finite set $S = \{s_1, s_2, ..., s_{|S|}\}$ ($|.|$ denotes the cardinality).
Solution: a set cover for S, *i.e.* a subset $C' \subseteq C$ such that every element of S belongs to at least one member of C.
Measure: cardinality of the set cover, *i.e.* $|C'|$.

The central theorem which we use in all our results is the following one.

Theorem 1. *[ACG+ 99, CK00] Unless $NP \subset DTIME[n^{\log\log n}]$, the problem* Min-Set-Cover *is not approximable to within $(1-\epsilon)\log|S|$ for any $\epsilon > 0$.*

By means of words, theorem 1 says that any (time) efficient algorithm shall not be able to break the logarithmic barrier $\log|S|$, that is, shall not beat significantly in the worst case the well-known *greedy set cover* approximation algorithm of [Joh74]. This algorithm guarantees to find a solution to any instance of Min-Set-Cover whose cost, $|C'|$, is not larger than

$$\mathcal{O}(\log|S|) \times \mathrm{opt}_{\textsc{Min-Set-Cover}},$$

where $\mathrm{opt}_{\textsc{Min-Set-Cover}}$ is the minimal cost for this instance.
In order to state our results, we shall need particular complexity classes based on particular time requirement functions. We say that a function is *polylog*(n) if it is $\mathcal{O}(\log^c n)$ for some constant c, and quasi-polynomial, $QP(n)$, if it is $\mathcal{O}(n^{polylog(n)})$.

3 The Hardness of Approximating Prototype Selection

A simple and formal objective to prototype selection can be thought of as an information preserving problem as underlined in [BL97]. Fix some function $f : [0,1] \rightarrow [0,1]$ satisfying the following properties:

1. f is symmetric about $1/2$,
2. $f(1/2) = 1$ and $f(0) = f(1) = 0$,
3. f is concave.

Such functions are called *permissible* in [KM96]. Clearly, the binary entropy

$$H(x) = -x \log(x) - (1-x) \log(1-x),$$

the Gini criterion

$$G(x) = 4x(1-x)$$

[KM96] and the criterion

$$A(x) = 2\sqrt{x(1-x)}$$

used in [KM96, SS98] are all permissible. Define $p_1(LS)$ as the fraction of positive examples in LS, and $p_0(LS)$ as the fraction of negative examples in LS. Define $LS_{v=a}$ to be for some variable v the subset of LS in which all examples have value a ($\in \{0,1\}$) for v. Finally, define the quantity $I_f(v, LS)$ defined as

$$I_f(v, LS) = f(p_1(LS)) - \left(\frac{|LS_{v=1}|}{|LS|} f(p_1(LS_{v=1})) + \frac{|LS_{v=0}|}{|LS|} f(p_1(LS_{v=0})) \right)$$

This quantity, with f replaced by the functions $H(x), G(x)$ or $A(x)$, represents the common information measure to split the internal nodes of decision trees in all state-of-the-art decision tree learning algorithms (see for example [BFOS84, KM96, Mit97, Qui94, SS98]).

One objective in prototype selection can be to reduce the number of examples in LS while ensuring that any informative variable before will remain informative after the removal. The corresponding optimization problem, which we call Min-PS$_f$ (for any f belonging to the category fixed above), is the following one:

Name: Min-PS$_f$
Instance: a learning sample LS of examples described over a set of n variables $V = \{v_1, v_2, ..., v_n\}$.
Solution: a subset LS' of LS such that $\forall 1 \leq i \leq n, I_f(v_i, LS) > 0 \Rightarrow I_f(v_i, LS') > 0$.
Measure: $|LS'|$.

There are two components in the self-improving reduction. The first one is to prove a basic inapproximability theorem. The second one, an *amplification lemma*, "blows-up" the result of the theorem. Then, we give some consequences illustrating the power of the amplification lemma.

Theorem 2. *Unless* $NP \subset DTIME[n^{\log\log n}]$, Min-PS$_f$ *is not approximable to within* $(1-\epsilon)\log n$ *for any* $\epsilon > 0$.

Proof. We show that Min-PS$_f$ is as hard to approximate as Min-Set-Cover: any solution to Min-Set-Cover can be polynomially translated to a solution to Min-PS$_f$ of the same cost, and reciprocally. Given an instance of Min-Set-Cover, we build a set LS of $|C|$ positive examples and 1 negative example, each described over $|S|$ variables. We define a set $\{v_1, v_2, ..., v_{|S|}\}$ of Boolean variables, in one-to-one correspondence with the elements of S. The negative example is the all-0 example. Each positive example is denoted $e_1, e_2, ..., e_{|C|}$. We construct each positive example e_j so that it encodes the content of the corresponding set c_j of C. Namely, $e_j[k]$ is 1 iff $s_k \in c_j$, and 0 otherwise. Here we suppose obviously that each element of S is element of at least one element of C, which means that $\forall 1 \leq i \leq n, I_f(v_i, LS) > 0$. Suppose there exists a solution to Min-Set-Cover of cost c. Then, we put in LS' the negative example, and all positive examples corresponding to the solution to Min-Set-Cover. We see that for any variable v_j, there exists some positive example of LS' having 1 in its j^{th} component, since otherwise the solution to Min-Set-Cover would not cover the elements of S. It is straightforward to check that $\forall 1 \leq i \leq n, I_f(v_i, LS') > 0$, which means that LS' is a solution to Min-PS$_f$ having cost $c+1$.

Now, suppose that there exists a feasible solution to Min-PS$_f$, of size c. There must be the negative example inside LS' since otherwise we would have $\forall 1 \leq i \leq n, I_f(v_i, LS') = 0$. Consider all elements of C corresponding to the $c-1$ positive examples of LS'. If some element s_i of S were not covered, the variable v_i would be assigned to zero over all examples of LS', be they positive or negative. In other words, we would have $I_f(v_i, LS') = 0$, which is impossible. In other words, we have build a solution of Min-Set-Cover of cost $c-1$.

If we denote opt$_{\text{Min-Set-Cover}}$ and opt$_{\text{Min-PS}}$ the optimal costs of the problems, we have immediately opt$_{\text{Min-PS}}$ = opt$_{\text{Min-Set-Cover}} + 1$. A possible interpretation of theorem 1 is the following one [Aro94]: there exists some $\mathcal{O}(n^{\log\log n})$-time reduction from some NP-hard problem, say "SAT" for example, to Min-Set-Cover, such that

- to any satisfiable instance of "SAT" corresponds a solution to Min-Set-Cover whose cost is α,
- unsatisfiable instance of "SAT" are such that any feasible solution to Min-Set-Cover will be of cost $> \alpha(1-\epsilon)\log|S|$ for any $\epsilon > 0$.

This property is also called a *hard gap* in [Bel96].
If we consider the reduction from Min-Set-Cover to Min-PS$_f$, we see that the ratio between unsatisfiable and satisfiable instances of "SAT" is now

$$\rho = \frac{\alpha(1-\epsilon)\log n + 1}{\alpha + 1}$$

For any $\epsilon' > 0$, if we choose $0 < \epsilon < \epsilon'$ (this is authorized by theorem 1), we have $\rho > (1-\epsilon')\log n$ for Min-PS$_f$, at least for sufficiently large instances of

"SAT". This concludes the proof of the theorem. □

The amplification lemma is based on the following self-improving reduction. Fix some integer value $d > 1$. Suppose we take again the instance of Min-Set-Cover, but we create $|S|^d$ variables instead of the initial $|S|$. Each variable represents now a d-tuple of examples. Suppose we number the variables $v_{i_1,i_2,...,i_d}$ with $i_1, i_2, ..., i_d \in \{1, 2, ..., |S|\}$, to represent the corresponding examples. The $|C| + 1$ old examples are replaced by $|C|^d + 1$ examples described over these variables, as follows:

- for any possible d-tuple $(c_{j_1}, c_{j_2}, ..., c_{j_d})$ of elements of C, we create a positive example $e_{j_1,j_2,...,j_d}$, having ones in variable $v_{i_1,i_2,...,i_d}$ iff

$$\forall k \in \{1, 2, ..., d\}, s_{i_k} \in c_{j_k},$$

  and zeroes everywhere else. Thus, the Hamming weight of the example's description is exactly $\prod_{k=1}^{d} |c_{j_k}|$. By this procedure, we create $|C|^d$ positive examples,
- we add the all-zero example, having negative class.

We call LS_d this new set of examples. Note that the time made for the reduction is no more than $\mathcal{O}(|S|^d|C|^d)$. The following lemma exhibits that the inapproximability ratio for Min-PS$_f$ actually grows as a particular function of d provided d is confined to reasonable values, in order to keep an overall reduction time not greater than $\mathcal{O}(n^{\log\log n})$. Informally, this assumption allows to use the inapproximability ratio of theorem 1 for our reduction. For the sake of simplicity in stating the lemma, we say that the reduction is *feasible* to state that this assumption holds.

Lemma 1. *Unless $NP \subset DTIME[n^{\log\log n}]$, provided the reduction is feasible, then* Min-PS$_f$ *is not approximable to within*

$$\left(\frac{(1-\epsilon)\log n}{d}\right)^d$$

for any $\epsilon > 0$.

Proof. Again, we suppose obviously that each element of S is element of at least one element of C, which means that each variable $v_{i_1,i_2,...,i_d}$ has

$$I_f(v_{i_1,i_2,...,i_d}, LS_d) > 0$$

Note that any feasible solution to Min-PS$_f$ contains the negative example (same reason as for theorem 2). Also, in any solution $C' = \{c'_1, c'_2, ..., c'_{|C'|}\}$ to Min-Set-Cover, the following property **P** is satisfied without loss of generality: any element of C belonging to it has at least one element (of S) which is present in no other element of C', since otherwise the solution could be transformed in polynomial time into a solution of lower cost (simply remove arbitrarily elements in C' to satisfy **P** while keeping a cover of S). As **P** is satisfied, we call any subset of cardinality $|C'|$ of S containing one such distinguished element for each element of C' a *distinguished* subset of S. Finally, remark that Min-PS$_f$ is equivalent to the problem of covering the set S^d using elements of C^d, and the minimal number of positive examples in LS_d is exactly the minimal cost c' of the instance of this generalization of Min-Set-Cover. But, since **P** holds, covering C^d requires to cover any d-tuple of distinguished subsets of S and because property **P** holds, c' is at least c^d where c is the optimal cost of the instance of Min-Set-Cover. Also, if we take all d-tuples of elements of C' feasible solution to Min-Set-Cover, we get a feasible solution to the generalization of Min-Set-Cover, which leads to the equality $c' = c^d$.

If we denote $\mathrm{opt}_{\textsc{Min-PS}}$ the optimal cost of Min-PS$_f$ on the new set of examples LS_d, we obtain that

$$\mathrm{opt}_{\textsc{Min-PS}} = (\mathrm{opt}_{\textsc{Min-Set-Cover}})^d + 1$$

Given that $n = |S|^d$, and using the same ideas as for theorem 2, we obtain the statement of the lemma. □

What can we hope to gain by using lemma 1, which was not already proven by theorem 2 ? It is easy to show that the largest inapproximability ratio authorized by the same complexity assumption is

$$\rho = \log^{\log\left(\frac{\log n^{1-\epsilon}}{\log\log n}\right)} n \qquad (1)$$

(by taking $d = \mathcal{O}(\log\log n)$), which implies the simpler one:

Theorem 3. *Unless $NP \subset DTIME[n^{\log\log n}]$,* Min-PS$_f$ *is not approximable to within*

$$\log^{(1-\epsilon)\log\log n} n$$

for any $\epsilon > 0$.

Another widely encountered complexity hypothesis, stronger than the one of theorem 3, is that $NP \not\subset QP$ [CK00]. In that case, the result of theorem 3 becomes stronger:

Theorem 4. *Unless $NP \subset QP$, $\exists\delta > 0$ such that* Min-PS$_f$ *is not approximable to within n^δ.*

Proof. We prove the result for $\delta < 1/e$, and take $d = (1-\delta)\log n$. A good choice of ϵ in theorem 2 proves the result. □

The preceeding model takes into account the information of the variables to select relevant prototypes. We now give a model for prototype selection based on the notion of relevance with respect to a concept. For any set of examples LS, denote as $\mathcal{C}_{opt}(LS)$ the set of concept representations having minimal size, and consistent with LS. The notion of size can be *e.g.* the overall number of variables of the concept (if a variable appears i times, it is counted i times). The nature of the concepts is not really important: these could be decision trees, decision lists, disjunctive normal form formulas, linear separators, as well as simple clauses. Our negative results will force the concepts of $\mathcal{C}_{opt}(LS)$ to belong to a particularly simple subclass, expressible in each class. This notion of relevance is closely related to a particular kind of ML algorithms in which we seek consistent formulas with limited size: Occam's razors [KV94, NJS98]. Formulated as an optimization problem, the Min-PS problem is the following one:

Name: Min-PS.
Instance: a learning sample LS of examples described over a set of variables $\{v_1, v_2, ..., v_n\}$.
Solution: a subset LS' of LS such that $\mathcal{C}_{opt}(LS') \subseteq \mathcal{C}_{opt}(LS)$.
Measure: $|LS'|$.

By means of words, PS is a problem of reducing the number of examples while ensuring that concepts consistent and minimal with respect to the subset of prototypes will also be valid for the whole set of examples. Our first result on the inapproximability of this new version of Min-PS is the following one.

Theorem 5. *Unless $NP \subset DTIME[n^{\log\log n}]$,* Min-PS *is not approximable to within $(1-\epsilon)\log n$ for any $\epsilon > 0$.*

Proof. (*sketch*) The proof resembles the one of theorem 2. Given an instance of Min-Set-Cover, we build a set LS of $|S|$ positive examples and 1 negative example, each described over $|C|$ variables. We define a set $\{v_1, v_2, ..., v_{|C|}\}$ of Boolean variables, in one-to-one correspondence with the elements of C. The negative example is the all-0 example. Each positive example is denoted $e_1, e_2, ..., e_{|S|}$. We construct each positive example e_j so that it encodes the membership of s_j into each element of C. Namely, $e_j[k]$ is 1 iff $s_j \in c_k$, and 0 otherwise. Similarly to theorem 2, the least number of examples which can be kept is exactly the cost of the optimal solution to Min-Set-Cover, plus one.

The proof is similar to that of theorem 2, with the following remark on the minimal concepts. It can be shown that minimal concepts belonging to each of the classes cited before (trees, lists, etc.) will contain a number of variables equal to the minimal solution to Min-Set-Cover, and each will be present only once. The reduction is indeed very generic and similar results were previously obtained by *e.g.* [NG95] (for linear separators and even multilinear polynomials), [NJ98b] (for decision lists), [HR76, HJLT94] (for decision trees), [Noc98] (for Disjunctive Normal Form formulas and simple clauses). From that, all minimal concepts will be equivalent to a simple clause whose variables correspond to C'. Property **P** in lemma 1 can still be used. □

The amplification lemma follows from a particular self-improving reduction. Again, fix some integer value $d > 1$. Suppose we take again the instance of Min-Set-Cover, but we create $d|C|$ variables instead of the initial $|C|$. Each variable is written $v_{i,j}$ to denote the j^{th} copy of initial variable i, with $i = 1, 2, ..., |C|$ and $j = 1, 2, ..., d$. The $|S| + 1$ old examples are replaced by $|S|^d + 1$ examples described over these variables, as follows:

- for any possible d-tuple $(s_{j_1}, s_{j_2}, ..., s_{j_d})$ of elements of S, we create a positive example $e_{j_1, j_2, ..., j_d}$, having ones in variable $v_{k,l}$ iff $s_{j_l} \in c_k$, and zeroes everywhere else. By this procedure, we create $|C|^d$ positive examples,
- we add the all-zero example, having negative class.

We call LS_d this new set of examples. Note that the time made for the reduction is no more than $\mathcal{O}(|S|^d |C|^d)$. The following lemma is again stated under the hypothesis that the reduction is *feasible*, that is, takes no more time than $\mathcal{O}(n^{\log \log n})$, to keep the same complexity assumption as in theorem 1 (proof omitted).

Lemma 2. *Unless $NP \subset DTIME[n^{\log \log n}]$, provided the reduction is feasible, then* Min-PS *is not approximable to within*

$$\left((1 - \epsilon) \log \left[\frac{n}{d}\right]\right)^d$$

for any $\epsilon > 0$.

What can we hope to gain by using lemma 2, which was not already proven by theorem 5 ? It is easy to show that the largest inapproximability ratio authorized by the same complexity assumption is now

$$\rho = \log^{\log \log \left(\frac{n}{\log \log n}\right)^{1-\epsilon}} n \tag{2}$$

which in turn implies the following one (greater than eq. 1):

Theorem 6. *Unless $NP \subset DTIME[n^{\log \log n}]$,* Min-PS *is not approximable to within*

$$\log^{\log \log \left(n^{1-\epsilon}\right)} n$$

for any $\epsilon > 0$.

With a slightly stronger hypothesis (and using $d = \mathcal{O}(polylog(n))$), we obtain

Theorem 7. *Unless $NP \subset QP$, $\forall c > 0$,* Min-PS *is not approximable to within $n^{\log^c n \log \log \log n}$.*

With respect to 1, lemma 2 brings results much more negative provided stronger complexity assumptions are made. [PR94] make the very strong complexity assumption $NP \not\subset DTIME(2^{n^{\Omega(1)}})$. This is the strongest complexity assumption, since NP is definitely contained in $DTIME(2^{poly(n)})$. Using this hypothesis with $d = n^{\Omega(1)}$, we obtain the following, very strong result:

Theorem 8. *Unless $NP \subset DTIME(2^{n^{\Omega(1)}})$, $\exists \gamma > 0$ such that* Min-PS *is not approximable to within*

$$2^{n^{\gamma} \log \log n}$$

What theorem 8 says is that approximating prototype selection up to exponential ratios

$$2^{n^{\gamma+o(1)}}$$

will be hard. Note that storing the examples would require 2^n examples in the worst case. Up to what is precisely hidden in the γ notation, approximating Min-PS might not be efficient at all with respect to the storing of all examples.

4 The Hardness of Approximating Feature Selection

The first model of feature selection is related to the distribution of the examples in LS. Let V_i be the set of all variables except v_i, *i.e.*

$$V_i = \{v_1, v_2, ..., v_{i-1}, v_{i+1}, ..., v_n\}$$

Denote by $v_{\setminus i}$ a value assignment to all variables in V_i.

Definition 1. *[JKP94] A variable v_i is* ***strongly*** *relevant iff there exists some v, y and $v_{\setminus i}$ for which $\mathbf{Pr}(v_i = v, V_i = v_{\setminus i}) > 0$ such that*

$$\mathbf{Pr}(Y = y|v_i = v, V_i = v_{\setminus i}) \neq \mathbf{Pr}(Y = y|V_i = v_{\setminus i})$$

Definition 2. *[JKP94] A variable v_i is* ***weakly*** *relevant iff it is not strongly relevant, and there exists a subset of features V'_i of V_i for which there exists some v, y and $v'_{\setminus i}$ with $\mathbf{Pr}(v_i = v, V'_i = v'_{\setminus i}) > 0$ such that*

$$\mathbf{Pr}(Y = y|v_i = v, V'_i = v'_{\setminus i}) \neq \mathbf{Pr}(Y = y|V'_i = v'_{\setminus i})$$

In other words, a feature is weakly relevant if it becomes strongly relevant after having deleted some subset of features. We now show that under these two definitions are hidden algorithmic problems of very different complexities. We formulate the selection of relevant features as an optimization problem by focusing on the class conditional probabilities, following the definition of *coherency* which we give below:

Definition 3. *Given a whole set V of features with which LS is described, a subset V' of V is said to be* ***coherent*** *iff for any class y and any observation s described with V whose restriction to V' is noted s', we have*

$$\mathbf{Pr}(Y = y|V = s) = \mathbf{Pr}(Y = y|V' = s')$$

By means of words, coherency aims at keeping the class conditional probabilities between the whole set of variables and the selected subset. Formulated as an optimization problem, the Min-S-FS problem is the following one:

- Name: Min-S-FS.
- Instance: a learning sample LS of examples described over a set of variables $V = \{v_1, v_2, ..., v_n\}$.

- Solution: a coherent subset V' of V containing strongly relevant features w.r.t. LS.
- Measure: $|V'|$.

The Min-W-FS problem is the following one:

- Name: Min-W-FS.
- Instance: a learning sample LS of examples described over a set of variables $V = \{v_1, v_2, ..., v_n\}$.
- Solution: a coherent subset V' of V containing weakly relevant features w.r.t. LS.
- Measure: $|V'|$.

Since strong relevance for a variable is not influenced by its peers, we easily obtain the following theorem

Theorem 9. *Minimizing* Min-S-FS *is polynomial.*

We now show that Min-W-FS is much more difficult to approximate.

Theorem 10. *Unless $NP \subset DTIME[n^{\log\log n}]$,* Min-W-FS *is not approximable to within $(1-\epsilon)\log n$ for any $\epsilon > 0$.*

Proof. The reduction is the same as for theorem 5. □

The result of theorem 10 shows that Min-W-FS is hard, but it does not rule out the possibility of efficient feature selection algorithms, since the ratio of inapproximability is quite far from critical bounds of order n^γ (given that the number of features is n). We now show that theorem 10 is also subject to be amplified so that we can effectively remove the possibility of efficient feature selection. Fix some integer value $d > 1$. Suppose we take again the instance of Min-Set-Cover of theorem 5, but we create $|C|^d$ variables instead of the initial $|C|$. Each variable represents now a d-tuple of elements of C. Suppose we number the variables $v_{i_1,i_2,...,i_d}$ with $i_1, i_2, ..., i_d \in \{1, 2, ..., |C|\}$, to represent the corresponding elements of C. The $|S|+1$ old examples are replaced by $|S|^d + 1$ examples described over these variables, as follows:

- for any possible d-tuple $(s_{j_1}, s_{j_2}, ..., s_{j_d})$ of elements of S, we create a positive example $e_{j_1,j_2,...,j_d}$, having ones in variable $v_{i_1,i_2,...,i_d}$ iff
$$\forall k \in \{1, 2, ..., d\}, s_{j_k} \in c_{j_k},$$
and zeroes everywhere else. By this procedure, we create $|S|^d$ positive examples,
- we add the all-zero example, having negative class.

We call LS_d this new set of examples. The reduction time is no more than $\mathcal{O}(|S|^d|C|^d)$. The following lemma is stated under the same hypothesis as for lemma 2.

Lemma 3. *Unless $NP \subset DTIME[n^{\log\log n}]$, provided the reduction is feasible,* Min-W-FS *is not approximable to within*

$$\left(\frac{(1-\epsilon)\log n}{d}\right)^d$$

for any $\epsilon > 0$.

An immediate consequence is the following.

Theorem 11. *Unless $NP \subset QP$, $\exists \delta > 0$ such that* Min-W-FS *is not approximable to within n^{δ}.*

In other words, up to what is be the maximal δ, theorem 11 shows that any non trivial algorithm cannot achieve a significant worst-case approximation of the Min-W-FS problem, with respect to the simple keeping of all variables.

Our second model for feature relevance defines it with respect to the target concept [BL97].

Definition 4. *[BL97] A variable v_i is said to be relevant to the target concept c iff there exists a pair of examples e_A and e_B in the instance space such that their observations differ only in their assignment to v_i and they have a different class.*

From this, [BL97] define the following complexity measure.

Definition 5. *[BL97] Given a sample LS and a set of concept $\mathcal{C}$, $r(LS, \mathcal{C})$ is the number of features relevant using definition 4 to a concept in C that, out of all those whose error over LS is least, has the fewest relevant features.*

We call $\mathcal{C}_{min}(LS)$ to be the set of concepts from $\mathcal{C}$ whose error on LS is least. It is straightforward to check that in definition 5, $r(LS, \mathcal{C})$ defines the optimum of the following minimization problem.

> Name: Min-FS.
> Instance: a learning sample LS of examples described over a set of variables $V = \{v_1, v_2, ..., v_n\}$, a class of concept $\mathcal{C}$.
> Solution: a subset V' of V such that there exists a concept in $\mathcal{C}_{min}(LS)$ which is described over V'.
> Measure: the cardinality of the subset of V' consisting of relevant features according to definition 4.

A result stated in the paper of [BL97] says that Min-FS is at least as hard to approximate as the Min-Set-Cover problem (thus, we get the inapproximability ratio of theorem 1). On the other hand, the greedy set cover algorithm of [Joh74] can be used to approximate $r(LS, \mathcal{C})$ when $\mathcal{C}$ is chosen to be the set of monomials. If we follow [KV94] using a comment of [BL97], the number of variables chosen is no more than

$$r(LS, \text{monomials}) \times \log |LS|,$$

but $|LS|$ can theoretically be as large as 2^n. The question is therefore to what extent we can increase the inapproximability ratio to come as close as possible to the trivial barrier n (we keep all variables). Actually, it can easily be shown that the amplification result of lemma 1 still holds with the reduction allowing to prove the equivalence of Min-Set-Cover and Min-FS. Therefore, we get

Lemma 4. *Unless $NP \subset DTIME[n^{\log \log n}]$, provided the reduction is feasible, then* Min-FS *is not approximable to within*

$$\left(\frac{(1-\epsilon)\log n}{d}\right)^d$$

for any $\epsilon > 0$.

Similarly to theorem 4, we also get as a consequence:

Theorem 12. *Unless $NP \subset QP$, $\exists \delta > 0$ such that* Min-FS *is not approximable to within n^δ.*

References

[ACG$^+$99] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, Marchetti Spaccamela A., and Protasi M. *Complexity and Approximation. Combinatorial Optimization Problems and their Approximability Properties.* Springer-Verlag, Berlin, 1999. 226

[Aro94] S. Arora. Probabilistic checking of proofs and hardness of approximation problems. Technical Report CS-TR-476-94, Princeton University, 1994. 225, 228

[Bel96] M. Bellare. Proof checking and Approximation: towards tight results. *SIGACT news*, 1996. 225, 228

[BFOS84] L. Breiman, J. H. Freidman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth, 1984. 227

[BL97] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, pages 245–272, 1997. 225, 227, 235

[Blu94] A. Blum. Relevant examples and relevant features: Thoughts from computational learning theory. In *AAAI Fall Symposium (survey paper)*, 1994. 224

[CK00] P. Crescenzi and V. Kann. *A Compendium of NP-Optimization problems.* WWW-Available at `http://www.nada.kth.se/∼viggo/wwwcompendium/`, 2000. 226, 230

[HJLT94] T. Hancock, T. Jiang, M. Li, and J. Tromp. Lower bounds on learning decision lists and trees. In *Proc. of the Symposium on Theoretical Aspects of Computer Science*, 1994. 225, 231

[HR76] L. Hyafil and R. Rivest. Constructing optimal decision trees is np-complete. *Inform. Process. Letters*, pages 15–17, 1976. 231

[JKP94] George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In *Proc. of the 11 th International Conference on Machine Learning*, pages 121–129, 1994. 233

[Joh74] D. S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sci.*, pages 256–278, 1974. 226, 235

[KKLP97] V. Kann, S. Khanna, J. Lagergren, and A. Panconesi. On the hardness of approximating MAX k-CUT and its dual. *Chicago Journal of Theoretical Computer Science*, 2, 1997. 225

[KM96] M.J. Kearns and Y. Mansour. On the boosting ability of top-down decision tree learning algorithms. *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, pages 459–468, 1996. 227

[Koh94] R. Kohavi. Feature subset selection as search with probabilistic estimates. In *AAAI Fall Symposium on Relevance*, 1994. 224

[KS95] R. Kohavi and D. Sommerfield. Feature subset selection using the wrapper model: overfitting and dynamic search space topology. In *First International Conference on Knowledge Discovery and Data Mining*, 1995. 224

[KS96] D. Koller and R. M. Sahami. Toward optimal feature selection. In *Proc. of the 13 th International Conference on Machine Learning*, 1996. 224

[KV94] M. J. Kearns and U. V. Vazirani. *An Introduction to Computational Learning Theory.* M.I.T. Press, 1994. 231, 235

[Mit97] T. Mitchell. *Machine Learning.* McGraw-Hill, 1997. 227

[NG95] R. Nock and O. Gascuel. On learning decision committees. In *Proc. of the 12 th International Conference on Machine Learning*, pages 413–420, 1995. 231

[NJ98a] R. Nock and P. Jappy. Function-free horn clauses are hard to approximate. In *Proc. of the Eighth International Conference on Inductive Logic Programming*, pages 195–204, 1998. 225

[NJ98b] R. Nock and P. Jappy. On the power of decision lists. In *Proc. of the 15 th International Conference on Machine Learning*, pages 413–420, 1998. 231

[NJS98] R. Nock, P. Jappy, and J. Sallantin. Generalized Graph Colorability and Compressibility of Boolean Formulae. In *Proc. of the 9 th International Symp. on Algorithms and Computation*, pages 237–246, 1998. 225, 226, 231

[Noc98] R. Nock. *Learning logical formulae having limited size : theoretical aspects, methods and results.* PhD thesis, Université Montpellier II, 1998. Also available as techreport RR-LIRMM-98014. 231

[PR94] K. Pillaipakkamnatt and V. Raghavan. On the limits of proper learnability of subclasses of DNF formulae. In *Proc. of the 7 th International Conference on Computational Learning Theory*, pages 118–129, 1994. 232

[Qui94] J. R. Quinlan. *C4.5 : programs for machine learning.* Morgan Kaufmann, 1994. 227

[Ska94] D. B. Skalak. Prototype and feature selection by sampling and random mutation hill-climbing algorithms. In *Eleventh International Conference on Machine Learning*, pages 293–301, 1994. 224

[SN00a] M. Sebban and R. Nock. Combining feature and prototype pruning by uncertainty minimization. In *Proc. of the 16 th International Conference on Uncertainty in Artificial Intelligence*, 2000. to appear. 224

[SN00b] M. Sebban and R. Nock. Prototype selection as an information-preserving problem. In *Proc. of the 17 th International Conference on Machine Learning*, 2000. to appear. 224

[SS98] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. In *Proceedings of the Eleventh Annual ACM Conference on Computational Learning Theory*, pages 80–91, 1998. 227

[WM97] D. Wilson and T. Martinez. Instance pruning techniques. In *Proc. of the 14 th International Conference on Machine Learning*, pages 404–411, 1997. 224