Complexity in the Case Against Accuracy: When Building One Function-Free Horn Clause Is as Hard as Any

Richard Nock

Department of Mathematics and Computer Science, Université des Antilles-Guyane,
Campus de Fouillole, 97159 Pointe-à-Pitre, France
rnock@univ-ag.fr

Abstract. Some authors have repeatedly pointed out that the use of the accuracy, in particular for comparing classifiers, is not adequate. The main argument discusses the validity of some assumptions underlying the use of this criterion. In this paper, we study the hardness of the accuracy's replacement in various ways, using a framework very sensitive to these assumptions: Inductive Logic Programming. Replacement is investigated in three ways: completion of the accuracy with an additional requirement, replacement of the accuracy by the ROC analysis, recently introduced from signal detection theory, and replacement of the accuracy by a single criterion. We prove strong hardness results for most of the possible replacements. The major point is that allowing arbitrary multiplication of clauses appears to be totally useless. Another point is the equivalence in difficulty of various criteria. In contrast, the accuracy criterion appears to be tractable in this framework.

1 Introduction

As the number of classification learning algorithms is rapidly increasing, the question of finding efficient criteria to compare their results is of particular relevance. This is also of importance for the algorithms themselves, as they can naturally optimize directly such criteria to achieve good results. A criterion frequently encountered to address both problems is the accuracy, which received recently on these topics some criticisms about its adequacy [7].
The primary inadequacy of the accuracy stems from a tacit assumption that the overall accuracy controls by-class accuracies, or similarly that class distributions among examples are constant and relatively balanced [6]. This is obviously not true : skewed distributions are frequent in agronomy, or more generally in life sciences. As an example, consider the human DNA, in which no more than 6% are coding genes [7]. In that cases, the interesting, unusual class is often the rare one, and the well-balanced hypothesis may not lead to discover the unusual individuals. Moreover, in real-world problems, not only is this assumption false, but also of heavy consequences may be the misclassification of some examples, another cost which is not integrated in the accuracy. Fraud detection is a good

O. Watanabe, T. Yokomori (Eds.): ALT'99, LNAI 1720, pp. 182–193, 1999.

example of such situations [7], but medical domains are typical. As an example, consider the case where a mutagen molecule is predicted as non-mutagen, and the case where an harmless molecule is predicted as mutagen. In that cases, the interesting class has the heaviest misclassification costs, and the equal error costs assumption may produce bad results. Finally, the accuracy may be inadequate in some cases because other parameters are to be taken into account. Constraints on size parameters are sometimes to be used because we want to obtain small formulae, for interpretation purposes. As an example, consider again the problem of mutagenesis prediction, where two equally accurate formulae are obtained. If one is much smaller, it is more likely to provide useful descriptions for the mining expert.

We have chosen for our framework a field particularly sensitive to these problems, Inductive Logic Programming (ILP). ILP is a rapidly growing research field, concerned by the use of variously restricted subclasses of Horn clauses to build Machine Learning (ML) algorithms. According to [9], almost seventy applications use ILP formalism, twenty of which are science applications, which can be partitioned into biological (four) and drug design (sixteen) applications. ILP-ML algorithms have been applied with some success in areas of biochemistry and molecular biology [9]. Using ILP formalism, we argue that the replacement of the accuracy raises structural complexity issues. The argument is structured as follows.

First, to address the latter problems, we explain that the single accuracy requirement can be completed by an additional requirement to provide more adequate criteria. We integrate various constraints over two important kinds of parameters: by-class error functions, and representation parameters such as feature selection ratios, size constraints. We show that any of such integration leads to a very negative structural complexity result, similar to NP-Hardness, which is not faced by the accuracy optimization alone. The result has a side effect which can be presented as a "loss" in the formalism's expressiveness, a rare property in classical ML complexity issues. Indeed, it authorizes the construction of arbitrary large (even exponential sized) sets of Horn clauses, but which we prove having no more expressive power than a single Horn clause. We prove a threshold in intractability since it appears immediately with the additional requirement, and is not a function of the tightness of it. Furthermore, the effects of the constraints on optimal accuracies vanish as the number of predicates increases, as optimal accuracies with or without the additional constraints are asymptotically equal. Finally, for some criteria, their mixing with the accuracy brings the most negative result: not only does the intractability appears immediately with the criterion, but also the error cannot be dropped down under that of the unbiased coin. We then study the replacement of the accuracy criterion using a general method [6, 7], derived from statistical decision theory, based on a specific bi-criteria optimization. We show that this method leads to the same drawbacks. Finally, we investigate the replacement of the error by a single criterion, and show that it is also to be analyzed very carefully, as some of the "candidates" lead exactly to the same negative results presented before. The reductions are

presented for a subclass of Horn formalism simple enough to be an element of the intersection of all classically encountered theoretical ILP studies.

2 Mono and Bi-criteria Solutions to Replace the Accuracy

Denote as $\mathcal{C}$ and $\mathcal{H}$ two classes of concepts representations, respectively called *target*'s class and *hypothesis* class. In real-world domains, we do not know the target concept's class, that is why we have to make *ad hoc* choices for $\mathcal{H}$ with a powerful enough formalism, yet ensuring tractability. Even if some benchmarks problems appear to be easily solvable [3], ML applications, and particularly ILP, face more difficult problems [9], for which the choice of $\mathcal{H}$ is crucial. Since most of the studies dealing with the accuracy replacement problem have been investigated with two classes [7], we also consider two-classes problems and not multi-class cases. It is not really important for us, as results already become hard in that setting. Let $c \in \mathcal{C}$. Suppose that we have drawn examples following some unknown but fixed distribution D, labelled according to c. We can denote the accuracy of $h \in \mathcal{H}$ with respect to (w.r.t.) c by $P_D(h = c) = \sum_{h(x)=c(x)} D(x)$.

2.1 Extending the Accuracy

The principal drawbacks of the accuracy are of two kinds: the equal costs assumption [6], and the well balanced assumption [7]. We propose a solution to the problem by the maximization of the accuracy subject to constraints. We also propose criteria on related problems, an example of such being the feature selection problem, in which we want to build formulae on restricted windows of the total features set. For any fixed positive rational ν, we use the following adequate notion of distance between two reals u, v : $d_\nu(u, v) = \frac{|u-v|}{u+v+\nu}$. We also use eight rates on the examples (definitions differ slightly from [7]): $TP = \sum_{h(x)=1=c(x)} D(x)$; $TPR = TP/P$; $FP = \sum_{h(x)=1\neq c(x)} D(x)$; $FPR = FP/N$; $TN = \sum_{h(x)=0=c(x)} D(x)$; $TNR = TN/N$; $FN = \sum_{h(x)=0\neq c(x)} D(x)$; $FNR = FN/P$, with $N = \sum_{c(x)=0} D(x)$ and $P = \sum_{c(x)=1} D(x)$. In order to complete the accuracy requirements, we imagine seven types of additional constraints, each of them being parameterized by a number ζ (between 0 and 1). Each of them defines a subset of $\mathcal{H}$, which shall be parameterized by D if the distribution controls the subset through the constraint. The first three subsets of $\mathcal{H}$ contain hypotheses for which the FP and FN are not far from each other, or a one-side error is upper bounded: $\mathcal{H}_{D,1}(\zeta) = \{h \in \mathcal{H} | d_\nu(FP, FN) \leq \zeta\}; \mathcal{H}_{D,2}(\zeta) = \{h \in \mathcal{H} | FN \leq \zeta\}; \mathcal{H}_{D,3}(\zeta) = \left\{h \in \mathcal{H} | FN \leq \frac{1}{\zeta} FP\right\}$. The two following subsets are parameterized by constraints equivalent to some frequently encountered in the information retrieval community [8], respectively (1 minus) the precision and (1 minus) the recall criteria: $\mathcal{H}_{D,4}(\zeta) = \{h \in \mathcal{H} | FP/(TP + FP) \leq \zeta\}$; $\mathcal{H}_{D,5}(\zeta) = \{h \in \mathcal{H} | FN/(TP + FN) \leq \zeta\}$. Define $\#\mathtt{P}(h)$ as the total number of different predicates of h, $\#\mathtt{W}(h)$ as the whole number of predicates of

h(if one predicate is present k times, it is counted k times), and $\#\mathtt{T}$ as the total number of different available predicates. The two last subsets of $\mathcal{H}$ are parameterized by formulae respectively having a sufficiently small fraction of the available predicates, or having a sufficiently small overall size: $\mathcal{H}_6(\zeta) = \{h \in \mathcal{H} | \#\mathtt{P}(h)/\#\mathtt{T} \leq \zeta\}$; $\mathcal{H}_7(\zeta) = \{h \in \mathcal{H} | \#\mathtt{W}(h)/\#\mathtt{T} \leq \zeta\}$. The division by the total number of different predicates in $\mathcal{H}_7(\zeta)$ is made only for technical reasons: to obtain hardness results for small values of ζ and thus, already for small sizes of formulae (in the last constraint). The first problem we address can be summarized as follows:

Problem 1: Given ζ and $a \in \{1, 2, ..., 7\}$, can we find an algorithm returning a set of Horn clauses from $\mathcal{H}_{(D,)a}(\zeta)$ whose error is no more than a given γ, if such an hypothesis exists ?

2.2 Replacing the Accuracy: The ROC Analysis

Receiver Operating Characteristic (ROC) analysis is a traditional methodology from signal detection theory [1]. It has been used in machine learning recently [6, 7] in order to correct the main drawbacks of the accuracy. In ROC space (this is the coordinate system), we visualize the performance of a classifier by plotting TPR on the Y axis, and FPR on the X axis. Figure 1 presents the ROC analysis, along with three possible outputs which we present and analyze. If a

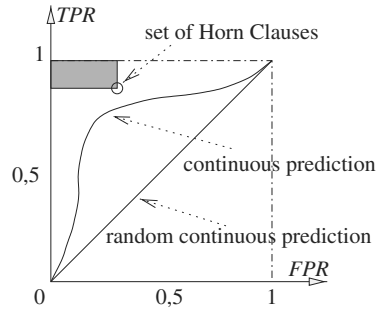

Fig. 1. The ROC analysis of a learning algorithm.

classifier produces a continuous output (such as an estimate of posterior probability of an instance's class membership [7]), for any possible value of FPR, we can get a value for TPR, by thresholding the output between its extreme bounds. If a classifier produces a discrete output (such as Horn clauses), then the classifier gives rise to a single point. If the classifier is the random choice of the class, either (if it is continuous) the curve is the line $y = x$, or (if it is discrete) there is a single dot, on the line $y = x$. One important thing to note is that the ROC representation gives the behavior of an algorithm without regarding the class distribution or the error cost [6]. And it allows to choose the best of some classifiers, by the following procedure. Fix as K^+ the cost

of misclassifying a positive example, and K^- the cost of misclassifying a negative example (these two costs depend on the problem). Then the *expected cost* of some classifier represented by point (FPR, TPR) is given by the following formula: $\sum_{c(x)=1} D(x) \times (1 - TPR) \times K^+ + \sum_{c(x)=0} D(x) \times FPR \times K^-$. Two algorithms, whose corresponding point are respectively (FPR_1, TPR_1) and (FPR_2, TPR_2), have the same expected cost iff $(TPR_2 - TPR_1)/(FPR_2 - FPR_1) = (\sum_{c(x)=1} D(x)K^+)/(\sum_{c(x)=0} D(x)K^-)$. This gives the slope of an isoperformance line, which only depends on the relative weights of the examples, and the respective misclassification costs. Given one point on the ROC, the classifiers performing better are those on the "northwest" of the isoperformance line with the preceding slope, and to which the point belongs. If we want to find an algorithm A performing *surely* better than an algorithm B, we therefore should strive to find A such that its point lies into the rectangle whose opposite vertices are the $(0, 1)$ point (the perfect classification) and B's point (a grey rectangle is shown on the top left of figure 1). From that, the second problem we address is the following (Note the constraint's weakness : the algorithm is required to work only on a *single* point):

Problem 2: Given one point (TPR_x, FPR_x) on the ROC, can we find an algorithm returning a set of Horn clauses whose point falls into the rectangle with opposite vertices $(0, 1)$ and (TPR_x, FPR_x), if such an hypothesis exists ?

2.3 Replacing the Accuracy by a Single Criterion

The question of whether the accuracy can be replaced by a single criterion instead of two (such as in ROC) has been raised in [6]. Some researchers [6] propose the use of the following criterion: $(1 - FPR) \times TPR$. A geometric interpretation of the criterion is the following [6]: it corresponds to the area of a rectangle whose opposite vertices are (FPR, TPR) and $(1, 0)$. The typical isoperformance curve is now an hyperbola. The third problem we address is therefore:

Problem 3: Given γ, can we find an algorithm returning a set of Horn clauses such that $(1 - FPR) \times TPR \geq \gamma$, if such an hypothesis exists ?

3 Introduction to the Proof Technique

We present here the basic ILP notions which we use, with a basic introduction to our proofs. Technical parts are proposed in two appendices.

3.1 ILP Background Needed

The ILP background needed to understand this article can be summarized as follows. More formalization and details are given in [4], but they are not needed here. Given a Horn clause language $\mathcal{L}$ and a correct inference relation on $\mathcal{L}$, an ILP learning problem can be formalized as follows. Assume a background knowledge $\mathcal{BK}$ expressed in a language $\mathcal{LB} \subseteq \mathcal{L}$, and a set of examples $\mathcal{E}$ in

a language $\mathcal{LE} \subseteq \mathcal{L}$. The goal is to produce an hypothesis h in an hypothesis class $\mathcal{H} \subseteq \mathcal{L}$ consistent with $\mathcal{BK}$ and $\mathcal{E}$ such that h and the background knowledge cover all positive examples and none of the negative ones. Sometimes the formalism cannot correctly classify all examples according to the preceding scenario, for the reason that the examples describe a complex concept. We may transform the ILP learning problem to a relaxed version, where we want the formulae to make sufficiently small errors over the examples. The choice of the representation languages for the background knowledge and the examples, and the inference relation greatly influence the complexity (or decidability) of the learning problem. A common restriction for both $\mathcal{BK}$ and $\mathcal{E}$ is to use ground facts. As in [5], we use θ-subsumption as the inference relation (a clause h_1 θ-subsumes a clause h_2 iff there exists a substitution θ such that $h_1\theta \subseteq h_2$ [5, 4]). In order to treat our problem as a classical ML problem, we use the following lemma, which authorizes us to create ordinary examples:

Lemma 1. *[5] Learning a Horn clause program from a set of ground background knowledge $\mathcal{BK}$ and ground examples $\mathcal{E}$, the inference relation being generalized subsumption, is equivalent to learning the same program with θ-subsumption, and empty background knowledge and examples defined as ground Horn clauses of the form $e \leftarrow b$, where $e \in \mathcal{E}$ and $b \in \mathcal{BK}$.*

In the following, we are interested in learning concepts in the form of (sets of) non recursive Horn clauses. It is important to note that all results are still valid when considering propositional, *determinate* or *local* Horn clauses, similarly to the study of [4], to which we refer for all necessary definitions. For the sake of simplicity in stating our results, we sometimes abbreviate "Function free Horn Clauses" by the acronym "*FfHC*".

3.2 Basic Tools for the Hardness Results

Concerning problem 1, fix $a \in \{1, 2, 3, 4, 5, 6, 7\}$. We want to approximate the best concept in $\mathcal{H}_{(D;)a}(\zeta)$ by one still in $\mathcal{H}_{(D;)a}(\zeta)$. However, the best concept in $\mathcal{H}_{(D;)a}(\zeta)$ generally does not have an error equal to the optimal one over $\mathcal{H}$ given D, $opt_{\mathcal{H}_D}(c)$. In fact, it has an error that we can denote $opt_{\mathcal{H}_{(D;)a}(\zeta)}(c) = \min_{h' \in \mathcal{H}_{(D;)a}(\zeta)} \sum_{h(x) \neq c(x)} D(x) \geq opt_{\mathcal{H}_D}(c)$. The goodness of the accuracy of a concept taken from $\mathcal{H}_{(D;)a}(\zeta)$ should be appreciated with respect to this latter quantity. Our results on problem 1 are all obtained by showing the hardness of solving the following decision problem:

Definition 1. *Approx-Constrained($\mathcal{H}, (a, \zeta)$):*
Instance *: A set of negative examples S^-, a set of positive examples S^+, a rational weight $0 < w(x_i) = \frac{n_i}{d_i} < 1$ for each example x_i, a rational $0 \leq \gamma < 1$. We assume that $\sum_{x \in S^+ \cup S^-} w(x_i) = 1$.*
Question *: $\exists ? h \in \mathcal{H}_{(D;)a}(\zeta)$ satisfying $\sum_{h(x) \neq c(x)} w(x) \leq \gamma$?*

Define as n_e the size of the largest example we dispose of. Note that when the constraint is too tight, it can be the case that $\mathcal{H}_{(D;)a}(\zeta) = \emptyset$. Define as $|h|$ the

size of some $h \in \mathcal{H}$ (in our case, it is the number of Horn clauses of h). In the non-empty subset of $\mathcal{H}$ where formulae are the most constrained (*i.e.* strengthening further the constraint gives an empty subset), define $n_{opt_{\mathcal{H}_{(D;)a}(\zeta)}(c)}$ as the size of the smallest hypothesis. Then, our reductions all satisfy $n_{opt_{\mathcal{H}_{(D;)a}(\zeta)}(c)} \leq (n_e)^3$. Note that the constraint makes generally $opt_{\mathcal{H}_{(D;)a}(\zeta)}(c) > opt_{\mathcal{H}_D}(c)$. However, the reductions all satisfy $d_\nu \left(opt_{\mathcal{H}_D}(c), opt_{\mathcal{H}_{(D;)a}(\zeta)}(c) \right) = o(1)$, *i.e.* asymptotic values coincide. In addition, the principal result we get (similar for all other problems) is that we can suppose that the whole time used to *write* the total set of Horn clauses is assimilated to $\mathcal{O}(n_e)$, for any set. By writing time, we mean time of any procedure consisting only in writing down clauses. Examples of such a procedure are "write down all clauses having k literals", or even "write down *all* Horn clauses". Such procedures can be viewed as *for-to*, or *repeat* algorithms. This property authorizes the construction of Horn clause sets having arbitrary sizes, even exponential. Problem 2 is addressed by studying the complexity of the following decision problem.

Definition 2. *Approx-Constrained-ROC($\mathcal{H}, \gamma_{FPR}, \gamma_{TPR}$):*
Instance *: A set of negative examples S^-, a set of positive examples S^+, a rational weight $0 < w(x_i) = \frac{n_i}{d_i} < 1$ for each example x_i, a rational $0 \leq \gamma < 1$. We assume that $\sum_{x \in S^+ \cup S^-} w(x_i) = 1$.*
Question *: $\exists ? h \in \mathcal{H}$ satisfying $1 - FPR \geq 1 - \gamma_{FPR}$ and $TPR \geq \gamma_{TPR}$?*

Concerning problem 3, the reductions study a single replacement criterion Γ, and the following decision problem.

Definition 3. *Approx-Constrained-Single($\mathcal{H}, \Gamma, \gamma$):*
Instance *: A set of negative examples S^-, a set of positive examples S^+, a rational weight $0 < w(x_i) = \frac{n_i}{d_i} < 1$ for each example x_i, a rational $0 \leq \gamma < 1$. We assume that $\sum_{x \in S^+ \cup S^-} w(x_i) = 1$.*
Question *: $\exists ? h \in \mathcal{H}$ satisfying $\Gamma(h) \leq \gamma$?*

4 Hardness Results

Theorem 1. *We have:*
[1] *$\forall 0 < \zeta < 1$, Approx-Constrained(FfHC, $(1, \zeta)$) is Hard, when $\nu < (1 - \zeta)/\zeta$.*
[2] *$\forall 0 < \zeta < \frac{1}{2}$, Approx-Constrained(FfHC, $(2, \zeta)$) is Hard.*
[3] *$\forall a \in \{3, 4, 5, 6, 7\}, \forall 0 < \zeta < 1$, Approx-Constrained(FfHC, (a, ζ)) is Hard.*

At that point, the notion of "hardness" needs to be clarified. By "Hard" we mean "cannot be solved in polynomial time under some particular complexity assumption". The notion of hardness used encompasses that of classical NP-completeness, since we use the results of [2] involving randomized complexity classes. All our hardness results are to be read with that precision in mind.
Due to space constraints, only proof of point [1] is presented in appendix 2; all other results strictly use the same type of reduction. Also, in appendix 1, we sketch the proof that all distributions under which our negative results are

proven lead to trivial positive results for the same problem when we remove the additional constraint, and optimize the accuracy alone. While negative results for optimizing the accuracy itself would naturally hold when considering the additional constraints, we therefore prove that optimizing the accuracy under constraint is a strictly more difficult problem, with non-trivial additional drawbacks. Furthermore, the upperbound error value (γ in def. 1) in constraints 4, 5, 6, 7 can be fixed arbitrarily in $]0, 1/2[$, *i.e.* requiring the Horn clauses set to perform slightly better than the unbiased coin does not make the problem easier. We now show that the classical ROC components as described by [7] lead to the same results as those we claimed for the preceding bi-criteria optimizations. The problem is all the more difficult as the difficulty appears as soon as we choose to use ROC analysis, and is not a function of the ROC bounds.

Theorem 2. *Approx-Constrained-ROC(FfHC, $\gamma_{FPR}, \gamma_{TPR}$) is Hard; the result holds $\forall 0 < \gamma_{FPR}, \gamma_{TPR} < 1$.*

The distribution under which the negative result is proven is an easy distribution for the accuracy's optimization alone, similarly to those of the seven constraints. We now investigate the replacement of the accuracy by a single criterion. The negative result stated in the following theorem is to be read with all additional drawbacks mentioned for the previous theorems. Again, the distribution under which the theorem is proven is easy when optimizing the accuracy alone.

Theorem 3. *$\exists \gamma_{max} > 0$ such that $\forall 0 < \gamma < \gamma_{max}$, the problem Approx-Constrained-Single(FfHC,$(1 - FPR) \times TPR, \gamma$) is Hard.*

(Proof sketch included in appendix 2). As far as we know, $\gamma_{max} \geq \frac{175}{41616}$ (roughly 4.2×10^{-3}), but we think that this bound can be much improved.

5 Appendix 1: The Global Reduction

Reductions are achieved from the NP-Complete problem "Clique" [2], whose instance is a graph A graph $G = (X, E)$, and an integer k. The question is "Does there exist a clique of size $\geq k$ in G?". Of course, "Clique" is not hard to solve for any value of k. The following lemma establishes values of k for which we can suppose that the problem is hard to solve ($\binom{n}{k} = n!/((n-k)!k!)$ is the binomial coefficient):

Theorem 4. *(i) We can suppose that $\binom{k}{2} \leq |E|$, and k is not a constant, otherwise "Clique" is polynomial. (ii) For any $\alpha \in]0, 1[$, "Clique" is hard for the value $k = \alpha|X|$ or $k = |X|^{\alpha}$.*

Proof. (i) is immediate ; (ii) follows from [2]: it is proven that the largest clique size is not approximable to within $|X|^{\beta}$, for any constant $0 < \beta < 1$. Therefore, the graphs generated have a clique number which is either l, or greater than $l \times |X|^{\beta}$, with $l < |X|^{1-\beta}$. Therefore, the decision problem is intractable for values of $k > l$, which is the case if $k = \alpha|X|$ or $k = |X|^{\alpha}$, with $\alpha \in]0, 1[$. □

The structure of the examples is the same for any of our reductions. Define a set of $|X|$ unary predicates $a_1(.), ..., a_{|X|}(.)$, in bijection with the vertices of G. To this set of unary predicates, we add two unary predicates, $s(.)$ and $t(.)$. The inferred predicate is denoted $q(.)$. The choice of unary predicates is made only for a simplicity purpose. We could have replaced each of them by l-ary predicates without changing our proof. Define a set of constant symbols useful for the description of the examples: $\{l_{i,j}, \forall (i,j) \in E\} \cup \{l_1, l_2, l_3, l_4\} \cup \{m_i, \forall i \in \{1, 2, ..., |X|\}\}$. Examples are described in the following way. Positive examples from S^+ are as follows:

$$\forall (i,j) \in E, p_{i,j} = q(l_{i,j}) \leftarrow \wedge_{k \in \{1,2,...,|X|\} \setminus \{i,j\}} a_k(l_{i,j}) \wedge t(l_{i,j}) \quad (1)$$

$$p_1 = q(l_1) \leftarrow \wedge_{k \in \{1,2,...,|X|\}} a_k(l_1) \wedge t(l_1) \quad (2)$$

$$p_2 = q(l_2) \leftarrow a_1(l_2) \quad (3)$$

Negative examples from S^- are as follows:

$$\forall i \in \{1, 2, ..., |X|\}, n_i = q(m_i) \leftarrow \wedge_{k \in \{1,2,...,|X|\} \setminus \{i\}} a_k(m_i) \wedge t(m_i) \quad (4)$$

$$n'_1 = q(l_3) \leftarrow \wedge_{k \in \{1,2,...,|X|\}} a_k(l_3) \wedge s(l_3) \wedge t(l_3) \quad (5)$$

$$n'_2 = q(l_4) \leftarrow \wedge_{k \in \{1,2,...,|X|\}} a_k(l_4) \wedge s(l_4) \quad (6)$$

It comes that $n_{opt_{\mathcal{H}_{(D,)a}(\zeta)}(c)} = \mathcal{O}(|X|^3)$ (coding size of positive examples) and $n_e = \mathcal{O}(|X|)$. Non-uniform weights are given to each example, depending on the constraint to be tackled with. The common-point to all reductions is that the weights of all examples n_j (resp. all $p_{i,j}$) are equal (resp. to w^- and w^+). In each reduction, examples and clauses satisfy:

H$_1$ p_2 is forced to be badly classified.
H$_2$ n'_1 is always badly classified.
H$_3$ $w(n'_2)$ ensures that n'_2 is always given the right class, forcing any clause to contain literal $t(.)$ (When we remove n'_2, we ensure that p_2 is removed too).

Lemma 2. *Any clause containing literal $s(.)$ can be removed.*

Proof. Suppose that one clause contains $s(.)$. Then it can be θ-subsumed by n'_1 and by no other example (even if n'_2 exists, because of **H**$_3$); but n'_1 θ-subsumes any clauses and also the empty clause. Therefore, removing the clause does not modify the value of any criteria based on the examples weights. Concerning the sixth constraint, the fraction of predicates used after removing the clause is at most the one before, thus, if the clause is an element of $\mathcal{H}_6(\zeta)$ before, it is still an element after. The same remark holds for the seventh constraint. □

As a consequence, p_1 is always given the positive class (even by the empty clause!). We now give a general outline of the proof for Problem 1 ; reductions are similar for the other problems. Given $h = \{h_1, h_2, ..., h_l\}$ a set of Horn clauses, we define the set $\mathcal{I} = \{i \in \{1, 2, ..., |X|\} | \exists j \in \{1, 2, ..., l\}, a_i(.) \notin h_j\}$, and we fix $|\mathcal{I}| = k'$. In our proofs, we define two functions taking rational values, $E(k')$ and $F_a(k')$ ($k' \in \{1, 2, ..., |X|\}, a = 1, 2, 3, 4, 5, 6, 7$). They are chosen such that:

- $E(k')$ is strictly increasing, $\sum_{x\in S^+\cup S^-|h(x)\neq c(x)} w(x) \geq E(k')$ and $E(k) = \gamma$.
- $F_a(k')$ is strictly decreasing, is a lowerbound of the function inside $\mathcal{H}_{(D,)a}(\zeta)$, and $F_a(k) = \zeta$ (excepted for $a = 3, F_3(k) = 1/\zeta$).

$\forall a \in \{1,2,3,4,5,6,7\}$, if there exists an unbounded set of Horn clauses $h \in \mathcal{H}_{(D,)a}(\zeta)$ satisfying $\sum_{(x\in S^+\wedge h(x)=0)\vee(x\in S^-\wedge h(x)=1)} w(x) \leq \gamma$, its error rate implies $k' \leq k$ and constraint implies $k' \geq k$. So $|\mathcal{I}| = k' = k$. The interest of the weights is then to force $\binom{k}{2}$ positive examples from the set $\{p_{i,j}\}_{(i,j)\in E}$ to be well classified, while we ensure the misclassification of at most k negative examples of the set $\{n_i\}_{i\in\{1,2,...,|X|\}}$. It comes that these $\binom{k}{2}$ examples correspond to the $\binom{k}{2}$ edges linking the $|\mathcal{I}| = k$ vertices corresponding to negative examples badly classified. We therefore dispose of a clique of size $\geq k$.
Conversely, $\forall a \in \{1,2,3,4,5,6,7\}$, given some clique of size k whose set of vertices is denoted $\mathcal{I}$, we show that singleton $h = q(X) \leftarrow \wedge_{i\in\{1,2,...,|X|\}\setminus\mathcal{I}} a_i(X) \wedge t(X)$ is $\in \mathcal{H}_{(D,)a}(\zeta)$, satisfying $\sum_{(x\in S^+\wedge h(x)=0)\vee(x\in S^-\wedge h(x)=1)} w(x) \leq \gamma$. In this case, $n_{opt_{\mathcal{H}_{(D,)a}(\zeta)}(c)}$ drops down to $\mathcal{O}(n_e)$.
All distributions used in theorems 1 and 3 are such that $w+ < w^-/|X|$, at least for graphs exceeding a fixed constant size. Also, due to the negative examples of weights w^-, if we remove the additional constraints and optimize the accuracy alone, we can suppose that the optimal Horn clause is a singleton: merging all clauses by keeping over predicates $a_j(.)$ only those present in all clauses does not decrease the accuracy. Under such a distribution, the optimal Horn clause necessarily contains all predicates $a_j(.)$, and the problem becomes trivial. The distribution in theorem 2 satisfies $w+ = w^-$. This is also a simple distribution for the accuracy's optimization alone: indeed, the optimal Horn clause over predicates $a_j(.)$ is such that it contains no predicates $a_j(.)$ that does not appear at least in one positive example. If the graph instance of "Clique" is connex (and we can suppose so, otherwise the problem boils down to find the largest clique in one of the connected components), then the optimal Horn clause does not contain any of the $a_j(.)$.

6 Appendix 2: Proofs of Negative Results

6.1 Proof of Point [1], Theorem 1

Weights of positive examples: $w(p_2) = \frac{1}{2(1-\zeta)}\left(\zeta\nu + |X|^2 w^-(1+\zeta)\right)$; $\forall (i,j) \in E$, $w(p_{i,j}) = w^+ = \frac{w^-}{(|X|+k)^2}$; $w(p_1) = \frac{1}{2}\left(1 - \frac{\zeta\nu}{1-\zeta}\right) - \frac{1}{2}\left(w^-|X|^2\left[\frac{1+\zeta}{1-\zeta} + |X| - k\right]\right) - \frac{1}{2}\left(w^+\left[\frac{1-\zeta}{1+\zeta}\left(|E| - \binom{k}{2}\right) + |X|\right]\right)$. Weights of negative examples: $w(n'_2) = 1/2$; $\forall j \in \{1,2,...,|X|\}, w(n_j) = w^- = \frac{1}{|X|^2|E|^2}$; $w(n'_1) = \frac{1}{2}\left(\frac{1-\zeta}{1+\zeta}\left(|E| - \binom{k}{2}\right)w^+\right) + \frac{1}{2}\left((|X|^2 - k)w^-\right)$.
Fix $\gamma = \left(w(p_2) + w(n'_1) + kw^- + \left(|E| - \binom{k}{2}\right)\right)/2$ (note that $w(n'_2)$ ensures that n'_2 is given the right class), and $k_{\max} = 1 + \max_{2\leq k''\leq |X|:|E|-\binom{k''}{2}\geq 0} k''$. From

the choice of weights, $\mathrm{lcm}(\cup_{x_i \in S^+ \cup S^-} d_i) = \mathcal{O}(|X|^8)$ ("lcm" is the least common multiple), which is polynomial. Define the functions: $\forall k' \in \{0,1\}, E(k') = |E|w^+ + k'w^- + w(p_2) + w(n_1)$; $\forall 2 \leq k' \leq k_{\max}, E(k') = \left(|E| - \binom{k'}{2}\right) w^+ + k'w^- + w(p_2) + w(n_1)$; $\forall k_{\max} < k' \leq |X|, E(k') = k'w^- + w(p_2) + w(n_1)$. From the choice of weights, $E(k) = \gamma$. $\forall k' \in \{0,1\}, F_1(k') = ||E|w^+ - k'w^- + w(p_2) - w(n_1)|/q$; $\forall 2 \leq k' \leq k_{\max}, F_1(k') = |\left(|E| - \binom{k'}{2}\right) - k'w^- + w(p_2) - w(n_1)|/q$; $\forall k_{\max} < k' \leq |X|, F_1(k') = |-k'w^- + w(p_2) - w(n_1)|/q$, with $q = \nu + |E|w^+ + k'w^- + w(p_2) + w(n_1)$. From the choice of weights, $F_1(k) = \zeta$.

The equation obtained when $k' < k_{\max}$ takes its maximum for integer values when $k' = (|X| + k)^2 + 0,5 \pm 0,5 > |X|$. Furthermore, $\forall 1 \leq k_{\max} \leq |X|, \left(|E| - \binom{k_{\max}-1}{2}\right) w^+ < w^-$, which leads to $E(k_{\max} - 1) < E(k_{\max})$. In a more general way, $E(k')$ is strictly increasing over natural integers. Now remark that the numerator of $F_1(k')$ is strictly decreasing, and its denominator strictly increasing. Therefore, $F_1(k')$ is strictly decreasing. Furthermore $d_\nu\left(\sum_{h(x)\neq 1=c(x)} w(x), \sum_{h(x)\neq 0=c(x)} w(x)\right) \geq F_1(k')$. If $\exists h \in \mathcal{H}_{\{w_i\},1}(\zeta)$ satisfying $\sum_{h(x)\neq c(x)} w(x) \leq \gamma$, the error rate implies $k' \leq k$ and the constraint implies $k' \geq k$. Thus $|\mathcal{I}| = k' = k$. As pointed out in the preceding appendix, this leads to the existence of a clique of size $\geq k$.

Reciprocally, the Horn clause h constructed in Appendix 1 satisfies both relations $h \in \mathcal{H}_{\{w_i\},1}(\zeta)$, and $\sum_{h(x)\neq c(x)} w(x) \leq \gamma$. Indeed, we have $\sum_{h(x)\neq 1=c(x)} w(x) = \left(|E| - \binom{k}{2}\right) w^+ + w(p_2)$, but also $\sum_{h(x)\neq 0=c(x)} w(x) = kw^- + w(n_1)$. Therefore, $d_\nu\left(\sum_{h(x)\neq 1=c(x)} w(x), \sum_{h(x)\neq 0=c(x)} w(x)\right) = F_1(k) = \zeta$ and $h \in \mathcal{H}_{\{w_i\},1}(\zeta)$. We have also $\sum_{h(x)\neq c(x)} w(x) = E(k) = \gamma$. The reduction is achieved. We end by remarking that $d_\nu\left(opt_{\mathcal{H}_{\{w_i\}}}(c), opt_{\mathcal{H}_{\{w_i\},1}(\zeta)}(c)\right) \leq (|E|w^+ + |X|w^-)/\left(\frac{\zeta\nu}{1-\zeta} + \nu\right)$, which is $o(1)$ (as $|X| \to \infty$ or $|E| \to \infty$), as claimed in subsection 3.2.

6.2 Proof Sketch of Theorem 3

Remark that $TPR(1-FPR) = TPR \times TNR$. Weights are as follows for positive examples (we do not use p_2):

$$\forall (i,j) \in E, w(p_{i,j}) = w^+ = \frac{\gamma}{(|X| - k)w^- \times \left[\binom{k}{2} + \frac{(|X|+1)^2 - \left(k - \frac{|X|+1}{3}\right)^2 - 3|X|}{6}\right]}$$

$w(p_1) = w^+ \times \left((|X|+1)^2 - \left(k - \frac{|X|+1}{3}\right)^2 - 3|X|\right)/6$. Weights are as follows for negative examples (we do not use n_2'): $\forall j \in \{1,2,...,|X|\}, w(n_j) = w^- = 1/(|X| + k)$; $w(n_1') = 1 - |E|w^+ - |X|w^- - w(p_1)$. The choice of γ_{max} comes from the necessity of keeping weights within correct limits. We explain how to the existence of a clique, by describing a polynomial of degree 3, $F(k')$ which upperbounds $TPR \times TNR$, and of course has the desirable property of having its maxi-

mum for $k' = k$, with value γ, and with no other equal or greater values on the interval $[0, |X|]$. Similarly to the other proofs, the value γ can only be reached when $k' = k$ represents k "holes" among predicates $\{a_j(.)\}$, and this induces a size-k clique in the graph. Define the function $F(k')$ as follows. $\forall k' \in \{0, 1\}, F(k') = w(p_1) \times (|X| - k')w^-$; $\forall 2 \leq k' \leq k_{\max}, F(k') = \left(\binom{k'}{2} w^+ + w(p_1)\right)(|X| - k')w^-$; $\forall k_{\max} < k' \leq |X|, F(k') = (|E|w^+ + w(p_1))(|X| - k')w^-$. With our choice of weights, and inside the values of k' for which we described k (clearly, in the second $F(k')$), F describes a polynomial of degree 3, shown in figure 2. F upper-

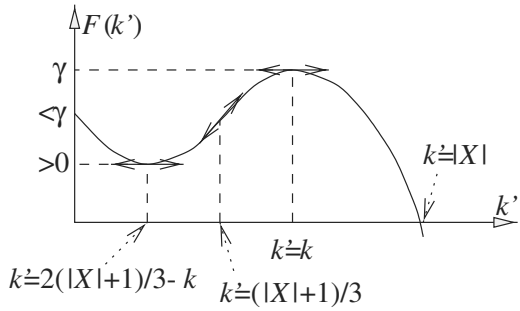

Fig. 2. Scheme of $F(k') = \left(\binom{k'}{2} w^+ + w(p_1)\right)(|X| - k')w^-$.

bounds $TPR \times TNR$ of any set of Horn clauses, and the demand on $TPR \times TNR$ leads to a single favorable case: the "holes" inside the set of Horn clauses describe a clique of size $k' = k$ in the graph.