A Phase Transition-based Perspective on Multiple Instance Kernels

Relational Learning, Multiple Instance Problems, Relational Kernels

Abstract : This paper is concerned with Relational Support Vector Machines, at the intersection of Support Vector Machines (SVM) and Inductive Logic Programming or Relational Learning. The so-called phase transition framework, primarily developed for constraint satisfaction problems (CSP), has been extended to relational learning, providing relevant insights into the limitations and difficulties thereof. The goal of this paper is to examine relational SVMs and specifically Multiple Instance Kernels along the phase transition framework; a specific CSP formalization for multiple instance problems, inspired by chemometry applications, is proposed. Ample empirical evidence based on a set of order parameters shows the existence of an unsatisfiability region for standard MIP-SVM approaches. A statistical analysis for these findings is proposed, establishing a lower bound of the generalization error depending on the satisfiability probability.

Key-words : Phase Transition, Multiple Instance Learning, Relational Kernels, MIP-Support Vector Machine

1 Introduction

This paper is concerned with Relational Support Vector Machines, at the intersection of Support Vector Machines (SVM) (Vapnik, 1998) and Relational Learning (Muggleton & De Raedt, 1994). After the so-called kernel trick, the extension of SVMs to relational representations relies on the design of specific kernels (see (Cuturi & Vert, 2004; Gärtner et al., 2006) among many others). Relational kernels thus achieve a particular type of propositionalization (Kramer et al., 2001), mapping every relational example in the problem domain onto a propositional space defined after the training examples.

However, relational representations intrinsically embed combinatorial issues. For instance the covering test checking whether a relational hypothesis covers an example and usually set to Plotkin's θ-subsumption can be cast as a constraint satisfaction problem (CSP) (Botta et al., 2003). The fact that relational learning involves the resolution of CSPs as a core routine has far-fetched consequences besides exponential (worst-case) complexity.

Indeed in some domains the worst-case complexity is a poor measure of difficulty; this was shown for CSPs since the early 90s (Cheeseman et al., 1991; Hogg et al., 1996). A more accurate perspective, referred to as Phase Transition paradigm, is provided by the statistical computational complexity (more on this in section 2.2). One main result based on the extension of the Phase Transition paradigm to relational learning (Giordana & Saitta, 2000) is to show and explain the failure of existing relational learners in some regions of the PT landscape (Botta et al., 2003).

The question investigated in this paper is whether relational SVMs avoid the limitations of relational learners related to the PT region. This question is examined wrt a particular relational setting, known as the multiple instance problem (MIP) (Dietterich et al., 1997) and considered as an intermediate setting between pure relational and pure propositional formalisms. While MIP-SVM approaches have been applied e.g. for chemometry (Mahé et al., 2006) applications, it is unclear whether they improve over standard relational algorithms on these applications. A related question is how, if MIP-SVMs ever meet difficulties related to the phase transition region, these difficulties can be amplified or alleviated within the propositionalization step.

This paper presents three contributions. Firstly, a set of order parameters is proposed to describe the critical factors of difficulty for multiple instance learning. Secondly, extensive and principled experiments designed after these parameters suggest that MIP-SVMs suffer from the high bias of the hypothesis search space in some regions of the MIP order parameter space. Thirdly, a statistical analysis of these findings is proposed, relating the satisfiability of the multiple instance problem formalized as a CSP, to the generalization error.

The paper is organized as follows. For the sake of self-containedness, the phase transition framework is briefly introduced in Section 2 together with Inductive Logic Programming and Relational Kernels. Section 3 describes the MIP setting and the goal of the MIP-PT study. Section 4 reports on the experimental evidence gathered and the paper ends with some perspective for further research.

2 State of the Art

After a brief discussion about the strengths, weaknesses and evolution of relational learning, this section presents the Phase Transition framework. Multiple Instance problems are finally introduced.

2.1 Relational Learning and Inductive Logic Programming

The last few years have witnessed an increasing demand for machine learning in structured domains where the examples and the sought target concepts can hardly be expressed in a propositional representation. Typical examples of such domains are bioinformatics, chemistry, or natural language processing.

The standard dilemma in relational learning is one of expressiveness and intelligibility versus efficiency (Muggleton & De Raedt, 1994). Although Inductive Logic Programming, at the crossroad of Machine Learning and Logic Programming, offers a principled and elegant framework for seamlessly learning, checking and running the hypotheses learned, ILP algorithms hardly scale up with respect to the size of the dataset and their defects such as noise. While these limitations prompted the development of hybrid logical-probabilistic inductive and deductive frameworks (Kersting & Raedt, 2001) an alternative was offered by the development of relational kernels, as will be seen in section 2.3.

2.2 The Phase Transition Framework

A new combinatoric paradigm has been studied in the Constraint Satisfaction community since the early 90s, motivated by computational complexity concerns: Where are the really hard problems (Cheeseman et al., 1991) ? As noted in the introduction, worst case complexity analysis poorly accounts for the fact that the empirical complexity is low for most CSP instances (in spite of their exponential worst-case complexity). These remarks led to developing the so-called *phase transition framework* (PT) (Hogg et al., 1996), which considers the satisfiability and the resolution complexity of CSP instances as random variables depending on order parameters of the problem instance (e.g. constraint density and tightness).

The phase transition paradigm has been transported to relational machine learning and inductive logic programming (ILP) by (Giordana & Saitta, 2000), motivated by the fact that the covering test most used in ILP (Muggleton & De Raedt, 1994) is equivalent to a CSP. This paradigm was instrumental in identifying and analyzing some limitations of relational learning (Botta et al., 2003) or grammatical inference (Pernot et al., 2005) algorithms.

The PT paradigm is deeply rooted in statistical physics: on one hand the average behavior of elements (particles or CSP instances) is governed by order parameters (e.g. temperature or tightness); on the other hand, this average behavior presents abrupt transitions for some particular values of the order parameters (from ice to liquid, from satisfiable to unsatisfiable). Particularly relevant to ML is the fact that in the landscape defined after the order parameters, one can empirically identify the typical regimes or behaviors of the algorithms, providing insights for the theoretical analysis thereof, and/or for the design of new algorithms (Rückert et al., 2003).

2.3 Multiple Instance Learning and Relational Kernels

First introduced by Dietterich et al (Dietterich et al., 1997), Multiple Instance Learning is viewed as the missing link between relational and propositional learning.

In this setting, each example is a bag of instances and the label of the example is positive iff (at least) one of its instances satisfies the target concept.

The prototypical application illustrating the MIP setting is the musk problem where examples are molecules. A molecule is described as a bag of instances, where each instance corresponds to a different 3D conformation of the molecule (described as a d-dimensional vector). The label of the molecule is positive iff there exists (at least) one conformation in the bag responsible for the musky smell. It is negative iff none of its instances satisfies the target concept; each instance in a negative bag can thus be turned into a negative example (made of a single instance).

Finally, an example or bag of instances can be viewed as a set of literals built on a single predicate symbol (conformation); equivalently, an example is a set of rows in a matrix where the columns are the arguments of the predicate. Let us assume in the rest of this section that the instance space is $\mathbb{R}^d$. Formally, each example $\mathbf{x}_i$ is a set of N_i instances noted $\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,N_i}$ and every $\mathbf{x}_{i,j}$ is a vector in $\mathbb{R}^d$.

Early MIP algorithms intensively relied on the so-called linearity bias assumption, i.e. the fact that one conformation alone can explain the target. Under this assumption, the extra complexity of MIP problems compared to pure propositional problems can be viewed as: finding the *one* instance, in every positive bag, responsible for the label. One possible approach, proposed in (Dietterich et al., 1997), is to search for a hyper-rectangle containing at least one instance of every positive bag, and no instance of the negative bags.

More recently, specific kernels were designed for MIP problems (Gärtner et al., 2006; Cuturi & Vert, 2004; Mahé et al., 2006; Kwok & Cheung, 2007). The basic idea is to define the kernel K of two bags of instances as the average of the kernels k between their instances:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{N_i} \frac{1}{N_j} \sum_{k=1}^{N_i} \sum_{\ell=1}^{N_j} k(\mathbf{x}_{i,k}, \mathbf{x}_{j,\ell}) \tag{1}$$

Note that such kernels do not involve the linearity bias in any way. Indeed, the similarity between two examples does not depend on whether both examples have at least an instance in the target concept; rather, $K(\mathbf{x}_i, \mathbf{x}_j)$ reflects the average similarity between the instances in both bags. The implications of this remark are examined in the next section.

3 Overview

This section describes the position of the problem considered in the paper; after formalizing the relationship between multiple instance learning and constraint satisfaction problems, we propose a set of order parameters in order to support the analytical and empirical study conducted in section 4.

3.1 Position of the problem

Considering the application domain of chemometry (Mahé et al., 2006), let the problem be to predict whether a molecule is bio-active or bio-inactive. After the MIP formalism, every molecule is represented as a set of patterns, e.g. a set of pharmacophore triangles. Every triangle is described from its type (the type of its atoms, represented as a symbol in some alphabet Σ) and a d-dimensional vector in $\mathbb{R}^d$ (e.g. describing the electrical and chemical properties of the triangle).

In all generality, the activity of a molecule might result from one among several causes, i.e. the target concept is disjunctive. This aspect will be discarded in the rest of the paper and left for further study. Assuming that there is a single cause for bio-activity, nevertheless the activity of a molecule might result from the fact that it simultaneously blocks several accepting sites in the biological environment, through different pharmacophore triangles. In such a case, the linearity bias does not hold: in order for the molecule to satisfy the target concept, it needs several instances with different types or properties.

A toy example of MIP with no linearity bias is displayed on Fig. 1. The target concept is the conjunction of four elementary concepts represented as balls; the positive example includes 7 instances (noted $+$) such that: i/ all concept balls are visited (actually some concept balls are visited several times); ii/ some instances do not belong to any concept ball. The negative example includes 4 instances (noted $-$); while two of them visit a concept ball, not all balls in the concept are visited.

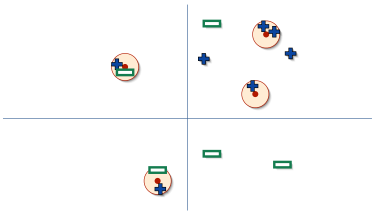


Figure 1: A multiple instance problem with no linearity bias in $\mathbb{R}^2$: the target concept (four balls), one positive example (7 instances noted $+$), one negative example (5 instances noted $-$).

Let the kernel instance be defined as follows. For $\mathbf{x}_i = (a_i, \vec{v}_i)$ and $\mathbf{x}_j = (a_j, \vec{v}_j)$ two instances in the instance space $\Sigma \times \mathbb{R}^d$, $k(\mathbf{x}_i, \mathbf{x}_j)$ is 0 if $\mathbf{x}_i$ and $\mathbf{x}_j$ do not bear the same symbol in Σ ($a_i \neq a_j$), and otherwise, it is defined as the polynomial or Gaussian kernel of $\vec{v}_i$ and $\vec{v}_j$ (Mahé et al., 2006).

As noted in section 2.3, MIP kernels compute the instance kernel value averaged over the pairs of example instances. The question thus is whether the existential information (does a given example $\mathbf{x}$ place an instance in every ball of the target concept) can be reconstructed from the average information available (the average distance between the $\mathbf{x}$ instances and those of every training example $\mathbf{x}_i$, for $\mathbf{x}_i$ ranging over the training set).

3.2 When MIP learning meets CSPs

In order to investigate the above question, one standard procedure is to generate artificial problems, where each problem is made of a training set and a test set, and to compute the test error of the classifier learned by the algorithm under examination from the training set. The test error, averaged over a sample of artificial problems generated after a set of parameter values, indeed measures the competence of the algorithm conditionally to these parameter values (Botta et al., 2003).

A different approach is followed in the present paper, for the following reason. Our goal is to examine how kernel tricks can be used to alleviate the specific difficulties of relational learning; in relational terms, the question is about the quality of the propositionalization achieved through relational kernels. In other words, the focus is on the representation (the capacity of the hypothesis search space defined after the MIP kernel) instead of a particular algorithm (the quality of the best hypothesis retrieved by this algorithm in this search space).

Accordingly, the methodology we followed is based on the generation of artificial problems composed of a training set $\mathcal{L} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ and a test set $\mathcal{T}$ = $\{(\mathbf{x'}_1, y'_1),\ \ldots, (\mathbf{x'}_{n'}, y'_{n'})\}$. The training set $\mathcal{L}$ induces a propositionalization of the domain space, mapping every MIP example $\mathbf{x}$ on the n-dimensional real vector $\Phi(\mathbf{x}) = (K(\mathbf{x}_1, \mathbf{x}), \ldots, K(\mathbf{x}_n, \mathbf{x}))$. Let $\mathcal{R}_{\mathcal{L}}$ denote this propositional representation based on the training set $\mathcal{L}$.

The novelty of the proposed methodology is to handle the MIP learning problem as a constraint satisfaction problem in the $\mathcal{R}_{\mathcal{L}}$ representation.

Specifically, we examine whether there exists a linear form h defined on $\mathbb{R}^n$, with

$$h(\vec{z}) = \sum_{j=1}^{n} \alpha_j y_j z_j + b$$

such that h actually belongs to the search space explored by the MIP-SVM algorithms and h separates the test examples mapped onto $\mathcal{R}_{\mathcal{L}}$, i.e. such that i/ $\alpha_j \geq 0$ for $j = 1 \ldots n$; and ii/ for each test example $(\mathbf{x}, y)$ the sign of $h(\Phi(\mathbf{x}))$ is y.

$$(Q1) \quad \begin{array}{l} \text{Find } \vec{\alpha} = (\alpha_1, \ldots \alpha_n) \in \mathbb{R}^n, b \in \mathbb{R} \\ \text{subject to} \\ \qquad y'_j\,(< \vec{\alpha}, \Phi(\mathbf{x'}_j) > +b) \geq 1, \qquad j = 1 \ldots n' \\ \qquad \alpha_i \geq 0, i = 1 \ldots n \end{array}$$

In other words, the question examined is (Q1) does there exist a separating hyperplane in the propositionalized representation $\mathcal{R}_{\mathcal{L}}$ defined from the training set, which belongs to the search space of MIP-SVMs and which correctly classifies the test set, as opposed to, (Q2) does the separating hyperplane which would have been learned using MIP-SVM algorithms from the training set, correctly classify the test set.

Clearly, (Q1) is much less constrained than (Q2), as one uses the *test* examples (ie, cheats...) in order to find the α_i coefficients. The claim is that CSP (Q1) gives much deeper insights into the quality of the propositionalization based on the kernel trick. Formally, with inspiration from (Kearns & Li, 1993), we show that the satisfiability probability (the percentage of times (Q1) succeeds) induces a lower bound on the generalization error reachable in the representation $\mathcal{R}_{\mathcal{L}}$.

Let p denote the generalization error of the optimal linear classifier h^* defined on $\mathcal{R}_{\mathcal{L}}$, and let $\hat{\tau}$ denote the fraction of (Q1) CSP defined after $\mathcal{L}$ that are satisfiable over N_τ independent test sets.

Proposition

With the above notations, let $\eta > 0$. With probability at least $1 - exp(-2\eta^2\, N_\tau)$,

$$p > 1 - (\hat{\tau} + \eta)^{\frac{1}{n'}}$$

Proof

The probability for a test sample $\mathcal{T}$ of n' examples to include no example misclassified by h^* is $(1-p)^{n'}$.

On the other hand, it is straightforward that if $\mathcal{T}$ does not contain examples that are misclassified by h^*, (Q1) succeeds for $\mathcal{T}$. Therefore the probability τ for (Q1) to succeed is greater than $(1-p)^{n'}$. Using Hoeffding's inequality, the probability τ for a test set

$\mathcal{T}$ to satisfy Q1 can be bounded from $\hat{\tau}$:

$$Pr(|\tau - \hat{\tau}| < \eta) > 1 - exp(-2\,\eta^2\,N_\tau)$$

It comes that with probability $1 - exp(-2\,\eta^2\,N_\tau)$

$$(1-p)^{n'} < \hat{\tau} + \eta$$

which concludes the proof.

3.3 The Order Parameters

As detailed in section 2.2, the Phase Transition framework defines the experimental complexity and other relevant indicators of algorithmic efficiency as random variables depending on the distribution of the problem instances. The distribution of the problems is parametrized after some order parameters, capturing the main factors of difficulty of the task.

Focusing on multiple-instance problems, three types of order parameters respectively devoted to instances, target concept and examples, are defined.

- At the *instance* level, each instance $I = (a, \vec{v})$ is formed of a symbol a drawn in an alphabet Σ, and a d-dimensional vector $\vec{v}$, in $[0,1]^d$. By definition, the ε ball of an instance I denoted $\mathcal{B}_\varepsilon(I)$ includes all instances $I' = (a', \vec{v}')$ such that I and I' bear the same symbol $a = a'$ and the distance $|\vec{v}_k - \vec{v}'_k|$ on each coordinate k of $\vec{v}$ and $\vec{v}'$ is less than ε.
- At the *concept level*, the target concept is characterized as the conjunction of P elementary concepts C_i, where C_i is the ε ball centered on some target instance I_i.
- At the *example level*, a positive (respectively negative) example $\mathbf{x}_i$ is characterized as a set of N^+ (resp. N^-) instances $\mathbf{x}_{i,l}$; example $\mathbf{x}_i$ is positive iff each C_i in the target concept contains at least one instance of $\mathbf{x}_i$.

The instances of the target concept are uniformly drawn in $[0,1]^d$. The N^+ instances of *positive examples* are drawn as follows: P_{ic} instances are drawn in the elementary concepts C_i, ensuring that at least one instance is drawn in every C_i ($P_{ic} \geq P$); $N^+ - P_{ic}$ other instances are uniformly drawn in $[0,1]^d$. Likewise, the N^- instances of *negative examples* involve N_{ic} instances drawn in the elementary concepts C_i, ensuring that nm (near-miss) C_i are not visited ($nm \geq 1$); the other $N^- - N_{ic}$ are randomly drawn in $[0,1]^d$.

Additionally, we introduce the notion of *Universe concept* to model the fact that the example instances are never uniformly drawn. Like the target concept, the Universe concept is made of Q balls with radius ε; the example instances are either sampled in the target concept balls or in the Universe concept balls.

By symmetry with the target concept, we similarly require that some balls of the Universe concept be not visited; the number of Universe balls not visited is set to nm_U.

4 Experiments

After describing the experimental setting, this section reports on the results obtained through an extensive campaign of tests, generating artificial multiple instance problems and observing the percentage of satisfiable problems after the order parameters.

4.1 Experimental setting

The reported experiments consider fixed values of the order parameters related to the instance space and the target concept.
The instance space is $\Sigma \times [0,1]^d$, with $|\Sigma| = 15$, $d = 30$. The target concept is the conjunction of $P = 30$ elementary concepts $B_\varepsilon(I_i)$, where $\varepsilon = .15$ and I_i is uniformly drawn in $[0,1]^{30}$. Every example is a bag of 100 instances ($N^+ = N^- = 100$). Every training set $\mathcal{L}$ includes 30 positive and 30 negative examples ($n = 60$); every test set $\mathcal{T}$ includes 100 positive and 100 negative examples ($n' = 200$).

The number P_{ic} (resp. N_{ic}) of instances in the positive (resp. negative) examples that belong to the concept balls varies in $[30, 100]$ (resp. $[0, 100]$).

The number nm of elementary concepts which are not visited by instances of negative examples varies in $[10, 25]$.

When instances are drawn in a Universe, the Universe is defined by 30 balls on which 15 are not visited by positive examples.

The list of order parameters together with their range of variations is given in Table 1.

$\|\Sigma\|$	15	d	30
P	30	ε	.15
N^+, N^-	100	nm	[10,25]
P_{ic}	[30,100]	N_{ic}	[10, 100]
n	60 (30 $+$, 30 $-$)	n'	200 (100 $+$, 100 $-$)
Q	30	nm_U	15

Table 1: Order parameters for the MIP constraint satisfaction problem and their range of variations

For each set of order parameter values, 40 independent MIP problems are constructed, made of a training set $\mathcal{L}$ and a test set $\mathcal{T}$. The instance kernel is a Gaussian kernel. The associated CSP (Q1) (section 3.2), involving $n' = 200$ constraints and $n + 1 = 61$ variables is constructed, solved using the GLPK package, and the average satisfiability for a set of parameter values is monitored.

The goal of the experiments is to examine how the average satisfiability, and hence the relevance of the MIP propositionalization, depends on the values of the order parameters, specifically focusing on the case where $P_{ic} = N_{ic}$. Does the MIP propositionalization handle the case where the positive and negative examples have a similar number of instances in the elementary concepts, and only differ by the distribution of these instances among the elementary concepts ?

For each experiment, the average satisfiability is displayed in the 2D plane P_{ic}, N_{ic} in black (resp. white) if the fraction of satisfiable CSPs is 0 (resp. 100%).

4.2 Sensitivity analysis wrt Near-miss

The first experiment reports the influence of the near-miss parameter nm, controlling the number of elementary concepts which are not visited by instances of negative examples. As expected, a failure region centered on the diagonal $P_{ic} = N_{ic}$ can be observed, and the failure region increases as the near-miss parameter increases.

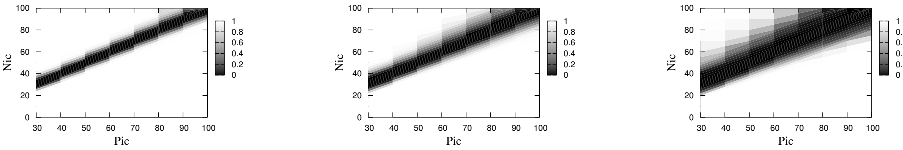


Figure 2: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the near-miss parameter: **Left:** nm = 10. **Center:** nm = 20. **Right:** nm = 25.

These results are explained as follows. The MIP propositionalization maps every example $\mathbf{x}$ onto the n-dimensional vector $\Phi(\mathbf{x})$=$(K(\mathbf{x}_1, \mathbf{x}), \cdots, K(\mathbf{x}_n, \mathbf{x}))$.

Let C (resp. c) denote the mean value of $k(I, I')$ for two instances I and I' belonging to the same elementary concept (resp. drawn uniformly in the instance space). These values depend on both the instance kernel and the instance order parameters d and $|\Sigma|$, which are constant in the experiments.

Let Z_i^+ (respectively Z_i^-) denote the random variable defined as $K(\mathbf{x}_i, \mathbf{x})$, where $\mathbf{x}_i$ is a positive (resp. negative) training example. Z_i^+ and Z_i^- follow Gaussian distributions. Realizations of (Z_i^+, Z_i^-) obtained for positive and negative $\mathbf{x}$, with legend $+$ (resp. $\times$) for positive (resp. negative) examples $\mathbf{x}$ are graphically depicted on Fig. 3.

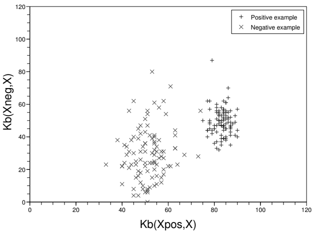


Figure 3: Distribution of $(K(\mathbf{x}_+, \mathbf{x}), K(\mathbf{x}_-, \mathbf{x}))$ for positive examples $\mathbf{x}$ (legend $+$) and negative examples $\mathbf{x}$ (legend $\times$), where $P = 30,\ nm = 20,\ P_{ic} = 50,\ N_{ic} = 30$.

With no difficulty, it is shown that when $\mathbf{x}_i$ and $\mathbf{x}$ are positive, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ is $\frac{P_{ic}^2}{P}(C - c) + c\,{N^+}^2$. Likewise, if both examples are negative, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ is $\frac{N_{ic}^2}{P}(C - c) + c\,N^{-2}$. Last, if both examples belong to different classes, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ is $\frac{P_{ic}N_{ic}}{P}(C - c) + c\,N^{+}N^{-}$.

So when $P_{ic} = N_{ic}$, the distribution of $K(\mathbf{x}_i, \mathbf{x})$ does not depend on the class of $\mathbf{x}$, which clearly hinders the linear discrimination task.

In the general case (when $P_{ic} \neq N_{ic}$), both distributions differ by their average value and by their variance. Still, as the clouds of positive and negative test examples in the propositionalized representation $\mathcal{R}_{\mathcal{L}}$ overlap, their linear separation is only made possible as the number of training examples increases.

Note that although the near-miss parameter nm has no effect on the center of both distributions, it controls their variance. Specifically, when nm increases the N_{ic} instances of the negative examples are concentrated within fewer elementary concepts, increasing the variance of the propositionalization. The larger dispersion of the propositional examples in turn adversely affects the satisfiability of the (Q1) CSP, as shown on Fig. 2.

4.3 Size of the training and test sets

As could have been expected, increasing the number of training examples n makes the failure region to decrease (Fig. 4 (a)). Two interpretations are proposed for the fact that, as usual, more training examples facilitate the learning task. On one hand $-$ provided that $N_{ic} \neq P_{ic}$ $-$, the distance between the centers of the propositionalized positive and negative examples increases proportionally to $\sqrt{n}$, where n is the number of training examples. On the other hand, the more training examples, the more likely one of them will derive a propositional attribute with good discrimination power.

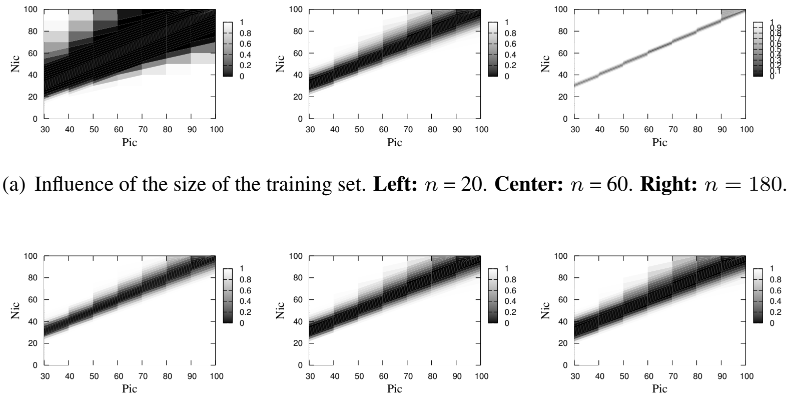
As could have been expected too, the size of the failure region increases with the size of the test set (Fig. 4 (b)). Indeed, the number of constraints in (Q1) is the number of test examples; the probability for the (Q1) CSP to be unsatisfiable thus increases with the number of test examples.

4.4 Sensitivity analysis wrt P_{ic} and N_{ic}

In order to examine the impact of P_{ic} and N_{ic}, complementary experiments are performed by varying the the number of instances in positive and negative training examples.

Firstly, the number of instances in positive (respectively, negative) training examples is uniformly drawn in $[P_{ic} - \Delta, P_{ic} + \Delta]$ (resp. $[N_{ic} - \Delta, N_{ic} + \Delta]$), with Δ varying in [0,10] while the number of instances in test examples is kept fixed.

When Δ increases, it is observed that the size of the failure region decreases (Fig. 5 (a)). The proposed explanation is the same as when the size of the training set increases (second interpretation): the higher variance among the training examples makes it more likely that one of them will derive a propositional attribute with good discrimination power.



(a) Influence of the size of the training set. **Left:** n = 20. **Center:** n = 60. **Right:** $n = 180$.

(b) Influence of the size of the test set. **Left:** n' = 100.**Center:** n' = 200. **Right:** $n' = 400$.

Figure 4: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs.

Secondly, the number of instances for training examples is fixed while the number of instances in positive (respectively, negative) test examples is uniformly drawn in $[P_{ic} - \Delta, P_{ic} + \Delta]$ (resp. $[N_{ic} - \Delta, N_{ic} + \Delta]$), with Δ varying in [0,10]. Here, the failure region increases with Δ (Fig. 5 (b)); this is explained as the higher variance among the test examples makes it more likely to generate inconsistent constraints.

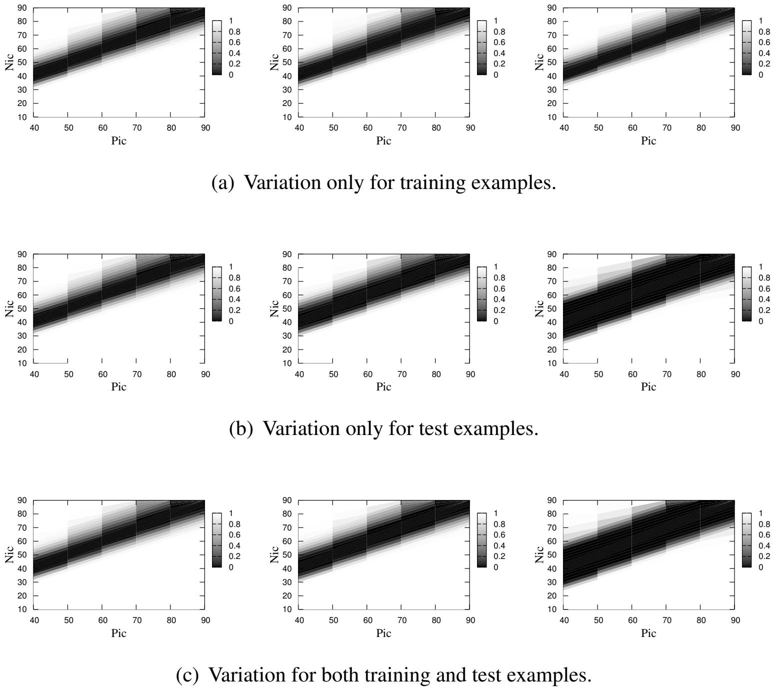
Finally, if the number of instances in all training and test examples varies, the overall effect is to increase the failure region (Fig. 5 (c)): even though there are propositional attributes with better discriminant power, there are more inconsistent constraints too, and the percentage of satisfiable problems decreases.

4.5 Sensitivity Analysis wrt Example size

The impact of the irrelevant instances (not belonging to any elementary target concept) is studied through increasing the example size N^+ and N^-. Experimentally, the failure region increases with N^+ and N^- (Fig. 6). The interpretation proposed for this goes as follows.

On one hand, the distance between positive and negative examples is increasingly due to the influence of irrelevant instances as N^+ and N^- increase. On the other hand, the instances in positive and negative examples are in majority irrelevant when N^+ and N^- increase; therefore the ratio signal to noise in the propositional representation decreases and the failure region increases.

On the other hand, the effect of irrelevant instances is limited as they are far away from each other, comparatively to relevant instances. Therefore increasing the number of irrelevant instances does not much modify $K(\mathbf{x}, \mathbf{x}')$ on average, which explains why the effect of N^+ and N^- appears to be moderate (Fig. 6).



(a) Variation only for training examples.

(b) Variation only for test examples.

(c) Variation for both training and test examples.

Figure 5: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the variability Δ on P_{ic} and N_{ic}. **Left:** Δ = 0. **Center:** Δ = 5. **Right:** Δ = 10.

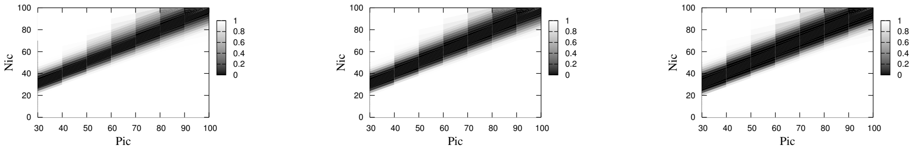


Figure 6: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the size of the examples. **Left:** $N^+ = N^-$ = 100.**Center:** $N^+ = N^-$ = 200. **Right:** $N^+ = N^- = 400$.

4.6 Sensitivity Analysis wrt the Universe Concept

So far, we have supposed that the instances are either drawn from target concept balls or are drawn uniformly outside of these balls.

This section describes how the results change – or more precisely do not change – when the negative instances are in fact drawn in balls defining a Universe concept. It would indeed be conceivable that a non uniform distribution of the negative instances entails different outcomes in learning with MIP kernels.

4.6.1 Effect of the size of the Universe

We suppose here that the Universe concept is made of Q balls, and we measure the effect on learnability of the value of Q.

Of course, a Universe made of a large number Q of uniformly drawn balls should not be different from a uniform Universe. Indeed, the failure region that is observed is similar to the one obtained without Universe. However, when $Q = 30$, corresponding to a Universe of intermediate size, the failure region becomes larger than without a Universe concept. The effect is even more pronounced for small values (Q=5) (Fig. 7).

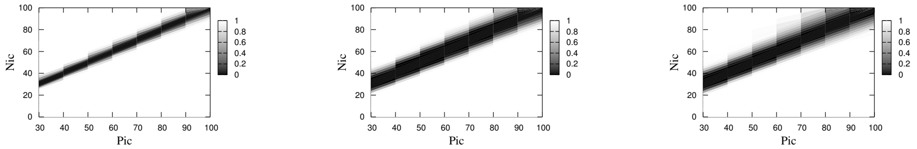


Figure 7: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the size Q of the Universe when nm_U = 0. **Left:** Q = 5. **Center:** Q = 30. **Right:** Q = 1000.

Indeed, as the instances are drawn in a Universe consisting of Q concept balls, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ when $\mathbf{x}$ and $\mathbf{x}_i$ are positive becomes $(C-c)P\frac{Q}{P+Q}(\frac{N^+-P_{ic}}{Q} - \frac{P_{ic}}{P})^2 + {N^+}^2(C\frac{1}{Q+P} + c(1 - \frac{1}{Q+P}))$. Likewise, if both examples are negative, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ is $(C-c)P\frac{Q}{P+Q}(\frac{N^--Q_{ic}}{Q} - \frac{Q_{ic}}{P})^2 + {N^-}^2(C\frac{1}{Q+P} + c(1 - \frac{1}{Q+P}))$. Last, if both examples belong to different classes, the expectation of $K(\mathbf{x}_i, \mathbf{x})$ is $(C-c)P\frac{Q}{P+Q}(\frac{N^--Q_{ic}}{Q} - \frac{Q_{ic}}{P})(\frac{N^+-P_{ic}}{Q} - \frac{P_{ic}}{P}) + N^+N^-(C\frac{1}{Q+P} + c(1 - \frac{1}{Q+P}))$.

Note that, as the size of the Universe approaches ∞, these expressions tend to be the same as when there is no Universe (section 4.2).

Thus, when $N^+ = N^-$ the absolute value of the gap between the expectation of Z_i^+ for a positive example $\mathbf{x}$ and the expectation of Z_i^+ for a negative example $\mathbf{x}$ changes from $(C-c)|P_{ic} - N_{ic}|\frac{P_{ic}}{P}$ (without Universe) to $(C-c)|P_{ic} - N_{ic}|(\frac{N^+-P_{ic}}{Q} - \frac{P_{ic}}{P})$ (with Universe). The ratio of both values follows the curves given by Fig. 8.

When the size of the Universe is huge, the observed values of P_{ic} are large compared to the poi, and the results are similar to those observed without a Universe concept.

When the Universe and the Target concept contain approximately the same number of elements, the observed values of P_{ic} are around $N^+\frac{P}{Q+P}$ and therefore the ratio is smaller than 1. Thus the gap between the expectations is smaller with a Universe than without, and consequently the failure region is larger.

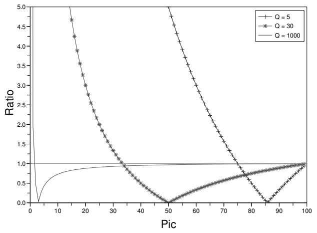


Figure 8: Ratio of the difference between the expectation of Z_i^+ for a positive example $\mathbf{x}$ and the expectation of Z_i^+ for a negative example $\mathbf{x}$ with a Universe by the same difference without Universe. P = 30, N^+ = 100. Q takes three values: 5 (curve legends with '+'), 30 (curve legends with '+') and 1000 (simple curve).

Last, when the Universe contains few elements, most of the values of P_{ic} are smaller than $N^+ \frac{P}{2Q+P}$ (the point for which the ratio is 1) and the ratio is greater than 1. The failure region is then smaller.

4.6.2 Effect of the near miss factor of the Universe

The number of near-miss nm (concept balls not visited by the negative instances) and the number nm_U (negative instances drawn from the Universe concept) have similar effects. They do not change the expectation of $\Phi(\mathbf{x})$ but change its variance: when nm_U increases, the variance of $\Phi(\mathbf{x})$ increases too, and consequently less CSP are satisfiable.

However, one can observe that nm has more effect for large values of P_{ic} and N_{ic} (Fig. 2) while nm_U makes the size of the failure region to increase for small values of P_{ic} and N_{ic} (Fig. 9).

This is reasonable as nm deals with the P_{ic} (resp. N_{ic}) instances in the target concept while nm_U acts on the $N^+ - P_{ic}$ (resp. $N^- - N_{ic}$) instances drawn in the universe.

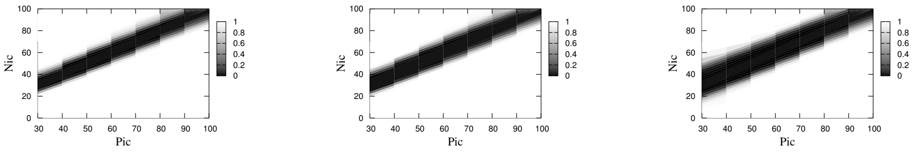


Figure 9: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the size of the near-miss factor of the Universe. **Left:** nm_U = 0. **Center:** nm_U = 15. **Right:** nm_U = 25.

4.6.3 Effect of the Universe on results given by other parameters

The introduction of the Universe does not change the global effect of other parameters. However it increases strongly the effect due to the size of the examples (Fig. 10).

Contrary to the test without Universe (section 4.5), here the uninformative instances are close enough on average to have an influence on the values taken by the kernel K. Therefore, their effect can be easily observed on the curves: the size of the failure region increases with the number of instances of examples.

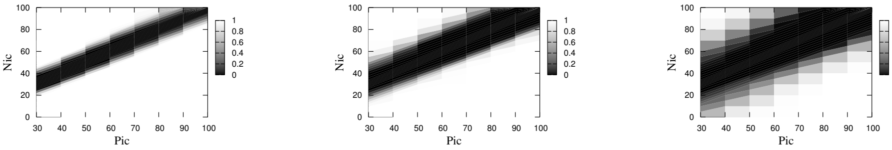


Figure 10: Fraction of satisfiable CSP (Q1) in plane P_{ic}, N_{ic} out of 40 runs. Influence of the size of the example using a Universe. **Left:** $N^+ = N^-$ = 100. **Center:** $N^+ = N^-$ = 200. **Right:** $N^+ = N^-$ = 400.

5 Discussion and Perspectives

The contribution of the paper is i/ to define a relaxed version of the MIP-SVM problem as a Constraint Satisfaction Problem; ii/ to establish the link between the satisfiability of this CSP, and the generalization error of the MIP problem; and iii/ to show that there exists indeed a region in the order parameter landscape where the CSP is not satisfiable.

Clearly, some care must be exercised to interpret the limitations of the well-founded MIP-SVM algorithms suggested by our experiments on artificial problems.

Still, the question of whether MIP-SVM algorithms enable to characterize *existential* properties as opposed to *average* properties makes sense in a relational perspective. Actually, in some domains where the number and/or the diversity of the available examples are limited, as in the domain of chemometry, one might learn average properties, these might do well on the test set, and still be poorly related to the target concept; some evidence for the possibility of such a phenomenon was presented in (Botta et al., 2003), where the test error could be 2% or lower although the concept learned was a gross overgeneralization of the true target concept.

A research perspective opened by this work is based on the further investigation of the CSP, hybridizing the CSP resolution and the kernel-based propositionalization.

References

Botta, M., Giordana, A., Saitta, L., & Sebag, M. (2003). Relational learning as search in a critical region. *Journal of Machine Learning Research*, *4*, 431–463.

Cheeseman, P., Kanefsky, B., & Taylor, W. (1991). Where the really hard problems are. *Proc. of Int. Joint Conf. on Artificial Intelligence* (pp. 331–337)

Cuturi, M., & Vert, J.-P. (2004). Semigroup kernels on finite sets. *NIPS04* (pp. 329–336).

Dietterich, T., Lathrop, R., & Lozano-Perez, T. (1997). Solving the multiple-instance problem with axis-parallel rectangles. *Artificial Intelligence*, *89 (1-2)*, 31–71.

Giordana, A., & Saitta, L. (2000). Phase transitions in relational learning. *Machine Learning*, *41*, 217–251.

Gärtner, T., Flach, P. A., Kowalczyk, A., & Smola, A. J. (2006). Multi-instance kernels. *Proc. ICML02* (pp. 179–186).

Hogg, T., Huberman, B., & (Eds), C. W. (1996). *Artificial intelligence: Special issue on frontiers in problem solving: Phase transitions and complexity*, vol. 81(1-2). Elsevier.

Kearns, M., & Li, M. (1993). Learning in the presence of malicious errors. *SIAM J. Comput.*, *22*, 807–837.

Kersting, K., & Raedt, L. D. (2001). Bayesian logic programs. *Proc. of the 11th Int. Conf. on Inductive Logic Programming*.

Kramer, S., Lavrac, N., & Flach, P. (2001). Propositionalization approaches to relational data mining. In S. Dzeroski and N. Lavrac (Eds.), *Relational data mining*, 262–291. Springer Verlag.

Kwok, J., Cheung, P.-M. (2007). Marginalized Multi-Instance Kernels. *Proc. of the 20th Int. Joint Conf. on Aritificial Intelligence* , 2007, 901–906.

Mahé, P., Ralaivola, L., Stoven, V., & Vert, J.-P. (2006). The pharmacophore kernel for virtual screening with support vector machines. *Journal of Chemical Information and Modeling*, *46*, 2003–2014.

Muggleton, S., & De Raedt, L. (1994). Inductive logic programming: Theory and methods. *Journal of Logic Programming*, *19*, 629–679.

Pernot, N., Cornuéjols, A., & Sebag, M. (2005). Phase transitions within grammatical inference. *Proc. Int. Conf. on Artificial Intelligence* (pp. 811–816). IOS Press.

Rückert, U., Kramer, S., & De Raedt, L. (2003). Stochastic local search in k-term dnf learning. *Proc. of the Int. Conf. on Machine Learning* (pp. 648–655). AAAI Press.

Vapnik, V. N. (1998). *Statistical learning theory*. Wiley.