Asymmetrical Semi-Supervised Learning and Prediction of Disulfide Connectivity in Proteins

Christophe N. Magnan

Laboratoire d'Informatique Fondamentale (LIF)
UMR CNRS 6166, Université de Provence
christophe.magnan@lif.univ-mrs.fr

ABSTRACT. *This paper presents a study in the asymmetrical semi-supervised learning framework, where only positive and unlabeled data are available, and an application to a bio-data processing problem. We show that under very mild assumptions, the Naive Bayes classifier can be identified from positive and unlabeled data. From this study, we derive algorithms that we experiment on artificial data. Lastly, we present an application of this work to the problem of the extraction of local affinities in proteins for the prediction of disulfide connectivity.*

RÉSUMÉ. *Cet article présente une étude en apprentissage automatique semi-supervisé asymétrique, où seules des données positives et non étiquetées sont disponibles, ainsi qu'une application à un problème bio-informatique. Nous montrons que sous des hypothèses faibles, le classifieur naïf de Bayes peut être identifié à partir de données positives et non étiquetées. Nous en déduisons des algorithmes que nous étudions sur des données artificielles. Enfin, nous présentons une application de ces travaux au problème de l'extraction d'affinités locales dans les protéines pour la prédiction des ponts disulfures. Les résultats permettent d'étayer une hypothèse sur la manière de formaliser les données biologiques pour des cas d'interactions physiques locales.*

KEYWORDS: *Semi-supervised learning, Naive Bayes classifier, EM, Disulfide bridges*

MOTS-CLÉS : *apprentissage semi-supervisé, algorithme naïf de Bayes, EM, ponts disulfures.*

RSTI - RIA — 20/2006. New Methods in Machine Learning, pages 673 to 695

1. Introduction

In this paper, we consider the problem of learning in the asymmetrical semi-supervised framework. This problem we tackle is also sometimes referred to the problem of learning from positive and unlabeled data. This particular context of learning suppose that we deal with binary learning problems and that available data consist in unlabeled data and data from only one of the two classes, called the positive class (the other one is called the negative one).

Let X be a discrete feature space and let $Y = \{0, 1\}$ be the set of classes, where 1 denotes the positive class and 0 denotes the negative one. The classical statistical learning framework assumes the existence of an underlying probability distribution P over $X \times Y$ and that the available examples are elements of $X \times Y$ independently drawn according to P.

This distribution P determines:

$$P(x) = \sum_{y \in Y} P(x, y) \text{ for any } x \in X$$

$$P(y) = \sum_{x \in X} P(x, y) \text{ for any } y \in Y$$

$$P(x|y) = \frac{P(x, y)}{P(y)} \text{ and } P(y|x) = \frac{P(x, y)}{P(x)}$$

$$\text{where respectively } P(y) \neq 0 \text{ and } P(x) \neq 0 \qquad [1]$$

In the asymmetrical semi-supervised framework, it is supposed that the available examples are:

– positive examples drawn according to $P(x|1)$

– unlabeled examples drawn according to $P(x)$

This context of learning is intermediary between the classical semi-supervised learning, where positive, negative and unlabeled data are available, randomly drawn from distributions $P(x|1)$, $P(x|0)$ and $P(x)$, and the unsupervised learning where only estimates of the distribution $P(x)$ are available through unlabeled examples: positive and unlabeled data provide less information than data in semi-supervised learning but more than unlabeled data. On the one hand, the classical results on semi-supervised learning context cannot be used because they assume some knowledge on the data distribution over the negative class. On the other hand, positive and unlabeled examples must provide more information than only unlabeled examples, so estimate accuracy and classification performances should be improved.

As in the classical learning framework, the goal is to compute from the data a classifier $f : X \to Y$ which minimizes the prediction risk, $R(f) = P(f(x) \neq y)$. Unfortunately, it can be easily shown that the distributions $P(x)$ and $P(x|1)$ on X

do not determine the distribution P over $X \times Y$. As a consequence, the best classifier cannot be inferred from positive and unlabeled data even if we had a complete knowledge of $P(x)$ and $P(x|1)$.

However, if we know that P belongs to some restricted class of distributions, it becomes possible that $P(x)$ and $P(x|1)$ determine the distribution P. An example of such a situation occurs when it is known that P is deterministic, *i.e.* $P(x, 0) = 0$ or $P(x, 1) = 0$ for all $x \in X$.

In this paper, we mainly study the case where it is known that P satisfies the Naive Bayes assumption: the attributes x^i of x are independent conditionally to each class ($P(x|y) = \prod P(x^i|y)$). The Naive Bayes rule (*cf* Section 2.1), which assumes that target distribution follows this assumption, is well known to give pretty good results in classification even if the assumption is not met (Domingos *et al.*, 1996).

In (Geiger *et al.*, 2001), it was shown that distributions which satisfy Naive Bayes assumption can be identified, up to a permutation of the classes, by unlabeled data when the number of attributes is at least three. Analytical formulas are provided in this paper. However, they do not take into account information provided by positive examples and we show in Section 4.2 that their approach requires a huge number of examples to obtain accurate estimates (*cf* Section 4.2), which makes it difficult to use these formulas in practice.

It is clear that if Naive Bayes distributions are identifiable from unlabeled data, they are identifiable from positive and unlabeled data. Nevertheless, we show in this paper that Naive Bayes distributions can be determined from positive and unlabeled data under slightly milder conditions than from only unlabeled data. Moreover, we give new analytical formulas to estimate target distribution from data. These formulas provide consistent estimators for these parameters. Experimental study (*cf* Section 4.2) shows that these estimators are significantly more accurate than those given in (Geiger *et al.*, 2001) and requires much less examples.

In (McCallum *et al.*, 1999), the authors propose a method based on EM (*cf* Section 2.3) to compute Naive Bayes distribution parameters in the classical semi-supervised framework based on a maximum likelihood criterion (*cf* Section 2.2). This method supposes the knowledge of the distribution $P(x^i|y = 0)$. In the asymmetrical semi-supervised context this distribution is however unknown. We propose a variant of this method to compute target distribution parameters with the criterion of the maximum likelihood (*cf* Section 2.2) using EM (*cf* Section 2.3) in the asymmetrical semi-supervised context.

In order to compare the rate of convergence of these estimators and their performances for learning tasks to the estimators defined by (McCallum *et al.*, 1999) for the semi-supervised context and by (Geiger *et al.*, 2001) for learning from unlabeled data, we carry out an experimental study on artificial data (Section 4). The obtained results show that learning from positive and unlabeled data provides significantly better performances than learning from unlabeled data and show that the results are very similar to the results obtained in the classical semi-supervised learning where labeled exam-

ples from both classes are available. It is a very interesting result because it shows that the loss of negative examples does not decrease performances.

This study was originally motivated by the problem of the prediction of disulfide connectivity in proteins (Section 5). Predicting proteins tridimensional structure, from their sequence of amino acids, is one of the challenges of the current researches in bioinformatics. This structure is constrained by different kinds of interactions: physical, electrostatic, etc. Correctly predicting these interactions should considerably reduce the number of potential structures for the proteins.

In this study, we consider the prediction of disulfide bonds which are the strongest of these interactions. This phenomena of local interaction arise between two distant amino acids - cysteines - after their oxidation. Many papers present methods to predict whether a cysteine is bonded, but few of them address the second part of the problem: predict which cysteine will form a bridge with a given cysteine. We are interested by this part of the problem.

Most of the contributions on the prediction of the disulfide connectivity in proteins (Fariselli *et al.*, 2001, Fariselli *et al.*, 2002, Vullo *et al.*, 2004) consider pairs of non bonded cysteines as negative examples, which cannot form a bond. Our main contribution on that topic is to consider these examples, pairs of non bonded cysteines, as unlabeled examples because we suppose that there is not enough information that would explain the lack of interaction. In this case, data are positive (pairs of bonded cysteines) and unlabeled (pairs of non bonded cysteines): asymmetrical semi-supervised learning methods are necessary. Our experiments show that considering that unbounded cysteines are unlabeled examples is a better strategy than considering that they constitute negative examples.

The paper is organized as follows. In Section 2, we give a short survey of methods and results about the Naive Bayes classifier, EM method and the identifiability problem. In Section 3, we study the identifiability of Naive Bayes models from positive and unlabeled data. We show that the identifiability problem is well posed and we give analytical formulas to compute Naive Bayes models for this context of learning. We also present an algorithm to estimate model parameters based on the maximum likelihood principle using EM. We give in Section 4 experimental results on artificial data which make it possible to conclude on the rate of convergence of our algorithms and their performances for classification tasks. Section 5 presents experimental results on biological data for the problem of the prediction of disulfide connectivity in proteins.

2. Preliminaries

Section 2.1 presents the Bayes rule, the Naive Bayes assumption and the associated classifiers. We give in Section 2.2 a short description of the maximum likelihood principle and its application to the Naive Bayes classifiers in supervised and semi-supervised contexts. Section 2.3 presents the EM method (Expectation Maximization) and the application of this method proposed in (McCallum *et al.*, 1999) to compute

Naive Bayes models parameters in semi-supervised context with the maximum likelihood criterion. Lastly, we give in Section 2.4 a short survey of the paper (Geiger *et al.*, 2001). This work shows Naive Bayes models identifiability from unlabeled data and provide analytical formulas to compute models parameters.

2.1. *The Bayes rule and the Naive Bayes classifier*

Let $X = \prod_{i=1}^{m} X^i$ be a domain defined by m symbolic attributes. For all $x \in X$, let us denote by x^i the projection of x on X^i and let us denote by $Dom(x^i)$ the set of possible values of x^i. Let P be a probability distribution over X and let Y be a set of classes ($Y = \{0, 1\}$ all along this paper) provided with conditional probability distributions $P(y|x)$ for all $x \in X$.

The optimal decision rule for assigning each object $x \in X$ to a class is the *Bayes rule* C_{Bayes} that selects the class $y \in Y$ with the highest probability knowing x.

$$\begin{aligned} C_{Bayes}(x) &= \operatorname*{argmax}_y \ P(y|x) \\ &= \operatorname*{argmax}_y \ P(x|y) \cdot P(y) \ \ (x \in X, y \in Y) \end{aligned} \quad [2]$$

The Bayes classifier requires complete knowledge of the underlying probability distribution. It is the reason why it is generally not possible to estimate this classifier without complementary information or hypotheses.

When the attributes are independent conditionally to each class, that is, the Naive Bayes assumption is met, then $P(x|y) = \prod_{i=1}^{m} P(x^i|y)$. In such a case, the number of parameters to be estimated is low: $O(dm)$ where $d = \max \ |Dom(x^i)|$. The Bayes classifier becomes the Naive Bayes classifier C_{NB}, defined by:

$$C_{NB}(x) = \operatorname*{argmax}_y \ P(y) \prod_{i=1}^{m} P(x^i|y) \ \ (x \in X, y \in Y) \quad [3]$$

The assumption of independence is generally not satisfied. However, Naive Bayes classifier is known to give pretty good results for classification tasks (Domingos *et al.*, 1996).

When $Y = \{0, 1\}$, Naive Bayes classifiers are completely specified by the following set of parameters: $\alpha = P(y = 1)$ and $\lambda_{ikj} = P(x^i = k|y = j)$ where

$1 \leq i \leq m$, $k \in Dom(x^i)$ and $j \in \{0,1\}$. An instance $\theta = \{\alpha, \lambda_{ikj}, i \in [1,...,m], j \in \{0,1\}, k \in Dom(x^i)\}$ of these parameters is called a *model*.

2.2. *The maximum likelihood principle*

Let $S = \{(x_s, y_s), s = 1, ..., l\}$ be a set of independent and identically distributed data according to the joint probability distribution $P(x,y) = P(x) \cdot P(y|x)$ and let θ be a model. One quality criterion of the model θ for a dataset S is the likelihood.

The *likelihood* $L(\theta, S)$ (resp. the *log-likelihood* $l(\theta, S)$) of S for the model θ is defined by:

$$L(\theta, S) = \prod_{s=1}^{l} P(x_s, y_s|\theta) \text{ and } l(\theta, S) = log\ L(\theta, S) \quad [4]$$

The maximum likelihood principle recommends to choose a model θ which maximizes $L(\theta, S)$ – and thus also $l(\theta, S)$.

2.2.1. *Likelihood of Naive Bayes models in supervised context*

Let n_0 (resp. n_1) denote the number of examples classified '0' (resp. '1') in S, $n_0 + n_1 = l$, and let n_{ij}^k denote the number of examples (x, y) in S such that $x^i = k$ and $y = j$. We have:

$$\begin{aligned} L(\theta, S) &= \prod_{s=1}^{l} P(y_s) \left[\prod_{i=1}^{m} P(x_s^i|y_s) \right] \\ &= \alpha^{n_1} \cdot (1-\alpha)^{n_0} \cdot \prod_{\substack{1 \leq i \leq m, 0 \leq j \leq 1 \\ k \in Dom(x^i)}} \lambda_{ikj}^{n_{ij}^k} \end{aligned} \quad [5]$$

$$\begin{aligned} l(\theta, S) &= log\ L(\theta, S) \\ &= n_1 log\alpha + n_0 log(1-\alpha) + \sum_{\substack{1 \leq i \leq m, 0 \leq j \leq 1 \\ k \in Dom(x^i)}} n_{ij}^k log\ \lambda_{ikj} \end{aligned} \quad [6]$$

It can be shown that $L(\theta, S)$ is maximal when:

$$\begin{aligned} \alpha &= \frac{n_1}{n_0 + n_1} \\ \lambda_{ij}^k &= \frac{n_{ij}^k}{\sum_{r \in Dom(x^i)} n_{ij}^r} \end{aligned} \quad [7]$$

2.2.2. *Likelihood of Naive Bayes models in a semi-supervised context*

In the semi-supervised learning context, two datasets are available: $S_{lab} = \{(x_1, y_1), ..., (x_l, y_l)\}$ is a set of labeled data, and $S_{unl} = \{x'_1, ..., x'_{l'}\}$ is a set of unlabeled data.

We suppose that S_{lab} and S_{unl} have been provided by an oracle which, with probability β, draws a labeled example and with probability $1 - \beta$ draws an unlabeled example. Let $\theta' = \theta \cup \{\beta\}$.

The probabilities to draw a labeled example $z = (x, y)$, or an unlabeled example $z = x$ with the model θ' are computed as follows:

$$\begin{aligned} P(z = (x, y)|\theta') &= \beta \cdot P(x, y|\theta) \\ P(z = x|\theta') &= (1 - \beta) \cdot P(x|\theta) \end{aligned} \quad [8]$$

with

$$P(x|\theta) = P(y = 1|\theta)P(x, y|y = 1, \theta) + P(y = 0|\theta)P(x, y|y = 0, \theta) \quad [9]$$

The likelihood can be written:

$$L(\theta', S_{lab}, S_{unl}) = \prod_{s=1}^{l} \beta P(x_s, y_s|\theta) \prod_{r=1}^{l'} (1 - \beta) P(x'_r|\theta) \quad [10]$$

With notations defined on previous section:

$$L(\theta', S_{lab}, S_{unl}) = \beta^l L(\theta, S_{lab})(1 - \beta)^{l'} L(\theta, S_{unl}) \quad [11]$$

with

$$L(\theta, S_{unl}) = \prod_{r=1}^{l'} \left(\alpha \prod_{\substack{1 \leq i \leq m \\ k/x_r^i = k}} \lambda_{ik1} + (1 - \alpha) \prod_{\substack{1 \leq i \leq m \\ k/x_r^i = k}} \lambda_{ik0} \right) \quad [12]$$

The value of β which maximizes the likelihood is $\beta = \frac{l}{l+l'}$, *i.e.* the proportion of labeled examples in the learning set. Nevertheless, the parameters α and λ_{ij}^k which maximize $L(\theta', S)$ cannot be computed using analytical formulas. However, they can be estimated using methods such as E.M. (*cf.* Section 2.3).

2.3. *Expectation-Maximization method (E.M.)*

The EM method was elaborated in (Dempster *et al.*, 1977) for inference of mixture models densities. This section presents a short survey of this method and an application to Naive Bayes models estimate in the semi-supervised context (McCallum *et al.*, 1999).

2.3.1. *Method*

This section describes the E.M. method following (Hastie *et al.*, 2001). Let θ' a model as previously defined, Z the set of observed data, Z_m the missing data and T the entire set of the data, $T = (Z, Z_m)$. Let us denote by:

– $l_0(\theta', T)$ the log-likelihood of T for the model θ',

– $l_1(\theta', Z_m|Z)$ the log-likelihood of Z_m for the model θ' knowing Z,

– $l(\theta', Z)$ the log-likelihood of Z for the model θ',

then $l(\theta', Z) + l_1(\theta', Z_m|Z) = l_0(\theta', T)$, that is to say:

$$l(\theta', Z) = l_0(\theta', T) - l_1(\theta', Z_m|Z) \quad [13]$$

Assuming that the data are drawn according to θ and that Z is observed, previous equality terms are random variables depending of Z_m, we can thus compute expected values of these variables:

$$E(l(\theta', Z)|Z, \theta) = E(l_0(\theta', T)|Z, \theta) - E(l_1(\theta', Z_m)|Z, \theta) \quad [14]$$

By denoting $Q(\theta', \theta) = E(l_0(\theta', T)|Z, \theta)$ and $R(\theta', \theta) = E(l_1(\theta', Z_m)|Z, \theta)$, and knowing that $E(l(\theta', Z)|Z, \theta) = l(\theta', Z)$, we have:

$$l(\theta', Z) = Q(\theta', \theta) - R(\theta', \theta) \quad [15]$$

We search a model θ' which maximizes $l(\theta', Z)$. The E.M. method is based on the following theorem, which says that maximizing Q can not decrease the likelihood.

Theorem 1. *If* $Q(\theta', \theta) > Q(\theta, \theta)$ *then* $l(\theta', Z) > l(\theta, Z)$ *(Dempster* et al.*, 1977)*

Algorithm 1 depicts the EM algorithm. The likelihood of the final model θ^c is a local maxima. (Dempster *et al.*, 1977) recommends to repeat the experience and to select the model θ^c which maximizes the likelihood.

Algorithm 1 EM

Require: Z

1) Choose an initial model $\hat{\theta}^0$.

2) Compute $Q(\hat{\theta}^i, \hat{\theta}^i)$ for the current i (expectation phase).

3) Find $\hat{\theta}^{i+1}$ such that $Q(\hat{\theta}^{i+1}, \hat{\theta}^i) > Q(\hat{\theta}^i, \hat{\theta}^i)$ (maximization phase).

4) Iterate to step 2 until convergence.

Ensure: a model θ^c

Algorithm 2 EM + NB semi-supervised

Require: S_{lab}, S_{unl}

1) $Let\hat{\theta}^0$ be the model learned on labeled data

2) $\forall x' \in S_{unl}$ compute $P(y' = j|x', \hat{\theta}^n)$

3) Compute $\hat{\theta}^{n+1}$ using formulas [16]

4) Goto step 2 until convergence

Ensure: $\hat{\theta}^f$

2.3.2. *EM method and the Naive Bayes classifier*

EM has been used in (McCallum *et al.*, 1999) to compute Naive Bayes models in a semi-supervised framework. Given S_{lab} and S_{unl}, and a model θ^n at step n, the parameters α and λ_{ikj} of the next model θ^{n+1} are computed as follows:

$$\alpha = \frac{n_1 + \sum_{s=1}^{l'} \hat{P}(y'_s = 1|x'_s, \theta^n)}{l + l'},$$

$$\lambda_{ikj} = \frac{n_{ij}^k + \sum_{s=1}^{l'} \hat{P}(y'_s = j|x_s^{'i} = k, \theta^n)}{\sum_r [n_{ij}^r + \sum_{s=1}^{l'} \hat{P}(y'_s = j|x_s^{'i} = r, \theta^n)]} \quad [16]$$

Where $\hat{P}(A|\theta^n)$ is the estimated probability of A within the model θ^n.
Algorithm 2 summarizes the algorithm.

Results for text classification tasks show a notable improvement of the performances when unlabeled data are added to the labeled data-set.

2.4. *Identifiability of Naive Bayes model parameters with distribution P(x)*

In (Geiger *et al.,* 2001), it was shown that under Naive Bayes hypothesis, parameters of the model are identifiable from distribution $P(x)$ on X when the number of attributes is at least equal to three up to a permutation of the classes (see paper for further details).

In order to compare our work to theirs, we shortly recall in this section the formulas established in (Geiger *et al.,* 2001) when $Y = \{0, 1\}$ and attributes are binary to compute Naive Bayes models parameters with distribution $P(x)$.

Let:

$$\begin{aligned} &z_{ij...r} = P(x^i = 1, x^j = 1, ..., x^r = 1), \\ &p_i = P(x^i = 1|y = 1), \\ &q_i = P(x^i = 1|y = 0), \\ &\alpha = P(y = 1). \end{aligned} \quad [17]$$

Therefore:

$$z_{ij...r} = \alpha p_i p_j ....p_r + (1 - \alpha) q_i q_j ...q_r \quad [18]$$

Let $s, x_1, ..., x_m, u_1, ..., u_m$ be the new coordinates after the following transformation:

$$\alpha = (s + 1)/2, p_i = x_i + (1 - s)u_i, q_i = x_i - (1 + s)u_i \quad [19]$$

A second transformation on coordinates z is recursively defined as follows:

$$\begin{aligned} &z_{ij} \leftarrow z_{ij} - z_i z_j, \\ &z_{ijr} \leftarrow z_{ijr} - z_{ij} z_r - z_{ir} z_j - z_{jr} z_i - z_i z_j z_r \\ &\text{and so forth...} \end{aligned} \quad [20]$$

Then, x, u, s can be computed as follows:

$$\begin{aligned} &x_i = z_i, \\ &u_1 = \pm\sqrt{z_{12} z_{13} z_{23} + (z_{123})^2/4}/z_{23}, \\ &s = -z_{123}/(2u_1 z_{23}), \\ &u_i = z_{1i}/(p_2(s)u_1) \text{ for } i > 1 \text{ with } p_2(s) = 1 - s^2. \end{aligned} \quad [21]$$

No application is proposed in the paper. In Section 3.4, we propose an algorithm based on these formulas to compute Naive Bayes models from unlabeled data.

3. Identifiability of Naive Bayes models for semi-supervised learning variants

Several authors (Denis *et al.*, 1999, Denis *et al.*, 2003, Liu *et al.*, 2003, Liu *et al.*, 2005) studied the asymmetrical semi-supervised learning, where the available labeled examples are all positive and drawn according to $P(x|y=1)$. Naive Bayes classifiers have also been used in this context of learning (Denis *et al.*, 2003). Note that in (Denis *et al.*, 1999, Denis *et al.*, 2003, Liu *et al.*, 2003), authors consider that the parameter $\alpha = P(y=1)$ is known, which is not true in all situations. We show below that distributions which assume Naive Bayes hypotheses are identifiable from distributions $P(x)$ and $P(x|y=1)$ without additional information.

In Section 3.1 we present general results about asymmetrical learning and show that without assumption on the distributions, asymmetrical semi-supervised learning is not a well-posed problem. Section 3.2 presents a theoretical study on the identifiability of Naive Bayes models in asymmetrical semi-supervised context. We provide a formula that shows that the model parameters are identifiable as soon as the number of attributes is at least equal to 2. This formula provides a consistent estimator for $P(y=1)$. Section 3.3 gives an adaptation of Algorithm 2 to positive and unlabeled examples. Lastly, in Section 3.4, we propose an algorithm to learn from unlabeled data using formulas given Section 2.4, provided in (Geiger *et al.*, 2001).

3.1. *General case*

Statistical learning suppose the existence of distributions $P(x)$ on X and $P(y|x)$ on Y for all $x \in X$. When $Y = \{0,1\}$, these distributions are determined by the knowledge of $P(x)$, $P(x|y=1)$ and $P(y=1)$. Indeed $P(y=1|x) = \frac{P(x|y=1)\cdot P(y=1)}{P(x)}$ and $P(y=0|x) = 1 - P(y=1|x)$. Positive and unlabeled datasets can be used to estimate $P(x)$ and $P(x|y=1)$ on X. But generally, the parameter $P(y=1)$ cannot be inferred from positive and unlabeled data.

Property 1. *Without further information, $P(y=1)$ is not determined by distributions $P(x)$ and $P(x|y=1)$.*

Proof. Let $r = \min\{\frac{P(x)}{P(x|y=1)} \mid x \in X \text{ and } P(x|y=1) \neq 0\}$. For all $\lambda \in]0, r]$, there exists P' defined on $X \times Y$ by:

- $P'(x,y) = \lambda \cdot P(x|y=1)$ if $P(x) \neq 0$ and $y = 1$
- $P'(x,y) = P(x) - \lambda \cdot P(x|y=1)$ if $P(x) \neq 0$ and $y = 0$
- $P'(x,y) = 0$ otherwise.

With this definition, P' is a generative distribution:

– $P'(y=1) = \lambda$
– $P'(x|y=1) = \frac{P'(x,y=1)}{\lambda} = P(x|y=1)$
– $P'(x|y=0) = \frac{P(x)-\lambda \cdot P'(x|y=1)}{1-\lambda}$
– $P'(x) = \lambda \cdot P'(x|y=1) + (1-\lambda) \cdot P'(x|y=0) = P(x)$

Let $\lambda_1 \in]0, r]$ and $\lambda_2 \in]0, r]$ such that $\lambda_1 \neq \lambda_2$. Let P'_1 and P'_2 be the distributions associated as defined previously. Then, $P'_1(x) = P'_2(x)\ \forall x \in X$, $P'_1(x|y=1) = P'_2(x|y=1)\ \forall x \in X$, but $P'_1(y=1) \neq P'_2(y=1)$, so $P(y=1)$ is not determined. □

Remark: When it is known that the distributions satisfy complementary properties, $P(x)$ and $P(x|y=1)$ may determine $P(y=1)$. For instance, deterministic models, *i.e.* $P(y=1|x) = 1$ or $P(y=1|x) = 0$ for all x, are such distributions. In this case:

$$P(y=1) = \sum_{x \in X} P(x)P(y=1|x) = \sum_{P(x|y=1)\neq 0} P(x). \qquad [22]$$

3.2. *Identification of Naive Bayes model parameters with distributions* $P(x)$ *and* $P(x|y=1)$

In this section, we show that for distributions which follows the Naive Bayes hypothesis, $P(x)$ and $P(x|y=1)$ determine $P(y=1)$ as soon as the number of attributes is at least 2. Moreover, we define a consistent estimator for this parameter.

Theorem 2. *Let P a probability distribution on a discrete feature space X such that P follows the Naive Bayes assumption. Then $P(x)$ and $P(x|y=1)$ determine $P(y=1)$ provided there exist at least two attributes x^i and x^j such that $P(x^i|y=1) \neq P(x^i)$ and $P(x^j|y=1) \neq P(x^j)$.*

Proof. First, let us consider extremal values for $P(y=1)$:

– remark that $P(y=1) \neq 0$ since we suppose the existence of positive data.
– if $P(y=1) = 1$, then $P(x) = P(x|y=1)$ which contradicts the hypothesis.

Consider now that $0 < P(y=1) < 1$ and note that :

$$P(x^i|y=1) \neq P(x^i) \Leftrightarrow P(x^i|y=1) \neq P(x^i|y=0).$$

Let:

– $p_{ik} = P(x^i = k|y=1)$
– $q_{ik} = P(x^i = k|y=0)$.
– $\alpha_{ik} = P(x^i = k)$

– $\alpha_{jl} = P(x^j = l)$
– $\alpha_{ik,jl} = P(x^i = k \cap x^j = l)$

For each attribute pair (i, j) ($i \neq j$), and for each pair of attributes values k, l ($k \in X^i, l \in X^j$), the following system holds:

$$\begin{cases} \alpha_{ik} = p_{ik}.P(y=1) \ + \ q_{ik}.(1 - P(y=1)) \\ \alpha_{jl} = p_{jl}.P(y=1) \ + \ q_{jl}.(1 - P(y=1)) \\ \alpha_{ik,jl} = p_{ik}.p_{jl}.P(y=1) \ + \ q_{ik}.q_{jl}.(1 - P(y=1)) \end{cases}$$

Let (i, j) and (k, l) be a pair of attributes and a pair of values such as $p_{ik} \neq q_{ik}$ and $p_{jl} \neq q_{jl}$, from the two first equations, we can write:

$$q_{ik} = \frac{\alpha_{ik} - p_{ik}.P(y=1)}{1 - P(y=1)} \quad \text{and} \quad q_{jl} = \frac{\alpha_{jl} - p_{jl}.P(y=1)}{1 - P(y=1)} \qquad [23]$$

By replacing q_{ik} and q_{jl} in the third equation, we obtain, after simplification:

$$P(y=1)(p_{ik}p_{jl} - \alpha_{ik}p_{jl} - \alpha_{jl}p_{ik} + \alpha_{ik,jl}) = \alpha_{ik,jl} - \alpha_{ik}\alpha_{jl} \qquad [24]$$

In order to obtain an analytical expression for $P(y = 1)$, it is necessary to show that $p_{ik}p_{jl} - \alpha_{ik}p_{jl} - \alpha_{jl}p_{ik} + \alpha_{ik,jl}$ is different from 0. By replacing α_{ik}, α_{jl}, $\alpha_{ik,jl}$ with their definitions, we obtain:

$$\begin{aligned} & p_{ik}p_{jl} - \alpha_{ik}p_{jl} - \alpha_{jl}p_{ik} + \alpha_{ik,jl} \\ & = (1 - P(y=1)).(p_{ik} - q_{ik}).(p_{jl} - q_{jl}) \end{aligned} \qquad [25]$$

which is not null under theorem assumptions. Therefore:

$$P(y=1) = \frac{\alpha_{ik,jl} - \alpha_{ik}\alpha_{jl}}{p_{ik}p_{jl} - \alpha_{ik}p_{jl} - \alpha_{jl}p_{ik} + \alpha_{ik,jl}} \qquad [26]$$

$P(y = 1)$ is thus determined by $P(x)$ and $P(x|y = 1)$. □

This formula provides a natural estimator for $P(y = 1)$. Let $\hat{\alpha}_{ik,jl}$, $\hat{\alpha}_{ik}, \hat{\alpha}_{jl}, \hat{p}_{ik}, \hat{p}_{jl}$ be estimates of $\alpha_{ik,jl}, \alpha_{ik}, \alpha_{jl}, p_{ik}, p_{jl}$ respectively, we consider:

$$\hat{P}(y=1) = \frac{\sum_{i,j,k,l} \hat{\alpha}_{ik,jl} - \hat{\alpha}_{ik}\hat{\alpha}_{jl}}{\sum_{i,j,k,l} \hat{p}_{ik}\hat{p}_{jl} - \hat{\alpha}_{ik}\hat{p}_{jl} - \hat{\alpha}_{jl}\hat{p}_{ik} + \hat{\alpha}_{ik,jl}} \qquad [27]$$

Algorithm 3 NB asymmetrical semi-supervised

Require: S_{pos}, S_{unl}

1) Compute estimators $\hat{\alpha}_{ik,jl}, \hat{\alpha}_{ik}, \hat{\alpha}_{jl}, \hat{p}_{ik}, \hat{p}_{jl}$ of $\alpha_{ik,jl}, \alpha_{ik}, \alpha_{jl}, p_{ik}, p_{jl}$ from S_{pos} and S_{unl}, $1 \leq i, j \leq m$, $k \in Dom(x^i)$, $l \in Dom(x^j)$

2) Compute $\hat{P}(y = 1)$ using [27]

3) Compute $\hat{q}_{ik}$, $\hat{q}_{jl}$ using [23]

Ensure: a model $\hat{\theta}$

Other estimators could be provided by formula [26].

In practice, the average of the $\hat{P}(y = 1)$ estimated with all the pairs of attributes (x^i, x^j) such that $\hat{p}_{ik} \neq \hat{q}_{ik}$ and $\hat{p}_{jl} \neq \hat{q}_{jl}$ may not provide accurate estimates of $P(y = 1)$ since $\hat{p}_{ik}\hat{p}_{jl} - \hat{\alpha}_{ik}\hat{p}_{jl} - \hat{\alpha}_{jl}\hat{p}_{ik} + \hat{\alpha}_{ik,jl}$ can be very close to zero when the sizes of the datasets are small. We do not have studied the question whether it is possible to have better estimators but this is an important question that we plan to address in future work.

We deduce from this study Algorithm 3.

In (Geiger *et al.*, 2001), it was shown that under Naive Bayes hypothesis, parameters of the model are identifiable from distribution $P(x)$ on X when the number of attributes is at least equal to three up to a permutation of the classes (see Section 3.4). Results obtained on artificial data (Section 4) show that estimator [27] converges really faster than the one computed from unlabeled data only (*c.f.* Section 3.4).

3.3. *Estimate Naive Bayes models parameters with the criteria of maximum likelihood*

Previous section presents an algorithm to compute the parameters of Naive Bayes models analytically. We were also interested to determine this parameters with the criteria of maximum likelihood. The likelihood $L(\theta', S_{pos}, S_{unl})$ of S_{pos} and S_{unl} for the model θ' can be written as follows:

$$L(\theta', S_{pos}, S_{unl}) = \beta^l L(\theta, S_{pos})(1 - \beta)^{l'} L(\theta, S_{unl}) \qquad [28]$$

With the same notation as in Section 2.2, and with β representing now the probability to draw a positive labeled example, we obtain:

Algorithm 4 EM+NB asym

Require: $S = \{S_{pos}, S_{unl}\}$

1) Estimate $\hat{P}(x^i = k|y = 1)$ and $\hat{P}(x^i = k)$ with S_{pos} et S_{unl}.

2) Compute $\hat{P}(y = 1)$ using formula 27

3) Compute an initial model θ^0 from the estimates $\hat{P}(x^i = k|y = 1)$, $\hat{P}(x^i = k)$ and $\hat{P}(y = 1)$.

4) $\forall x' \in S_{unl}$, compute $P(y = j|x', \theta^k)$, $j \in \{0, 1\}$

5) Compute a new model θ^{k+1} (see Section 2.3.2 for details)

6) Iterate to step 4 until stabilization

Ensure: $\hat{\theta}^S_{ML}$

$$L(\theta', S_{pos}, S_{unl})$$
$$= \beta^l \alpha^l \prod_{r=1}^{l} \Big(\prod_{\substack{1 \le i \le m \\ k/x^i_r = k}} \lambda_{ik1} \Big) (1-\beta)^{l'} \prod_{r=1}^{l'} \Big(\alpha \prod_{\substack{1 \le i \le m \\ k/x^i_r = k}} \lambda_{ik1} + (1-\alpha) \prod_{\substack{1 \le i \le m \\ k/x^i_r = k}} \lambda_{ik0} \Big)$$

[29]

As in the classical semi-supervised case, the values of the parameters that maximize the likelihood cannot be computed analytically, so methods such as EM must be used. We present now an iterative algorithm (Algorithm 4), adapted from Algorithm 2 (McCallum *et al.*, 1999), which estimates parameter $P(y = 1)$ by maximizing the likelihood using E.M. and using estimator [27].

Experiments on artificial data show that if the size of S_{pos} is small, computing the model parameters by using the criterion of maximum likelihood improved classification performances and accuracy of the estimates.

3.4. *Algorithm to compute Naive Bayes models from unlabeled data*

We use formulas provided by (Geiger *et al.*, 2001) (*cf* Section 2.4) to compute the parameters of Naive Bayes models from unlabeled data. Note that the $z_{ij\ldots r}$ (*cf* Section 2.4) can be estimated from unlabeled data. Note also that two models can be computed according to the sign of u_1. We deduce from these formulas Algorithm 5.

Experimental results on artificial data (Section 4) show that huge samples are necessary to provide accurate estimates of the target Naive Bayes models. When positive examples are available, they can be used to identify classes and to provide better estimates.

Algorithm 5 NB unl

Require: z

1) Estimate $u_k^+ = \frac{\sum_{\substack{1\leq i,j\leq m \\ i\neq j\neq k}} \sqrt{z_{ki}z_{kj}z_{ij}+(z_{kij})^2/4}}{\sum_{1\leq i,j\leq m, i\neq j\neq k} z_{ij}}$ $\forall k \in \{1,..,m\}$, $u_k^- = -u_k^+$

2) Estimate $s^+ = -\frac{\sum_{1\leq i,j,k\leq m, i\neq j\neq k} z_{ijk}}{\sum_{1\leq i,j,k\leq m, i\neq j\neq k} 2u_i z_{jk}}$ and $s^- = -s^+$

3) Compute model θ^+ from u_1^+ and u_i^+ or u_i^- ($i > 1$) according to the sign of z_{1i} i.e. such that $sign(u_i) = sign(z_{1i}/(p_2(s)u_1^+))$

4) Compute model θ^- from u_1^- and u_i^+ or u_i^- ($i > 1$) according to the sign of z_{1i} i.e. such that $sign(u_i) = sign(z_{1i}/(p_2(s)u_1^-))$

Ensure: two models θ^+ and θ^-.

4. Experimental results on artificial data

In order to compare the algorithms presented in this paper, we lead an experimental study on artificial data. Section 4.1 presents the experimental protocol. Next sections present the results that have been obtained. They make it possible to compare the accuracy of the estimates (Section 4.2) and the classification performances when the size of S_{pos} (S_{lab} for algorithm 1) and S_{unl} grows (Section 4.3).

4.1. *Experimental protocol*

Ten target models $\theta_c = \{P(y = 1), P(x|y = 1), P(x|y = 0)\}$ are randomly drawn. Distributions $P(x|y = 1)$ and $P(x|y = 0)$ being product distributions (which satisfy the Naive Bayes assumption) over $\{0, 1\}^n$ ($n \in \{20, 50\}$) drawn from a discrete uniform distribution. The learning datasets are generated with models θ_c. For each $n \in \{100, 300, 500\}$ and for each model θ_c, 20 independent datasets S_{lab} (resp. S_{unl}) of n labeled (resp. $10n$ unlabeled) examples are drawn. The results (Table 1 and 2) are averages computed on 200 datasets (20 datasets for each model). Test sets S_{test} contain 10000 examples generated from θ_c. Positive examples sets S_{pos} are extracted from S_{lab}, so $|S_{pos}| \approxeq P(y = 1) * |S_{lab}|$.

4.2. *Parameters estimate accuracy*

We first compare the accuracy of $P(y = 1)$ estimates provided by algorithms 2, 3, 4 and 5. Table 1 shows the mean square error of $P(y = 1)$ estimates obtained by the four algorithms.

We point out that algorithm 2 learns from S_{lab} and S_{unl}, algorithms 3 and 4 from S_{pos} and S_{unl} and algorithm 5 with S_{unl}. The first line indicates the size of the samples, the second one the number of binary attributes and others the square root of the mean square error of $P(y = 1)$ estimates for the four algorithms.

$\|S_{lab}\|, \|S_{unl}\|$	$100, 1000$		$300, 3000$		$500, 5000$	
Number of attributes	20	50	20	50	20	50
EM+NB semi-sup. (Algorithm 2)	**0.014** (**0.003**)	**0.010** (**0.002**)	**0.008** (**0.002**)	**0.006** (**0.001**)	**0.006** (**0.002**)	**0.005** (**0.000**)
NB asy. semi-sup. (Algorithm 3)	0.047 (0.013)	0.022 (0.009)	0.023 (0.006)	0.012 (0.004)	0.012 (0.004)	0.007 (0.003)
EM+NB asym (Algorithm 4)	**0.014** (**0.003**)	**0.010** (**0.002**)	**0.008** (**0.002**)	**0.006** (**0.001**)	**0.006** (**0.002**)	**0.005** (**0.000**)
NB unl (Algorithm 5)	0.127 (0.081)	0.080 (0.053)	0.088 (0.081)	0.048 (0.038)	0.069 (0.069)	0.032 (0.025)

Table 1. *Square root of the mean square error of $P(y = 1)$ estimates obtained by the four algorithms. Best results are in boldface. Standard deviations are indicated between brackets*

Algorithms 2, 3 and 4 provides the better results according to stability of the estimators. We have carried out other experiments where EM is run on randomly drawn initial models: many runs are necessary to obtain a accurate estimate of $P(y = 1)$ while using the model inferred by Algorithm 3 as the initial model makes it possible to run EM only once.

Worse results are obtained by algorithm 5 (NB unl). This estimator converges much more slowly than others. It requires too much examples to provide good estimates in practice.

We can also observe that the two algorithms which use E.M. method tends to have the same results.

4.3. *Classification performances*

We now present the results obtained for classification tasks. The experimental protocol is described in Section 4.1. We consider the criterion of prediction error rate ($\hat{P}(f(x) \neq y)$). Results are reported in Table 2. First line indicates the number of binary attributes. The second one gives the averaged prediction error rate of the target models θ_c on S_{test} and the standard deviation of it for the 200 experiences.

We can note that results obtained by algorithms learning from positive and unlabeled data tend to be similar to results obtained with labeled and unlabeled data. It

shows that the loss of negative examples do not decrease performances. This is very interesting to know that it is not penalizing not to have labelled examples for one on the two classes. For some classifications problems, data of one class are most difficult to obtain than data of the other class.

Note that standard deviations decreases when size of learning datasets increases. They are very large for small sizes of the data-sets, but smaller when these sizes grow. This result is not really surprising after having observe the parameters estimate accuracy in Section 4.2.

The less accurate algorithm is the algorithm 5, which learns with only unlabeled examples. We can explain this result by the high mean square error obtained by this estimator.

Nb of binary attributes x^i			20	50
θ_c performances			0.049 (0.022)	0.001 (0.002)
Sets size	Algorithm	Datasets	Performances	
$\|S_{lab}\| = 100$ $\|S_{unl}\| = 1000$	Algorithm 2 EM+NB semi-sup	S_{lab}, S_{unl}	**0.051** (**0.022**)	**0.002** (**0.002**)
	Algorithm 3 NB asy. semi-sup	S_{pos}, S_{unl}	0.106 (0.036)	0.015 (0.021)
	Algorithm 4 EM+NB asym	S_{pos}, S_{unl}	**0.051** (**0.022**)	**0.002** (**0.002**)
	Algorithm 5 NB unl	S_{unl}	0.225 (0.082)	0.113 (0.087)
$\|S_{lab}\| = 300$ $\|S_{unl}\| = 3000$	Algorithm 2 EM+NB semi-sup	S_{lab}, S_{unl}	**0.049** (**0.022**)	**0.001** (**0.002**)
	Algorithm 3 NB asy. semi-sup	S_{pos}, S_{unl}	0.068 (0.024)	0.004 (0.005)
	Algorithm 4 EM+NB asym	S_{pos}, S_{unl}	0.050 (0.022)	**0.001** (**0.002**)
	Algorithm 5 NB unl	S_{unl}	0.181 (0.086)	0.108 (0.078)
$\|S_{lab}\| = 500$ $\|S_{unl}\| = 5000$	Algorithm 2 EM+NB semi-sup	S_{lab}, S_{unl}	**0.049** (**0.022**)	**0.001** (**0.002**)
	Algorithm 3 NB asy. semi-sup	S_{pos}, S_{unl}	0.057 (0.022)	0.002 (0.003)
	Algorithm 4 EM+NB asym	S_{pos}, S_{unl}	**0.049** (**0.022**)	**0.001** (**0.002**)
	Algorithm 5 NB unl	S_{unl}	0.164 (0.091)	0.106 (0.075)

Table 2. *Performances of the algorithms on artificial data*

5. Prediction of the disulfide connectivity into proteins

This section presents experiments of the Algorithm 4 on biological data. The biological problem is to predict the *disulfide bridges* within a protein.

A protein may be represented by its primary structure – a sequence of amino acids– from which a tridimensional structure is gathered; disulfide bridges are involved in the 3D conformation of a protein, as covalent bonds between two cysteines (amino acid C). As a consequence, predicting such bridges from the primary sequences would be a first step towards the prediction of the tridimensional structure of proteins.

One part of the information necessary (but not sufficient) for predicting such bridges is located around each cysteine. In our approach for predicting disulfide bridges, we thus suppose that the amino acids located around cysteines contribute to establish an *affinity* (also called *propensity*) between those two cysteines.

Determining whether two fragments of a protein have affinity one for the other can be represented as a problem of supervised classification, where the pairs of fragments around two bonded cysteines are positive examples, and pairs of fragments around two unbonded cysteines are negative examples. Instead, we make the hypothesis that two bonded fragments are positive examples, but that two unbonded fragments could actually be bonded in another context: the local information around unbonded cysteines do not give enough information about the concept of affinity. We are thus in a case of asymmetric semi-supervised learning: a pair of cysteines that are not bonded is a non determined (unlabeled) example rather than a negative example. Indeed, a cysteine cannot be involved in more than one bridge whereas it may have affinity for more than one other cysteine.

5.1. *Data*

The data are extracted from the *Protein Data Bank (PDB)* by *Christophe Geourjon (IBCP, Lyon, France)* for the working group ACI GENOTOD (*http://www.loria.fr/ guermeur/GdT/*): 227 proteins are available, annotated with regards to known disulfide bridges within each protein. Note that only oxidized cysteines can be involved in a bridge, and either all cysteines of a protein are bonded, or none of them form a bridge. Figure 1 describes the data according to the number of amino acids in proteins, and Figure 2 describes the proteins according to the number of bridges.

5.2. *Experimental protocol*

Data representation

We try to estimate local affinities into proteins. In the case of disulfide bridges, these interactions arise between amino acids located close to cysteines. We thus extract from the protein sequence a set of fragments centered on cysteines. Let us denote by

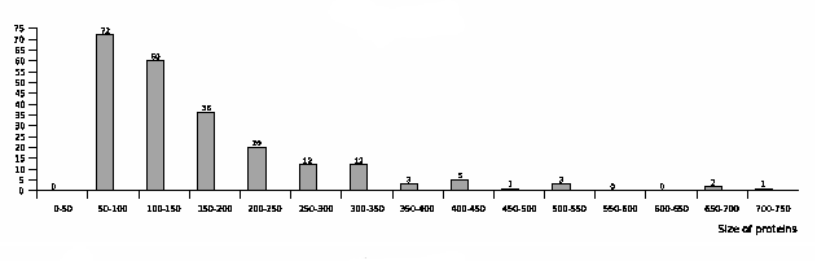

Figure 1. *Histogram of proteins sizes*

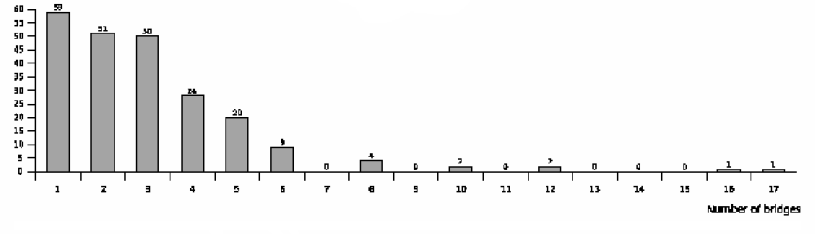

Figure 2. *Distribution of the proteins grouped according to the number of disulfide bridges*

windows such fragments, and let us denote $x_{-n}, ..., x_{-1}, x_0, x_1, ..., x_n$ a window of radius n (x_0 is thus a cysteine). We work with an alphabet of size 231 (number of ordered couples on an alphabet of size 21: the 20 amino acids and a letter representing unknown amino acids or missing amino acids when the cysteine is too much close to one end of sequence). We set up three codings:

– simple coding: $\{(x_i, x'_i)\},\ i \in \{-n, ..., n\}, i \neq 0, x_i \in f, x'_i \in f'$

– double coding: $\{(x_i, x'_i)\} \cup \{(x_i, x'_{-i})\},\ i \in \{-n, ..., n\}, i \neq 0, x_i \in f, x'_i \in f'$

– crossed coding: $\{(x_i, x'_j)\},\ i, j \in \{-n, ..., n\}, x_i \in f, x'_i \in f'$

The first coding represents the pairs of aligned amino acids between two windows of the same size. The double coding takes into account the fact that we do not know the directions of segments and considers the both possibilities. The last coding considers all the pairs of amino acids that we can form with two segments.

Learning protocol

For a protein containing n bridges, we have $n(2n-1)$ pairs of windows potentially in interaction. If a pair is bonded, we consider it as a positive example. Non bonded pairs are considered as negative examples in the Naive Bayes approach, while they are considered as unlabeled examples in Algorithm 4.

We split the set of annotated proteins according to the number of bridges: the learning is independent from one set to another. We studied cases n =2,3,4, and 5 (for $n = 1$, the case is trivial, for $n > 5$, not enough data is available for a significative learning).

Test protocol
In order to test the quality of affinity estimation, we set up a test protocol which accounts for the following information: for each pair of windows in a test protein, we compute the probability that the two windows have a high affinity, the connectivity that maximizes likelihood is considered. It comes down to computing the maximal weight perfect coupling in a full graph. The vertices of that graph are the protein windows, and an edge between two windows is the probability that these windows have high affinity between them.

We use 10-fold cross-validations for each of the three previous encodings. We compare results with random selection of the bridges. For a protein containing n bridges, the expected number of correctly predicted bridges with a random selection is $\frac{n}{2n-1}$.

5.3. *Experimental results*

The best results were obtained with the crossed encoding: only these results are reported here. They are actually an average of 100 experiments.

Nb of bridges/cysteines per protein	2/4	3/6	4/8	5/10
Nb and % of correctly predict bridges with a random selection	34 33,33%	30 20%	16 14,3%	11,1 11,1%
Nb and % of correctly predict bridges with Algorithm NB (supervised)	41 40,2%	26,25 17,5%	14,22 12,7%	5,8 5,8%
Nb and % of correctly predict bridges with Algorithme 4 (asym. semi-sup.)	60 **58,8%**	50,1 **33,4%**	18,26 **16,3%**	13,2 **13,2%**

Table 3. *Experimental results on biological data: the contribution of the algorithm 4 for predicting disulfide bridges is obvious*

Other results are available in (Fariselli *et al.*, 2001, Fariselli *et al.*, 2002, Vullo *et al.*, 2004). The best results (Fariselli *et al.*, 2002) are mostly better than ours (table below), but they were obtained by much more sophisticated methods (recursive neural networks), more data, and their methods integrated another major information which is the information about evolution (they encoded fragments according to profiles). The differences between their context of experiments and ours, make difficult any accurate comparison.

Nb of bridges per protein	2 br.	3 br.	4 br.	5 br.
Nb of proteins	156	146	99	45
Correctly predicted bridges	**73%**	**56%**	**37%**	**30%**

Table 4. *Results obtained in (Fariselli* et al.*, 2002)*

Our purpose was to determine whether an unbounded pair of cysteines should be considered as negative example or as an unlabeled example. The results obtained are sufficient to conclude that our biological hypothesis seems to be confirmed: it is relevant to consider non bonded pairs of cysteines as unlabeled examples rather than negative examples. This hypotheses must now be integrated in more sophisticated methods such as RNN, SVMs, etc.

6. Conclusion

In this paper, we lead a study in the asymmetrical semi-supervised context, where only positive and unlabeled examples are available. We show that the asymmetrical semi-supervised learning is a well-posed problem when attributes follow Naive Bayes assumption. This result can be deduce from (Geiger *et al.,* 2001). In this paper authors show that Naive Bayes models are identifiable from unlabeled examples only. This result is stronger than our but we show that taking into account information provided by positive examples increases significantly accuracy of estimates. We give analytical methods to identify models at the limit which outperforms those gived in (Geiger *et al.,* 2001). We also propose iterative algorithm to compute models on the criteria of maximum likelihood, inspired of the algorithm proposed in (McCallum *et al.,* 1999) for classical semi-supervised learning. Both methods provide similar results, which signify that the loss of negative examples do not penalize the learning.

The application of this work for the prediction of disulfide connectivity supports an original assumption for data representation. It seems to be better to consider unbounded pairs of cysteines as unlabeled examples, which provide no information concerning the class, rather than negative examples. We are currently working to apply this method to other biological data (in particular Beta sheets) and to determine a protocol to decide whether there are local affinity in molecules. We also look for methods such as SVM which are notably more effective than Naive Bayes classifier and which could be developed in this framework.

Acknowledgements

This study is partially supported by the A.C.I. "Masses de données" GENOTO3D.

7. References

Dempster A., N.M.Laird, D.B.Rubin, « Maximum likelihood from incomplete data via the em algorithm », *Journal of the Royal Statistical Society*, p. 39:1-38, 1977.

Denis F., DeComite F., Gilleron R., Letouzey F., « Positive and Unlabeled Examples help learning », *The 10th International Workshop on ALT*, 1999.

Denis F., Gilleron R., Laurent A., Tommasi M., « Text Classification and co-training from Positive and Unlabeled Examples », *Proceedings of the ICML 2003 Workshop : The Continuum from Labeled to Unlabeled Data*, p. 80-87, 2003.

Domingos P., Pazzani M., « Simple bayesian classifiers do not assume independance », *Proceedings of the Thirteenth NCAI and the Eighth IAAIC*, 1996.

Fariselli P., Casadio R., « Prediction of disulfide connectivity in proteins », *Bioinformatics*, number 17(10), p. 957-964, 2001.

Fariselli P., Martelli P., Casadio R., « A neural network-based method for predicting the disulfide connectivity in proteins », *Proceedings of KES 2002, Knowledga based intelligent information engineering systems and allied technologies*, 2002.

Geiger D., Heckerman D., King H., Meek C., « Stratified Exponential Families : Graphical Models and Model Selection », *The Annals of Statistics (29)*, 2001.

Hastie T., Tibshirani R., Friedman J., *The elements of statistical learning*, 2001, Springer.

Liu B., Li X., « Learning to classify text using positive and unlabeled data », *Proceedings of Eighteenth IJCAI*, 2003.

Liu B., Li X., « Learning from Positive and Unlabeled Examples with Different Data Distributions », *Proceedings of the 16th European Conference on Machine Learning*, 2005.

McCallum A., Thrun S., Mitchell T., « Text classification from Labeled and Unlabeled Documents using E.M. », *Machine Learning*, number 39 (2-3), p. 103-134, 2000.

Vullo A., Frasconi P., « Disulfide connectivity prediction using recursive neural networks and evolutionary information », *Bioinformatics*, number 20, 2004.