On semi-supervised learning

Alejandro Cholaquidis[a], Ricardo Fraiman[a] and Mariela Sued [b]
[a] CABIDA and Centro de Matemática,
Facultad de Ciencias, Universidad de la República, Uruguay
[b] Instituto de Cálculo,
Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires

Abstract

Semi-supervised learning deals with the problem of how, if possible, to take advantage of a huge amount of unclassified data, to perform a classification in situations when, typically, there is little labeled data. Even though this is not always possible (it depends on how useful, for inferring the labels, it would be to know the distribution of the unlabeled data), several algorithm have been proposed recently.

A new algorithm is proposed, that under almost necessary conditions, attains asymptotically the performance of the best theoretical rule as the amount of unlabeled data tends to infinity. The set of necessary assumptions, although reasonable, show that semi-supervised classification only works for very well conditioned problems. The focus is on understanding when and why semi-supervised learning works when the size of the initial training sample remains fixed and the asymptotic is on the size of the unlabeled data. The performance of the algorithm is assessed in the well known "Isolet" real-data of phonemes, where a strong dependence on the choice of the initial training sample is shown.

1 Introduction

Semi-supervised learning (SSL) dates back to the 60's, starting with the pioneering works of Scudder (1965), Fralick (1967) and Agrawala (1970), among others. Recently it has gained paramount importance due to the huge amount of data coming from diverse sources, such as the internet, genomic research, text classification, and many others; see, for instance Zhu (2008) or Chapelle, Schölkopf and Zien, eds. (2006) for a survey on SSL. This large amount of data is typically unlabeled; the main purpose of SSL is to jointly classify these data in the presence of a small "training sample". Namely, a lot of unlabeled data together with a small quantity of labeled data must be combined to classify each unlabeled observation. Several methods have been proposed to achieve this goal: self-training, co-training, transductive support vector machines, and graph-methods are some of them. However, a

natural question still remains unsolved, as mentioned in Chapelle, Schölkopf and Zien, eds. (2006): "in comparison with a supervised algorithm that uses only labeled data, can one hope to have a more accurate prediction by taking into account the unlabeled points? [...] In principle, the answer is yes". Nevertheless, having a large set of data to classify is like knowing $p(x)$, the distribution of the features vector; thus, the gain in prediction accuracy depends on how useful it is to know $p(x)$ in the inference of $p(y|x)$. Typically, for the density $p(x)$ to be useful, it needs to have deep valleys between classes. In other words, clustering techniques have to perform well in the presence of only unlabeled data. Smoothness of the labels with respect to the features, or low density at the decision boundary, are examples of the kind of hypotheses required to get satisfactory results in the cluster analysis literature.

Another important issue in SSL is the amount of labeled data necessary to be able to classify the unlabeled data. In the framework of generative models, when $p(x)$ is assumed to be an identifiable mixture of parametric distributions, Zhu (2008) argued that "ideally we only need one labeled example per component to fully determine the mixture distribution". Indeed, under the regularity conditions presented in Section 5, one labeled example per component will also be enough to prove the consistency of the algorithm that we propose in this work.

Although there is a large body of literature on SSL, as pointed out by Azizyan et al. (2013), "making precise how and when these assumptions actually improve inferences is surprisingly elusive, and most papers do not address this issue; some exceptions are Rigollet (2007), Singh et al. (2008), Lafferty and Wasserman (2007), Nadler et al. (2009), Ben-David et al. (2008), Sinha and Belkin (2009), Belkin and Niyogi (2004) and Niyogi (2008)". In Haffari and Sarkar (2007) the Yarowski algorithm is analyzed, while in Azizyan et al. (2013) an interesting method called "adaptive semi-supervised inference" is introduced, and a minimax framework for the problem is provided.

Our proposal takes a different direction: it is focused on the case of a small training sample size n (i.e. the labeled data), but the amount l of the unlabeled data goes to infinity (see Figure 1). We provide a simple algorithm to classify the unlabeled data, which has a resemblance to the Yarowski algorithm . We prove that, under some natural and necessary conditions, our method performs as good as the theoretical (unknown) best rule, with probability one, asymptotically in l. The algorithm is of the "self-training" type; this means that at every step a point from the unlabeled set is labeled using the training sample built up to that step, and incorporated into the

training sample. In this way the training sample increases from one step to the next. A simplified, computationally more efficient alternative algorithm is also provided in Section 6. This paper is organized as follows: Section 2 introduces the basic notation and the set-up necessary to read the rest of the paper. Section 3 proves that the theoretical (unknown) best rule to classify the unlabeled sample is to use the Bayes rule. In Section 4 we introduce the algorithm and prove that all the unlabeled data are classified. Section 5 proves that, as the number of unlabeled data grows to infinity, the algorithm performs as good as Bayes rule. In Section 6 we introduce a simplified and faster algorithm. Section 7 analyses two examples using simulated data, and a third one based on a real data set. Lastly, Section 8 discusses the hypotheses. The proofs are included in Appendixes A and B.

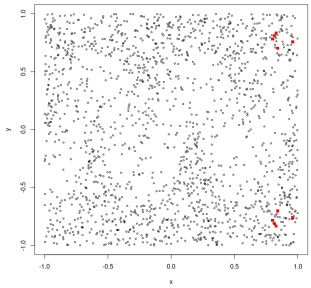

Figure 1: In black: the X_j without labels, in red a small training sample (5 data from each subpopulation).

2 Notation and set-up

We consider $\mathbb{R}^d$ endowed with the Euclidean norm $\|\cdot\|$. The open ball of radius $r \geq 0$ centered at x is denoted by $B(x,r)$. With a slight abuse of notation, if $A \subset \mathbb{R}^d$, then we write $B(A,r) = \cup_{s\in A} B(s,r)$. The d-dimensional Lebesgue measure is denoted by μ_L, while $\omega_d = \mu_L(B(0,1))$. For $\delta > 0$ and $A \subset \mathbb{R}^d$, the δ-interior of A is defined as $A \ominus B(0,\delta) = \{x : B(x,\delta) \subset A\}$. The distance from a point x to a set A is denoted by $d(x,A)$, i.e. $d(x,A) = \inf\{\|x-a\| : a \in A\}$. If $A \subset \mathbb{R}^d$, then ∂A denotes its boundary, $int(A)$ its interior, A^c its complement, and $\overline{A}$ its closure. Let $\mathcal{D}^n = (\mathcal{X}^n, \mathcal{Y}^n) = \{(X^1,Y^1),\ldots,(X^n,Y^n)\}$ be a given realization of a sample with the same distribution as $(X,Y) \in S \times \{0,1\}$, where $S \subset \mathbb{R}^d$. We

assume that they are identically distributed but not necessarily independent. Let $\eta(x)$ denote the conditional mean of Y given $X = x$; namely, $\eta(x) = \mathbb{E}(Y|X = x)$. Consider $\mathcal{D}_l = (\mathcal{X}_l, \mathcal{Y}_l) = \{(X_1, Y_1), \ldots, (X_l, Y_l)\}$ an iid sample with the same distribution as (X, Y), where $n \ll l$. The sample $\mathcal{X}_l = (X_1, \ldots, X_l)$ is known while the labels $\mathcal{Y}_l = (Y_1, \ldots, Y_l)$ are unobserved.

3 Theoretical best rule

It is well known that the optimal rule for classifying a single new datum X is given by the Bayes rule, $g^*(X) = \mathbb{I}_{\{\eta(X) \geq 1/2\}}$. In the present paper, we move from the classification problem of a single datum X to a framework where each coordinate of $\mathcal{X}_l = (X_1, \ldots, X_l)$ must be classified. The label associated with each coordinate X_i may be constructed on the basis of the entire vector and, therefore, a classification rule $\mathbf{g}_l = (g_1, \ldots, g_l)$ comprises l functions $g_i : S^l \to \{0, 1\}$, where $g_i(\mathcal{X}_l)$ indicates the label assigned to X_i based on the entire set of observations $\mathcal{X}_l$. The performance of a rule $\mathbf{g}_l = (g_1, \ldots, g_l)$ is given by its mean classification error, namely $L(\mathbf{g}_l) := \mathbb{E}\Big(\frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_i(\mathcal{X}_l) \neq Y_i}\Big)$. Observe that the random variable $\#\{i : g_i(\mathcal{X}_l) \neq Y_i, (X_i, Y_i) \in \mathcal{D}_l\}$ is not necessarily $Binomial(l, p)$ for some $p \geq 0$.

The next result establishes that the optimal classification rule classifies each element ignoring the presence of the rest of the observations, by means of invoking the Bayes rule for each individual observation.

Proposition 1. *The performance of a rule* $\mathbf{g}_l$ *is bounded from below by* $L^* = \mathbb{P}(g^*(X) \neq Y)$*, and the lower bound is attained with the rule* $\mathbf{g}_l^* = (g_1^*, \ldots, g_l^*)$*, where* $g_i^*(\mathcal{X}_l) = g^*(X_i)$ *for all* $i = 1, \ldots, l$*.*

In practice, since the distribution of (X, Y) is unknown, we try to find a sequence $\mathbf{g}_{n,l} = (g_{n,l,1}, \ldots, g_{n,l,l})$ (where the third index indicates the step of the algorithm) depending on $\mathcal{D}^n$ and $\mathcal{X}_l$, such that

$$\lim_{l \to \infty} \mathbb{E}_{\mathcal{D}_l}\Big(\frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l) \neq Y_i}\Big) - L(\mathbf{g}_l^*) = 0, \text{ for a fixed realization } \mathcal{D}_n, \quad (1)$$

where $r(i)$ is the step of the algorithm at which the point X_i is classified and $\mathbb{E}_{\mathcal{D}_l}$ denotes the expectation wrt $\mathcal{D}_l$. In the next section we present an algorithm that, under almost necessary conditions (discussed in Section 8), satisfies a stronger property. Specifically, we will show that

$$\lim_{l \to \infty} \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l) \neq Y_i} = \mathbb{P}\{g^*(X) \neq Y\} \quad a.s.$$

4 Algorithm

We provide an algorithm which is asymptotically optimal in the sense of satisfying condition (1). For this purpose, we update the training sample sequentially incorporating into the initial set $\mathcal{D}^n$ an observation X_{j_i} in $\mathcal{X}_l$ with a predicted label $\tilde{Y}_{j_i} \in \{0,1\}$. At each step we choose the point whose score to predict its label is as extreme as possible, as stated in display (3). Scores are constructed according to the majority rule in a neighborhood of the corresponding observations to be classified; i.e., we estimate $\eta(x)$ with a Nadaraya-Watson estimator using a uniform kernel, based on both $\mathcal{D}^n$ and those points already classified by the algorithm up to the present step. In this way we choose the "*best classifiable point*" from those that remain unclassified, as indicated in the following recipe:

Initialization: Let $\mathcal{Z}_0 = \mathcal{X}^n$, $\mathcal{U}_0 = \mathcal{X}_l$, $\mathcal{T}_0 = \mathcal{D}^n$.

STEP j: For j in $\{1,\dots,l\}$, choose the *best classifiable point* in $\mathcal{U}_{j-1}$, from those that are at a distance smaller than h_l from the points already classified, as follows: let $\mathcal{U}_{j-1}(h_l) = \{X \in \mathcal{U}_{j-1} : d(\mathcal{Z}_{j-1}, X) < h_l\}$; for $X_i \in \mathcal{U}_{j-1}(h_l)$, consider

$$\hat{\eta}_{j-1}(X_i) = \frac{\sum_{r:(X^r,Y^r)\in\mathcal{D}^n} Y_r \mathbb{I}_{B(X_i,h_l)}(X^r) + \sum_{r:(X_r,\tilde{Y}_r)\in\mathcal{T}_{j-1}\setminus\mathcal{D}^n} \tilde{Y}_r \mathbb{I}_{B(X_i,h_l)}(X_r)}{\sum_{r:(X^r,Y^r)\in\mathcal{D}^n} \mathbb{I}_{B(X_i,h_l)}(X^r) + \sum_{r:(X_r,\tilde{Y}_r)\in\mathcal{T}_{j-1}\setminus\mathcal{D}^n} \mathbb{I}_{B(X_i,h_l)}(X_r)}, \tag{2}$$

and define $$X_{i_j} = \underset{i:X_i\in\mathcal{U}_{j-1}(h_l)}{\arg\max} \max\Big\{\hat{\eta}_{j-1}(X_i), 1-\hat{\eta}_{j-1}(X_i)\Big\}. \tag{3}$$

If there is more than one i_j satisfying (3), choose one that maximizes

$$\#\{\mathcal{X}_l \cap B(X_{i_j}, h_l)\}. \tag{4}$$

Then label X_{i_j} with $\tilde{Y}_{i_j}$ defined by $\tilde{Y}_{i_j} = g_{n,l,j-1}(X_{i_j})$, where $g_{n,l,j-1}$ is the classification rule associated with $\hat{\eta}_{j-1}$ defined in (2). Namely, $\tilde{Y}_{i_j} = \mathbb{I}_{\{\hat{\eta}_{j-1}(X_{i_j})\geq 1/2\}}$. Consider

$$\mathcal{Z}_j = \mathcal{Z}_{j-1}\cup\{X_{i_j}\}, \quad \mathcal{U}_j = \mathcal{U}_{j-1}\setminus\{X_{i_j}\} \quad \text{and} \quad \mathcal{T}_j = \mathcal{T}_{j-1}\cup\{(X_{i_j},\tilde{Y}_{i_j})\}.$$

OUTPUT: $\{(X_{i_1},\tilde{Y}_{i_1}),\dots,(X_{i_l},\tilde{Y}_{i_l})\}$.

Alternatively, to reduce the computation time, in Step j, instead of choosing only one point satisfying (3) and maximizing (4), it is possible

to choose, among the points that satisfy (3), all those fulfilling (4). More precisely, we define $\aleph_j$ as the set of all the points that satisfy (3) and $\Gamma_j = \{X_{1_j}, \ldots, X_{m_j}\} \subset \aleph_j$ that maximize $\#\{\mathcal{X}_l \cap B(X_{r_j}, h_l)\}$. Then we label $X_{1_j}, \ldots, X_{m_j}$ with $\tilde{Y}_{1_j}, \ldots, \tilde{Y}_{m_j}$ defined by $\tilde{Y}_{r_j} = g_{n,l,j-1}(X_{r_j})$ for all $X_{r_j} \in \Gamma_j$, where $g_{n,l,j-1}$ is the classification rule associated with $\hat{\eta}_{j-1}$ defined in (2). More precisely, $\tilde{Y}_{r_j} = \mathbb{I}_{\{\hat{\eta}_{j-1}(X_{r_j}) \geq 1/2\}}$. Lastly $\mathcal{Z}_j = \mathcal{Z}_{j-1} \cup \Gamma_j$, $\mathcal{U}_j = \mathcal{U}_{j-1} \setminus \Gamma_j$ and $\mathcal{T}_j = \mathcal{T}_{j-1} \cup \{(X_{1_j}, \tilde{Y}_{1_j}), \ldots, (X_{m_j}, \tilde{Y}_{m_j})\}$. The results discussed in the remainder of this work hold for both versions of the algorithm. To simplify the notation, they are only presented for the first version, labeling one point at each step. However, the data analysis developed in Section 7 includes the second version of the algorithm.

It remains to be proved that the algorithm classifies the whole set $\mathcal{X}_l$. For that purpose, define $I_0 = \eta^{-1}\{[0, 1/2)\}$, $I_1 = \eta^{-1}\{(1/2, 1]\}$, and assume that I_0 and I_1 are connected and *coverable*, as stated in condition H3 below. Observe that $I_1 \cup I_0 \cup \eta^{-1}(1/2) = S$, where S is assumed to be the support of the random vector X. We decided to include H3 to facilitate the proof of Proposition 2. In Proposition 3 we will provide sufficient conditions which guarantee the validity of H3. Such conditions are expressed in terms of geometric restrictions on I_a, $a = 0, 1$, regularity assumptions on the density function f of the distribution of X, and on the rate at which the bandwidth h_l decreases to zero. These conditions will also be discussed in Section 8. Additionally, we require to have at least one point of training sample in I_a, for $a = 0, 1$. To be more precise, consider the following assumptions:

H1. $\mathbb{P}\{X \in \eta^{-1}(1/2)\} = 0$.

H2. For $a = 0, 1$, *i)* I_a is connected, and *ii)* $\mathbb{P}(X \in I_a) > 0$

H3. The covering property: $\mathbb{P}(\mathcal{I}_a) = 1$, for $a = 0, 1$, where,

$$\mathcal{I}_a = \bigcup_{l_0} \bigcap_{l \geq l_0} \mathcal{I}_{a,l} \quad \text{and} \quad \mathcal{I}_{a,l} = \left\{ I_a \subseteq \bigcup_{X \in \mathcal{X}_l \cap I_a} B(X, h_l/2) \right\}, l \in \mathbb{N}.$$

H4. There exists X_a^* in $\mathcal{D}^n$ such that $X_a^* \in I_a$, for $a = 0, 1$.

In the sequel, we will assume H1 and therefore $\mathbb{P}(X \in I_0 \cup I_1) = 1$. We can now establish that, for l large enough, the algorithm assigns labels to each point in $\mathcal{X}_l$.

Proposition 2. *Assume H1, H2 i), H3 and H4. Then, with probability one, for l large enough, all the points in $\mathcal{X}_l$ are classified by the algorithm: $\mathbb{P}(\mathcal{F}) = 1$, where $\mathcal{F} = \cup_{L=1}^{\infty} \cap_{l=L}^{\infty} \mathcal{F}_l$ and, for $l \in \mathbb{N}$, $\mathcal{F}_l = \{\omega : \mathcal{X}_l(\omega)$ is entirely classified$\}$.*

5 Consistency of the algorithm

To prove the consistency of the algorithm additional conditions are required. They involve regularity properties of different sets and the rate at which h_l decreases. Define the following sets, illustrated in Figure 7:

$$A_0^\delta = I_0 \ominus B(0,\delta)\ , \quad A_1^\delta = I_1 \ominus B(0,\delta)\ ,$$
$$B_0^h = I_0 \cap B(I_1,h)\ , \quad B_1^h = I_1 \cap B(I_0,h)\ .$$

Beside H1-H4 introduced in Section 4, we will also assume that both the δ-interior A_0^δ and A_1^δ of I_0 and I_1, respectively, are connected and *coverable*, as stated in H5. This hypothesis (as we will see in Appendix B) is fulfilled if we assume that the set $\overline{I_a^c}$, $a = 0, 1$, has positive reach, (as introduced in Federer (1959)) and $lh_l^d/\log(l) \to \infty$. Assumption H7 holds if $lh_l^{2d}/\log(l) \to \infty$, as it is proved in Abdous and Theodorescu (1989). Moreover, the density f of the distribution of X needs to take larger values on the interiors $A_0^\delta \cup A_1^\delta$ than on the borders $B_0^h \cup B_1^h$, as indicated in H6. Finally, all the labels in the training set $\mathcal{D}^n$ must agree with those determined by the Bayes's rule, beside being well located, as presented in H8. Namely, consider the following set of hypotheses, which will be discussed in Section 8:

H5. There exists $\delta_0 > 0$ such that, for $a = 0, 1$ and for any $\delta < \delta_0$, *i)* A_a^δ is connected, and *ii)* $\mathbb{P}(\mathcal{A}_a^\delta) = 1$, where

$$\mathcal{A}_a^\delta = \bigcup_{l_0} \bigcap_{l \geq l_0} \mathcal{A}_{a,l}^\delta \quad \text{and} \quad \mathcal{A}_{a,l}^\delta = \left\{ A_a^\delta \subseteq \bigcup_{X \in \mathcal{X}_l \cap A_a^\delta} B(X, h_l/2) \right\}, l \in \mathbb{N}. \quad (5)$$

H6. The Valley Condition: The probability function P_X induced by X has a density f verifying that, there exists $\delta_1 > 0$ such that for all $\delta < \delta_1$ there exists $\gamma = \gamma(\delta) > 0$, such that when $h < \delta$.

$$f(a) - f(b) > \gamma\ , \text{ for all } a \in A_0^\delta \cup A_1^\delta \text{ and all } b \in B_1^h \cup B_0^h, \quad (6)$$

H7. The kernel density estimator $\hat{f}_l(u) = (\omega_d l h^d)^{-1} \sum_{i=1}^{l} \mathbb{I}_{B(u,h_l)}(X_i)$ converges to $f(u)$ uniformly over its support S, almost surely:

$$\mathbb{P}\left(\bigcup_{l_0} \bigcap_{l \geq l_0} \sup_{u \in S} |\hat{f}_l(u) - f(u)| < \varepsilon \right) = 1 \ , \forall \varepsilon > 0. \tag{7}$$

H8. Good training set: $Y^i = g^*(X^i)$ for all $(X^i, Y^i) \in \mathcal{D}^n$; moreover, there exist X_a^* in $\mathcal{D}^n$ such that $X_a^n \in A_a^{\delta_2}$, for $a = 0, 1$, for some $\delta_2 > 0$. Observe that H8 implies H4.

Even no condition is imposed on the bandwidth h_l, the algorithm assumes, implicitly, that it converges to zero. Indeed, in Proposition 3, we ask for rates of convergence to guarantee the validity of condition H3, H5 and H7, besides some regularity conditions on f and the sets I_a for $a = 0, 1$. Following the notation in Federer (1959), let $\mathrm{Unp}(S)$ be the set of points $x \in \mathbb{R}^d$ with a unique projection on S, denoted by $\pi_S(x)$. That is, for $x \in \mathrm{Unp}(S)$, $\pi_S(x)$ is the unique point that achieves the minimum of $\|x - y\|$ for $y \in S$. For $x \in S$, let $reach(S, x) = \sup\{r > 0 : B(x, r) \subset \mathrm{Unp}(S)\}$. The reach of S is defined by $\mathrm{reach}(S) = \inf\{\mathrm{reach}(S, x) : x \in S\}$, and S is said to be of positive reach if $\mathrm{reach}(S) > 0$.

Proposition 3. *Assume that H2 i) and ii) hold and that f is compact supported, continuous, bounded from below by a positive constant. Assume also that $reach(\overline{I_a^c}) > 0$, for $a = 0, 1$. The bandwidth h_l fulfils $h_l \to 0$ and $l h_l^{2d} / \log(l) \to \infty$. Then H3, H5 and H7 hold.*

The main result of this work is presented in Theorem 1; it states that the algorithm proposed in Section 4 is consistent, in the sense defined in (1). To prove this result, we will invoke the following preliminary lemmas. The first of them, Lemma 1, establishes that the first point classified differently from the Bayes rule is in the boundary region $B_1^h \cup B_0^h$. Then, in Lemma 2, we combine the valley condition with the uniform consistency of the kernel estimator to show that, asymptotically, there are more point of $\mathcal{X}_l$ in $A_0^\delta \cup A_1^\delta$ than in $B_0^{h_l} \cup B_1^{h_l}$. Lemma 3 states that all the points far enough from the boundary region are labeled by the algorithm, with the same label that the one given by the Bayes rule. To be more precise, recall that, $\mathcal{F}_l = \{\omega : \mathcal{X}_l(\omega) \quad \text{is entirely classified}\}$ and define

$$\mathcal{B}_l = \{\omega : \text{ there exists } X_{i_j} \in \mathcal{X}_l : \tilde{Y}_{i_j} \neq g^*(X_{i_j})\} \cap \mathcal{F}_l.$$

Look at the first time, j_{bad}, where the algorithm assigns a label different from that prescribed by the Bayes rule, if such a step exists; otherwise, define $j_{bad} = \infty$. Namely,

$$j_{bad} = \inf\{j : \tilde{Y}_{i_j} \neq g^*(X_{i_j})\} \quad \text{on } \mathcal{B}_l, \quad \text{and } j_{bad} = \infty \text{ on } \mathcal{B}_l^c. \tag{8}$$

From now on, we will say that a point $X_{i_j} \in \mathcal{X}_l$ is *badly classified* whenever $\tilde{Y}_{i_j} \neq g^*(X_{i_j})$; otherwise the point will be called well classified. The next result establishes that $X_{i_{j_{bad}}}$ is in $B_0^{h_l} \cup B_1^{h_l}$.

Lemma 1. *Assume that H1 and H8 hold. Then,* $\mathcal{B}_l \subset \{X_{i_{j_{bad}}} \in B_0^{h_l} \cup B_1^{h_l}\}$.

Lemma 2. *Assume H6 and H7. Then,* $\mathbb{P}(\mathcal{V}^\delta) = 1$, *for any* $\delta < \delta_1$ *where*

$$\mathcal{V}^\delta = \bigcup_{l_0} \bigcap_{l \geq l_0} \mathcal{V}_l^\delta \quad \textit{and} \quad \mathcal{V}_l^\delta = \left\{ \inf_{a \in A_0^\delta \cup A_1^\delta} \sum_{i=1}^{l} \mathbb{I}_{B(a,h_l)}(X_i) \;\geq\; \sup_{b \in B_0^{h_l} \cup B_1^{h_l}} \sum_{i=1}^{l} \mathbb{I}_{B(b,h_l)}(X_i) \right\}.$$

Lemma 3. *Assume H1–H8. Then, for any* $\delta < \min\{\delta_0, \delta_1, \delta_2\}$

$$\mathcal{F}_l \cap \mathcal{A}_{a,l}^\delta \cap \mathcal{V}_l^\delta \;\subset\; \left\{ \mathcal{X}_l \cap A_a^\delta \cap (\mathcal{Z}_{j_{bad}-1})^c = \emptyset \right\}, \quad a = 0, 1, \tag{9}$$

and therefore, on $\mathcal{F}_l \cap \mathcal{A}_{0,l}^\delta \cap \mathcal{A}_{1,l}^\delta \cap \mathcal{V}_l^\delta$, *we have that*

$$\mathbb{I}_{\tilde{Y}_i = g^*(X_i)} \geq \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i), \quad i = 1, \ldots, l. \tag{10}$$

Theorem 1. *Assume that* $\mathcal{D}_n$ *is a good training set, in the sense that fulfills H8. Then, under H1–H3, H5–H7, the algorithm presented in Section 4 satisfies*

$$\lim_{l \to \infty} \frac{1}{l} \sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l) \neq Y_i} = \mathbb{P}\{g^*(X) \neq Y\} \quad a.s.$$

and therefore, it is consistent, as defined in (1).

6 A faster algorithm

The algorithm presented in Section 4 classifies a few points of $\mathcal{X}_l$ at each step. This can be discouraging when l is too large. In order to overcome this issue, we will introduce a simple modification that gives rise to a faster procedure in terms of computational time (see Table 2), at the expense of

introducing a small increment in the classification error rate (this increment can be controlled but with computational cost).

The idea is to pre-process the sample $\mathcal{X}_l$, and *project it* on a grid $\mathcal{T}_l$, as we describe in what follows. Since S is a compact set, we can assume that $S \subset (a,b)^d$ with $a < b$. For N fixed, to be determined by the practitioner, consider $a_i = a + i(b-a)/N$ for $i = 0, \ldots, N$. The N-grid $\mathcal{T}$ on $(a,b)^d$ is determined by the N^d points of the form $\mathbf{a} = (a_{i_1}, \ldots, a_{i_d})$ with $i_j \in \{0, \ldots, N-1\}$, for $j = 1, \ldots, d$. Each point $\mathbf{a}$ in the grid determines a cell $\mathcal{C}_{\mathbf{a}} = \prod_{j=1}^{d} (a_{i_j}, a_{i_j+1}]$.

Given $\mathcal{X}_l$, let $\mathcal{T}_l$ be the set of points $\mathbf{a}$ in the grid whose corresponding cell $\mathcal{C}_{\mathbf{a}}$ interescts $\mathcal{X}_l$; now project (or collapse) $\mathcal{X}_l$ on $\mathcal{T}_l$, in the sense that the algorithm will be applied to $\mathcal{T}_l$ in lieu of $\mathcal{X}_l$. Then, all the points in $\mathcal{X}_l \cap \mathcal{C}_{\mathbf{a}}$ will be classified with the label assigned to $\mathbf{a}$ by the algorithm.

7 Examples with simulated and real data

In this section we report some numerical results, comparing the performance of three algorithms that can be used in the semi supervised framework discussed in this work. k–nearest neighbors (k–NN) is the first of them and labels each element in $\mathcal{X}_l$ according the the majority rule on the basis of the training sample $\mathcal{D}_n$.

The second one is the algorithm presented in Section 4 , while the last one is its faster version, introduced in Section 6.

The classification error rate of each algorithm is computed in three scenarios. In the first two, we use artificially generated data, whereas in the last one we employ a real data set. The first example compares efficiency of the three algorithms. The second one shows the effect of the grid size with respect to classification error rate and computational time. The third one is a well known real-data set where we illustrate the crucial effect of the initial training sample $\mathcal{D}^n$.

7.1 A first simulated example

The joint distribution of (X, Y) is generated as follows: consider first the curve C in the square $[-1,1]^2$, defined by $C = \{(x, (1/2)\sin(4x)) : -1 \leq x \leq 1\}$. All the points in the square that are below C will be labeled with $Y = 0$ while those that are above the curve C will be labeled with $Y = 1$. Now, to emulate the valley condition, those points close to C will be chosen with less probability than those far away. To do so, let S_1 and S_2 denote the set of points in the square which are at $\|\cdot\|_\infty$-distance larger / smaller

than 0.2 from C, respectively. Namely, $S_1 = \{B_{\|\cdot\|_\infty}(C, 0.2)\}^c \cap [-1, 1]^2$ and $S_2 = B_{\|\cdot\|_\infty}(C, 0.2) \cap [-1, 1]^2$, where $\|\cdot\|_\infty$ is the supremum norm. Let U_1, U_2 and B be independent random variables, with $U_1 \sim \text{Uniform}(S_1)$, $U_2 \sim \text{Uniform}(S_2)$ and $B \sim Bernoulli(7/8)$. Consider the random variable $X = BU_1 + (1 - B)U_2$, while $(X, Y) = ((X_1, X_2), 1)$ if $X_2 > (1/2)\sin(4X_1)$ and $(X, Y) = ((X_1, X_2), 0)$ if $X_2 \leq (1/2)\sin(4X_1)$. To compare the performance of different classifiers, we generate $\mathcal{X}_l = \{X_1, \ldots, X_l\}$ iid, with X_i distributed as X and sample size $l = 2400$, while $\mathcal{D}^n = \{(X^1, Y^1), \ldots, (X^n, Y^n)\}$ is a sample of size $n = 20$ of vectors iid, distributed as (X, Y).

Figure 2 exhibits the labels assigned by three different methods to a fixed realization of both $\mathcal{X}_l$ and $\mathcal{D}_n$. In the first panel we show the labels assigned by k – NN, with k=5; the second panel corresponds to the output of the algorithm ran with bandwidth $h_l = 0.15$; and the third one corresponds to the faster version of the algorithm, presented in Section 6, using a N-grid with $N = 21$ (distance 0.1 between points in each dimension). The classification error rates corresponding to each method are 0.135, 0.027, and 0.038, respectively. Finally, we study the performance of our algorithm, analyzing the error rates among 50 replications of the described scheme ($n = 20$, $l = 2400$ and $h_l = 0.15$). An histogram of the classification errors is presented on the right panel of Figure 3 and a summary is reported in Table 1. There are four out of the fifty replications where the classification errors are much higher than in the other cases. These extreme results can be attributed to the initial training sample $\mathcal{D}^n$ (see assumption H8). The initial training sample for the best and the worst case (in terms of classification error rate) are also shown on the left panels of Figure 3.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.0179	0.0267	0.0365	0.0458	0.0469	0.1554

Table 1: Summary of the classification error rate over 50 repetitions.

7.2 A second example using simulated data

To generate the data consider two bi-variate normal random vectors $Z_0 \sim N(\mu_0, \Sigma)$ and $Z_1 \sim N(\mu_1, \Sigma)$. Let $Y \sim Bernoulli(0.5)$. The conditional distribution of X given $Y = y$, for $y = 0, 1$, is given by $X \mid Y = y \sim Z_y \mid \|Z_y - \mu_y\| < 1.5$.

We consider two cases: $\mu_0 = (1.5, 1.5)$, $\mu_1 = (0, 0)$ (see Figure 4 left) and $\mu_0 = (1.2, 1.2)$, $\mu_1 = (0.0)$ (see Figure 4 right); in both cases $\Sigma =$

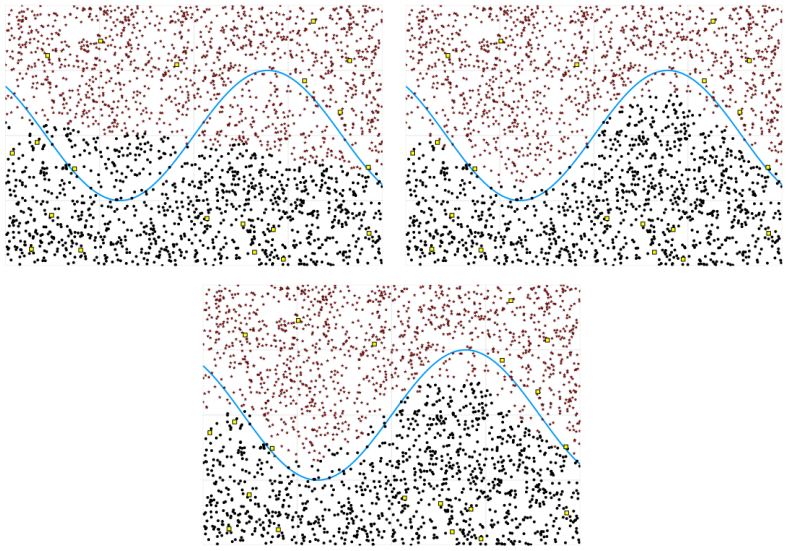

Figure 2: Labels assigned by three different methods to a fixed realization of both $\mathcal{X}_l$ and $\mathcal{D}_n$. Red stars are points labeled as 1 while black dots are labeled as 0. The initial training sample $\mathcal{D}^n$ is represented as yellow squares. Left panel: labels assigned by k – NN, with k=5. Middle panel: output of the algorithm ran with bandwidth $h_l = 0.15$. Right panel: faster version of the algorithm, presented in Section 6, using a N-grid with $N = 21$ (distance 0.1 between points in each dimension).

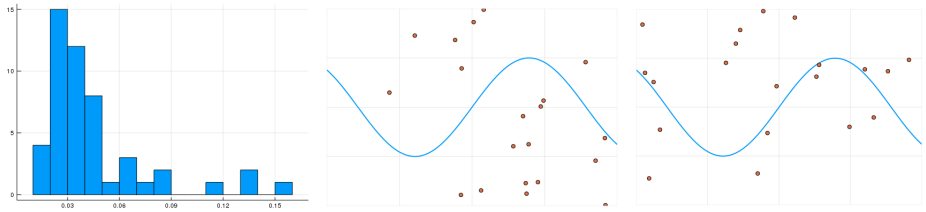

Figure 3: Left panel: histogram of the classification error of our algorithm ($n = 20$, $l = 2400$ and $h_l = 0.15$) among 50 replications. Middle panel: Initial training sample in the worst case. Right panel: Initial training sample in the best case.

diag$(0.6, 0.6)$. In the first case the Bayes error is 0.025 and in the second one is 0.067.

We generate $\mathcal{X}_l = (X_1, \ldots, X_l)$ iid, with X_i distributed as X, and sample size $l = 2000$. In each replication, we used $\mathcal{D}_n$ $\{((0,0),1),((1.5,1.5),0)\}$ and bandwidth $h = 0.4$ to run the algorithm.

The average of the computational time as well as the error rate over 50 replications are reported in Table 2, for different grid sizes. As it is shown in Table 2, there is a trade-off between computation time and efficiency. However, if the cell sizes of the grid are reasonably small (as in the first column of Table 2), the classification errors are essentially the same, while the computational time decreases. The simulation was performed in Julia 1.0.2, running on an Intel i7-8550U.

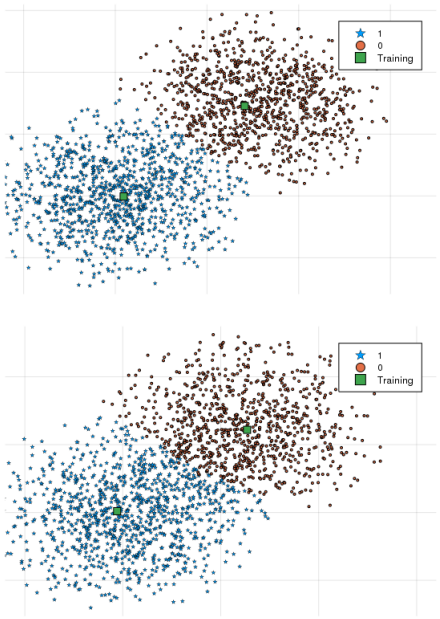

Figure 4: The two populations of bi-variate truncated Gaussian distributions.

7.3 A real data example

We consider the well known Isolet data set of speech features from the UCI Machine Learning Repository Asuncion and Newman (2007), comprising 617

Without Grid		Grid step 0.1		Grid step 0.15	
First Case					
Time	Error	Time	Error	Time	Error
4.2s	0.0323	2.8s	0.043	1.1s	0.046
Second Case					
Time	Error	Time	Error	Time	Error
5.15s	0.084	2.7s	0.10	0.98s	0.117

Table 2: Average of the computation time and miss-classification errors over 50 replications.

attributes associated to the English pronunciation of the 26 letters of the alphabet. The data come from 150 people who spoke the name of each letter twice. There are three missing data, not considered in the study. Feature vectors include: spectral coefficients, contour features, sonorant features, pre-sonorant features, and post-sonorant features, and are described in Fanty and Cole (1991). The spectral coefficients account for 352 of the features. The exact order of appearance of the features is not known.

We apply the semi-supervised algorithm to the binary problem given by the E-set comprising the letters $\{b, c, d, e, g, p, t, v, z\}$ and the R-set with the remaining letters except for the letters $\{m, n\}$, starting with a small labeled data set of 10 elements from each group. Then $\mathcal{D}^n$ consists of 20 data.

To pre-process the data, we first removed the first repetition of every letter. Next, we kept only those data whose nearest neighbour is at a distance smaller than a threshold (the value 8 was selected to reduce the classification error, and to reduce the computational time, in order repeat it 100 times). This pruning procedure reduced the sample $\mathcal{X}_l$ to 2171 data. To study how the classification error varies with respect to the training sample, we randomly chose a training sample 100 times. A summary of the classification error rate is shown in Table 3, while the density of the errors is shown in Figure 5.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.028	0.076	0.139	0.130	0.184	0.247

Table 3: Summary of the missclassification error rate over 100 replications.

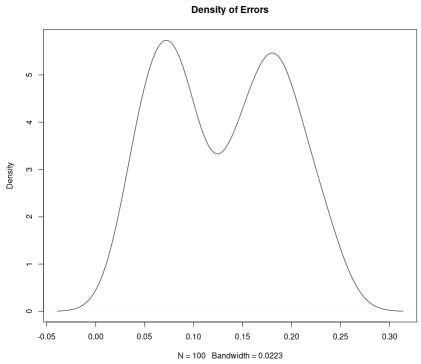

Figure 5: Density of the errors.

8 Some remarks regarding the assumptions

We discuss briefly the set of assumptions considered. Firstly, we would like to point out that the results we are presenting in this work are quite ambitious, since the training sample is frozen at a small fixed size n, while the asymptotic is on the size l of the unlabeled data set. These facts should not be misinterpreted. Without these hypotheses, the semi-supervised classification methods may work better than the classical supervised classification methods but the consistency will not be verified if the size n of the training sample remains fixed.

1) In order for an algorithm to work for the semi-supervised classification problem, the initial training sample $\mathcal{D}^n$ (whose size does not need to tend to infinity) must be well located. We require that $\mathcal{D}^n = (\mathcal{X}^n, \mathcal{Y}^n)$ satisfies $Y^i = g^*(X^i)$ for all $i = 1, \ldots, n$, which is a quite mild hypothesis. In many applications, a stronger condition can be assumed. For instance, if the two populations are sick or healthy, the initial training sample can be chosen as the set of individuals for whom the covariate X ensures the condition on the patient, that is, $\mathbb{P}(Y = 1|X) = 1$ or $\mathbb{P}(Y = 1|X) = 0$. On the other hand, if the initial training sample is not well located, then any algorithm might classify almost all observations wrongly. Indeed, consider the case where the distribution of the population with label 0 is $N(0, 1)$ and the other is $N(1, 1)$. This will be the case if we start for instance with the pairs $\{(0.4, 1), (0.6, 0)\}$.
The effect of the initial training sample $\mathcal{D}_n$ is illustrated in the real-

data example where the classification error varies between 0.028 and 0.247 by changing at random $\mathcal{D}_n$.

2) The connectedness of I_0 and I_1 is also critical. In a situation like the one shown in Figure 6, the points in the connected component for which there is no point in $\mathcal{D}^n$ (represented as squares) will be classified as the circles by the algorithm. However, if I_0 and I_1 have a finite number of connected components and there is at least one pair $(X^i, Y^i) \in \mathcal{D}^n$ in each of them with $g^*(X^i) = Y^i$, it is easy to see that the algorithm will be consistent.

3) The uniform kernel can be replaced by any regular kernel satisfying $c_1 I_{B(0,1)}(u) \leq K(u) \leq c_2 I_{B(0,1)}(u)$, for some positive constants c_1, c_2, and the results still hold.

4) We also assume that P_X has a continuous density f with compact support S. If that is not the case, it is possible to take a large enough compact set S such that $P_X(S^c)$ is very small and therefore just a few data from $\mathcal{X}^l$ is left out.

5) The following example shows that H5 is necessary for consistency. Indeed, suppose that $U_1 := X|Y = 1 \sim U([a, 1])$ and $U_0 = X|Y = 0 \sim U([0, a])$ with $a = P(Y = 0)$, then for all $a \in [0, 1]$, $P_X = aU_0 + (1 - a)U_1 \sim U([0, 1])$. Unless the training sample $\mathcal{D}^n$ contains two points $(X_1, 0)$ and $(X_2, 1)$ with X_1 and X_2 very close to a, semisupervised methods will fail. Regardless of the value of a, the classes 0 and 1 are indistinguishable since the joint distribution is in all cases $U[0, 1]$.

   Moreover, there is no consistent semi-supervised algorithm for n fixed. To see this, consider $(X_1, Y_1), \ldots, (X_n, Y_n)$ a training sample in $[0, 1]$ with fixed size n. Let us denote $X_m = \min\{X_i : (X_i, 1) \in \mathcal{D}_n\}$, $X^M = \max\{X_i : (X_i, 0) \in \mathcal{D}_n\}$, $\mathcal{X}_l \cap (X_m, X^M) = \{X_{i_1}, \ldots X_{i_k}\}$ and $\tilde{Y}_{i_1}, \ldots, \tilde{Y}_{i_k}$ the labels assigned by any algorithm. Then if $\sum_{j=1}^k \tilde{Y}_{i_k} > k/2$, conditioned to the training sample $\mathcal{D}_n$, if we choose $a = X^M$ we will miss-classify at least $k/2$ data-points, and if $\sum_{j=1}^k \tilde{Y}_{i_k} \leq k/2$, $a = X_m$ we will do the same.

Appendix A

Proof of Proposition 1.
Observe that $\mathbb{P}\big(g_i(\mathcal{X}_l) \neq Y_i \mid \mathcal{X}_l \setminus X_i\big) \geq \mathbb{P}(g^*(X_i) \neq Y_i)$, for $i = 1, \ldots, l$.

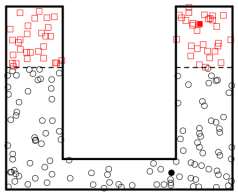

Figure 6: The points labeled as 0 are represented with squares while the points labeled as 1 are represented with circles. Filled points belong to $\mathcal{D}^n$.

Thus,

$$\mathbb{E}\Big(\mathbb{I}_{g_i(\mathcal{X}_l)\neq Y_i}\Big) = \mathbb{P}(g_i(\mathcal{X}_l) \neq Y_i) = \mathbb{E}\Big(\mathbb{P}\big(g_i(\mathcal{X}_l) \neq Y_i | \mathcal{X}_l \backslash X_i\big)\Big) \geq \mathbb{P}(g^*(X_i) \neq Y_i),$$

and therefore, $L(\mathbf{g}_l) = \mathbb{E}\Big(\frac{1}{l}\sum_{i=1}^{l} I_{g_i(\mathcal{X}_l)\neq Y_i}\Big) \geq \mathbb{P}(g^*(X_i) \neq Y_i)$, showing that $L(\mathbf{g}_l) \geq \mathbb{P}(g^*(X) \neq Y)$, for any $\mathbf{g}_l = (g_1, \ldots, g_l)$. The lower bound is attained by choosing the ith coordinate of $\mathbf{g}_l$ equal to $g^*(X_i)$. Moreover, the accuracy of $\mathbf{g}_l^*$ equals that of a single coordinate; namely $L(\mathbf{g}_l^*) = \mathbb{P}(g^*(X) \neq Y) = L^*$.

Proof of Proposition 2.
We will prove that if H1, H2 i) and H4 are satisfied, then $\mathcal{I}_{0,l} \cap \mathcal{I}_{1,l} \subset \mathcal{F}_l$. Combining this inclusion with H3 we conclude that $\mathbb{P}(\mathcal{F}) = 1$. To prove that $\mathcal{I}_{0,l} \cap \mathcal{I}_{1,l} \subset \mathcal{F}_l$, we will see that if

$$I_a \subseteq \bigcup_{X \in \mathcal{X}_l \cap I_a} B(X, h_l/2)\,, \quad a = 0, 1, \tag{11}$$

all the elements of $\mathcal{X}_l$ are label by the algorithm. To do so, note that, by H4, there exists X_a^* in $\mathcal{X}^n$ such that $X_a^* \in I_a$, for $a = 0, 1$. We will now prove that the algorithm starts. Since X_1^* is in I_1 and (11) holds with $a = 1$, there exists $X_j^1 \in \mathcal{X}_l \cap I_1$ with $d(X_1^*, X_j^1) < h_l$. In particular, $d(\mathcal{X}^n, X_j^1) < h_l$ and so $X_j^1 \in \mathcal{U}_0(h_l)$. This guarantees that $\mathcal{U}_0(h_l) \neq \emptyset$ and hence the algorithm can start.

Assume now that we have classified $j < l$ points of $\mathcal{X}_l$. We will prove that there exists at least one point satisfying the iteration condition required at step $j + 1$: $\mathcal{U}_j(h_l) \neq \emptyset$. By H1 we can assume that $\mathcal{U}_j = \mathcal{U}_j \cap (I_0 \cup I_1)$. Take a such that $\mathcal{U}_j \cap I_a \neq \emptyset$. We will consider now two possible cases: (i) if $\mathcal{X}_l \cap I_a \cap \mathcal{U}_j^c = \emptyset$, then $\mathcal{X}_l \cap I_a = \mathcal{X}_l \cap I_a \cap \mathcal{U}_j$ and so, by (11), $X_a^* \in B(X, h_l/2)$

for some $X \in \mathcal{X}_l \cap \mathcal{U}_j$. Since X_a^* is in $\mathcal{Z}_j$ and $X \in \mathcal{U}_j$, we conclude that $X \in \mathcal{U}_j(h_l)$. Assume now that (ii) $\mathcal{X}_l \cap I_a \cap \mathcal{U}_j^c \neq \emptyset$. Since I_a is connected and (11) holds, the union of $B(X, h_l/2)$, with $X \in \mathcal{X}_l \cap I_a$, is also a connected set and, therefore,

$$\left(\bigcup_{X \in \mathcal{X}_l \cap I_a \cap \mathcal{U}_j^c} B(X, h_l/2) \right) \bigcap \left(\bigcup_{X \in \mathcal{X}_l \cap I_a \cap \mathcal{U}_j} B(X, h_l/2) \right) \neq \emptyset.$$

Finally, take $X \in \mathcal{X}_l \cap I_a \cap \mathcal{U}_j^c$ and $\tilde{X} \in \mathcal{X}_l \cap I_a \cap \mathcal{U}_j$ such that $B(X, h_l/2) \cap B(\tilde{X}, h_l/2) \neq \emptyset$ to conclude that $\tilde{X} \in \mathcal{U}_j(h_l)$.

Proof of Lemma 1.
By H1, we can assume that $\eta(X) \neq 1/2$ for all $X \in \mathcal{X}^n \cup \mathcal{X}_l$. Assume first that $\eta(X_{j_{bad}}) > 1/2$, that is, $X_{j_{bad}} \in I_1$, $\tilde{Y}_{j_{bad}} = 0$, and all the points labeled up to the step $j_{bad} - 1$ by the algorithm are well classified. Now, suppose by contradiction $X_{j_{bad}} \notin B_1^{h_l}$, which means that $X_{j_{bad}} \notin B(I_0, h_l)$ and thus, $B(X_{j_{bad}}, h_l) \cap I_0 = \emptyset$. This implies that $g^*(X) = 1$ for all $X \in (\mathcal{X}^n \cup \{X_{i_1}, \ldots, X_{j_{bad}-1}\}) \cap B(X_{j_{bad}}, h_l)$, contradicting the label assigned to $X_{j_{bad}}$ according to the majority rule that is used by the algorithm. Thus, $B(X_{j_{bad}}, h_l) \cap I_0 \neq \emptyset$, and so $X_{j_{bad}} \in B_1^{h_l}$. Analogously, if $\eta(X_{j_{bad}}) < 1/2$, we deduce that $X_{j_{bad}} \in B_0^{h_l}$.

Proof of Lemma 2.
Given $\delta < \delta_1$, choose ε such that $\gamma(\delta) - 2\varepsilon > 0$, for $\gamma(\delta)$ introduced in H6. We will prove $\mathcal{S}_l^\varepsilon = \{\sup_{u \in S} |\hat{f}_l(u) - f(u)| < \varepsilon\}$ is included in $\mathcal{V}_l^\delta$ as far as $h_l < \delta$ and therefore, from (7), we conclude that $\mathbb{P}(\mathcal{V}^\delta) = 1$.

Now, note that on $\mathcal{S}_l^\varepsilon$, we get that $f(u) - \varepsilon < \hat{f}_l(u) < f(u) + \varepsilon$, and so, on $\mathcal{S}_l^\varepsilon$, for $a \in A_0^\delta \cup A_1^\delta$ and $b \in B_1^h \cup B_0^h$, $\hat{f}_l(b) < f(b) + \varepsilon < f(a) - \gamma + \varepsilon < \hat{f}_l(a) + 2\varepsilon - \gamma$. Thus, on $\mathcal{S}_l^\varepsilon$,

$$\sup_{b \in B_0^{h_l} \cup B_1^{h_l}} \hat{f}_l(b) \leq \inf_{a \in A_0^\delta \cup A_1^\delta} \hat{f}_l(a) + 2\varepsilon - \gamma < \inf_{a \in A_0^\delta \cup A_1^\delta} \hat{f}_l(a),$$

when $2\varepsilon - \gamma < 0$. This proves that $\mathcal{S}_l^\varepsilon \subseteq F_l^\delta$, for l such that $8h_l < \delta$.

Proof of Lemma 3.
When $j_{bad} = \infty$, $\mathcal{Z}_{j_{bad}-1} = \mathcal{X}_n \cup \mathcal{X}_l$. This fact implies that, on the event $\mathcal{F}_l \cap \mathcal{B}_l^c$, the following identity holds: $\mathcal{X}_l \cap (\mathcal{Z}_{j_{bad}-1})^c = \emptyset$. Thus, to prove (9), we need to show that, for $a = 0, 1$ $\mathcal{F}_l \cap \mathcal{A}_{a,l}^\delta \cap \mathcal{V}_l^\delta \cap \mathcal{B}_l \subset \left\{\mathcal{X}_l \cap A_a^\delta \cap (\mathcal{Z}_{j_{bad}-1})^c = \emptyset\right\}$. We will argue by contradiction, assuming that there exists $\omega \in \mathcal{F}_l \cap \mathcal{A}_{a,l}^\delta \cap$

$\mathcal{V}_l^\delta \cap \mathcal{B}_l$ for which $\emptyset \neq \mathcal{X}_l \cap A_a^\delta \cap (\mathcal{Z}_{j_{bad}-1})^c = \{W_1, \ldots, W_m\}$. Invoking H8, $\mathcal{X}^n \subseteq \mathcal{Z}_{j_{bad}-1}$ and there exists $X_a^* \in A_a^\delta \cap \mathcal{X}^n$. These facts guarantee that $X_a^* \in A_0^\delta \cap \mathcal{Z}_{j_{bad}-1}$, and since we are working on $\mathcal{A}_{a,l}^\delta$, we get that

$$X_a^* \in A_a^\delta \subseteq \bigcup_{X \in \mathcal{X}_l \cap A_a^\delta} B(X, h_l/2) \quad \text{and} \quad X_a^* \in \mathcal{Z}_{j_{bad}-1}. \tag{12}$$

Next, we will argue that there exist $W^* \in \{W_1, \ldots, W_m\}$ such that $d(W^*, \mathcal{Z}_{j_{bad}-1}) < h_l$. To do so, consider the following two cases:

(i) $\mathcal{X}_l \cap A_a^\delta \cap \mathcal{Z}_{j_{bad}-1} = \emptyset$. In such a case, from (12) we get that A_a^δ can be covered by balls centered at $\{W_1, \ldots, W_m\}$ and, since $X_a^* \in A_a^\delta$, $X_a^* \in B(W^*, h_l/2)$ for some $W^* \in \{W_1, \ldots, W_m\}$. Therefore, $d(X_a^*, W^*) < h_l$. Recalling that, as stated in (12), $X_a^* \in \mathcal{Z}_{j_{bad}-1}$., we conclude that $d(W^*, \mathcal{Z}_{j_{bad}-1}) < h_l$.

(ii) Assume now that $\mathcal{X}_l \cap A_a^\delta \cap \mathcal{Z}_{j_{bad}-1} \neq \emptyset$. Since A_a^δ is connected, the union of balls given in (12) is connected, and then,

$$\left\{ \bigcup_{X \in \mathcal{X}_l \cap A_a^\delta \cap \mathcal{Z}_{j_{bad}-1}} B(X, h_l/2) \right\} \bigcap \left\{ \bigcup_{1 \leq i \leq m} B(W_i, h_l/2) \right\} \neq \emptyset.$$

Thus, there exist $X \in \mathcal{Z}_{j_{bad}-1}$ and $W^* \in \{W_1, \ldots, W_m\}$ with $d(X^*, W^*) < h_l$, which implies that $d(W^*, \mathcal{Z}_{j_{bad}-1}) < h_l$.

To finish the proof, we will show that such a W^* should have been chosen by the algorithm to be labeled before $X_{j_{bad}}$, which implies that $W^* \in \mathcal{Z}_{j_{bad}-1}$, contradicting that $W^* \in (\mathcal{Z}_{j_{bad}-1})^c$. This contradiction show that no such W^* exists, as announced. Since $d(W^*, \mathcal{Z}_{j_{bad}-1}) < h_l$, we get that $W^* \in \mathcal{U}_{j_{bad}-1}(h_l)$, the set of candidates to be labeled by the algorithm at step j_{bad}. Indeed, since $W^* \in A_a^\delta$ and $h < \delta$, $B(W^*, h_l) \subseteq I_a$. Thus, $\hat{\eta}_{j_{bad}-1}(W^*) = a$ implying that W^* attains the maximum stated in (3). Invoking now Lemma 2, since $W^* \in A_a^\delta$ while $X_{j_{bad}}$ is in $B_0^h \cap B_1^h$ (see Lemma 1), we know that $\#\{\mathcal{X}_l \cap B(W^*, h_l)\} \geq \#\{\mathcal{X}_l \cap B(X_{j_{bad}}, h_l)\}$; thus, W^* should have been chosen before $X_{j_{bad}}$. This conclude the prof of the result.

Proof of Theorem 1

Recall that $g_{n,l,r(i)}(\mathcal{X}_l)$ denotes the label assigned by the algorithm to the

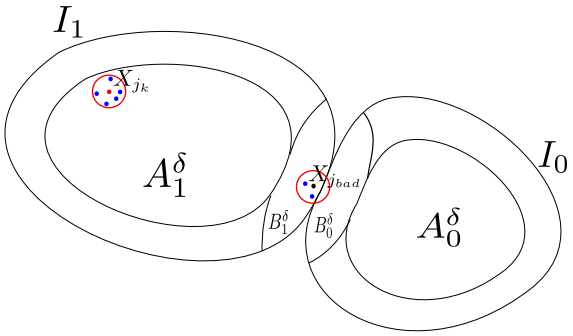

Figure 7: We show: in black $X_{j_{bad}}$, in red X_{j_k}, in blue we represent the points of $\mathcal{X}_{l_k}$ belonging to $B(X_{j_k}, h_{l_k})$ and $B(X_{j_{bad}}, h_{l_k})$.

observation$X_i \in \mathcal{X}_l$. The empirical mean accuracy of classification satisfies

$$\begin{aligned}\frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i} \geq & \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i}\ \mathbb{I}_{g^*(X_i)=Y_i}\ \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i) \\ = & \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=g^*(X_i)}\ \mathbb{I}_{g^*(X_i)=Y_i}\ \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i).\end{aligned}$$

Consider $\mathcal{T}_l^\delta = \mathcal{F}_l \cap \mathcal{A}_{0,l}^\delta \cap \mathcal{A}_{1,l}^\delta \cap \mathcal{V}_l^\delta$, and $\mathcal{T}^\delta = \bigcup_{l_0} \bigcap_{l \geq l_0} \mathcal{T}_l^\delta$. Combining the results obtained in Proposition 2 and Lemma 2 with condition H5, we conclude that $\mathbb{P}(\mathcal{T}^\delta) = 1$, for $\delta < \min\{\delta_0, \delta_1, \delta_2\}$. By (10), on $\mathcal{T}_l$, we have that $\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=g^*(X_i)} \geq \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i)$ for all $i = 1, \ldots, l$, and therefore

$$\frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=g^*(X_i)}\ \mathbb{I}_{g^*(X_i)=Y_i}\ \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i) \geq \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g^*(X_i)=Y_i}\ \mathbb{I}_{A_0^\delta \cup A_1^\delta}(X_i).$$

Then, on $\mathcal{T}^\delta$, we have that $\liminf_{l\to\infty} \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i} \geq \mathbb{P}\{g^*(X) = Y, X \in A_0^\delta \cup A_1^\delta\}$ and so

$$\liminf_{l\to\infty} \frac{1}{l}\sum_{i=1}^{l} \mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i} \geq \mathbb{P}\{g^*(X) = Y\} \quad a.s. \tag{13}$$

On the other hand,

$$\frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i} = \frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i}\mathbb{I}_{g^*(X_i)=Y_i} + \frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i}\,\mathbb{I}_{g^*(X_i)\neq Y_i}$$
$$\leq \frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g^*(X_i)=Y_i} + \frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)\neq g^*(X_i)}$$

From Lemma 3, on $\mathcal{T}_l^\delta$, $\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)\neq g^*(X_i)} \leq \mathbb{I}_{(A_0^\delta \cup A_1^\delta)^c}(X_i)$, and therefore, on $\mathcal{T}^\delta$,

$$\limsup_{l\to\infty}\frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)=Y_i} \leq \mathbb{P}(g^*(X)=Y) + \mathbb{P}(X\notin\{A_0^\delta\cup A_1^\delta\}).$$

By H2 ii), the last term in the previous display converges to zero when $\delta\to 0$, and thus

$$\limsup_{l\to\infty}\frac{1}{l}\sum_{i=1}^{l}\mathbb{I}_{g_{n,l,r(i)}(\mathcal{X}_l)} \leq \mathbb{P}(g^*(X)=Y) \quad a.s. \tag{14}$$

Combining (13) and (14) we deduce the announced convergence. The consistency defined in (1) follows from the Dominated convergence theorem.

Appendix B

In this section we will prove that under $H2$, conditions $H3$, and $H6$ holds if we impose some geometric restrictions on I_0 and I_1. In order to make this Appendix self contained, we need some geometric definitions and also include some results which will be invoked.

First we introduce the concept of Hausdorff distance. Given two compact non-empty sets $A, C \subset \mathbb{R}^d$, the *Hausdorff distance* or *Hausdorff–Pompei distance* between A and C is defined by $d_H(A,C) = \inf\{\varepsilon \geq 0 : \text{such that } A \subset B(C,\varepsilon) \text{ and } C \subset B(A,\varepsilon)\}$.

Next, we define standard sets, according to Cuevas and Fraiman (1997) (see also Cuevas and Rodríguez-Casal (2004)).

Definition 1. *A bounded set $S \subset \mathbb{R}^d$ is said to be standard with respect to a Borel measure μ if there exists $\lambda > 0$ and $\beta > 0$ such that $\mu\big(B(x,\varepsilon)\cap S\big) \geq \beta\mu_L(B(x,\varepsilon))$ for all $x \in S$, $0 < \varepsilon \leq \lambda$, where μ_L denotes the Lebesgue measure on $\mathbb{R}^d$.*

Roughly speaking, standardness prevents the set from having peaks that are too sharp.

The following theorem is proved in Cuevas and Rodríguez-Casal (2004)).

Theorem 2. *(Cuevas and Rodríguez-Casal (2004)) Let $Z_1, Z_2, \ldots$ be a sequence of iid observations in $\mathbb{R}^d$ drawn from a distribution P_Z. Assume that the support Q of P_Z is compact and standard with respect to P_Z. Then*

$$\limsup_{l\to\infty} \left(\frac{l}{\log(l)}\right)^{1/d} d_H(\mathcal{Z}_l, Q) \leq \left(\frac{2}{\beta\omega_d}\right)^{1/d} \qquad a.s., \tag{15}$$

where $\omega_d = \mu_L(B(0,1))$, $\mathcal{Z}_l = \{Z_1, \ldots, Z_l\}$, and β is the standardness constant introduced in Definition 1.

Remark 1. *Theorem 2 implies that, if we choose $\epsilon_l = C\left(\frac{\log(l)}{l}\right)^{1/d}$ with $C > (2/(\beta\omega_d))$, then $Q \subset \cup_{i=1}^{l} B(Z_i, \epsilon_l)$ for l large enough. This in turn implies that if Q is connected, $\cup_{i=1}^{l} B(X_i, \epsilon_l)$ is connected.*

As a consequence of Theorem 2, we get the following covering property that will be used to prove Proposition 2 and H5 alone Proposition 3.

Lemma 4. *Let $X_1, X_2, \ldots$ be a sequence of iid observations in $\mathbb{R}^d$ drawn from a distribution P_X with support S. Let $Q \subset S$, be compact and standard with respect to P_X restricted to Q, with $P_X(Q) > 0$. Consider $(h_l)_{l\geq 1}$ such that $h_l \to 0$ and $lh_l^d/\log(l) \to \infty$. Then, with probability one, for l large, $Q \subset \bigcup_{X\in\mathcal{X}_l\cap Q} B(X, h_l/2)$, where $\mathcal{X}_l = \{X_1, \ldots, X_l\}$.*

Proof. We need to work with $\mathcal{X}_l$ restricted to Q, in order to do that, consider the sequence of stopping times defined by $\tau_0 \equiv 0$, $\tau_1 = \inf\{l : X_l \in Q\}$, $\tau_j = \inf\{l \geq \tau_{j-1} : X_l \in Q\}$, and the sequence of visits to Q given by $Z_j := X_{\tau_j}$. Then, $(Z_j)_{j\geq 1}$ are iid, distributed as $X \mid (X \in Q)$, with support Q. Observe that the distribution P_Z of Z is the restriction of P_X to Q. Since Q is compact and standard wrt P_Z we can invoke Theorem 1 for $(Z_j)_{j\geq 1}$, in order to conclude that there exists a positive constant C_Q depending on Q, such that for $k \geq k_0 = k_0(\omega)$,

$$d_H(\mathcal{Z}_k, Q) \leq C_Q(\log(k)/k)^{1/d}, \tag{16}$$

where $\mathcal{Z}_k = \{Z_1, \ldots, Z_k\}$. Define now V_l as the number of visits to the set Q up to time l. Namely, $V_l = \sum_{i=1}^{l} I_{\{X_i\in Q\}}$. By the law of large numbers, $V_l/l \to P(X \in Q) > 0$ a.e., and therefore, for l large enough, $V_l \geq k_0$. Thus, by (16), recalling that $h_l^d l/\log(l) \to \infty$, we get that

$$d_H(\mathcal{Z}_{V_l}, Q) \leq C_Q(\log(V_l)/V_l)^{1/d} \leq \tilde{C}_Q(\log(l)/l)^{1/d} \leq \frac{h_l}{2}.$$

In particular, $Q \subseteq \bigcup_{Z_j \in \mathcal{Z}_{V_l}} B(Z_j, h_l/2) = \bigcup_{X \in \mathcal{X}_l \cap Q} B(X, h_l/2)$.

□

This last lemma will be applied to get the covering properties stated in H2 and H5 for I_a and A_a^δ. The following results are needed to show that these sets satisfy the conditions imposed in Lemma 4.

Lemma 5. *Let ν be a distribution with support I such that $int(I) \neq \emptyset$ and $reach(\overline{I^c}) > 0$. Assume that ν has density f bounded from below by $f_0 > 0$. Let $Q = \overline{I \ominus B(0,\gamma)}$ such $\nu(Q) > 0$, then Q is standard with respect to ν_Q, the restriction of ν to Q (i.e $\nu_Q(A) = \nu(A \cap Q)/\nu(Q)$), for all $0 \leq \gamma < reach(I^c)$, with $\beta = f_0/(3\nu(Q))$.*

Proof. Let $0 \leq \gamma < reach(\overline{I^c})$. By corollary 4.9 in Federer (1959) applied to I^c, we get that $reach(\overline{(I \ominus B(0,\gamma))^c}) \geq reach(\overline{I^c}) - \gamma > 0$, and now by proposition 1 in Aaron, Cholaquidis and Cuevas (2017), ν_Q is standard, with $\beta = f_0/(3\nu(Q))$ (see Definition 1). □

Lemma 6. *Let $I \subset \mathbb{R}^d$ be a non-empty, connected, compact set with $reach(\overline{I^c}) > 0$. Then for all $0 < \varepsilon \leq reach(I^c)$, $I \ominus B(0,\varepsilon)$ is connected.*

Proof. Let $0 < \varepsilon \leq reach(\overline{I^c})$. By corollary 4.9 in Federer (1959) applied to I^c , $reach(I \ominus B(0,\varepsilon)) > \varepsilon$. Then, the function $f(x) = x$ if $x \in I \ominus B(0,\varepsilon)$, and $f(x) = \pi_{\partial(I \ominus B(0,\varepsilon))}(x)$ if $x \in I \setminus (I \ominus B(0,\varepsilon))$ where $\pi_{\partial S}$ denotes the metric projection onto ∂S, is well defined. By item 4 of theorem 4.8 in Federer (1959), f is a continuous function, so it follows that $f(I) = I \ominus B(0,\varepsilon)$ is connected. □

Proof of Proposition 3.
Since $reach(\overline{I_a^c}) > 0$ $P_X(\partial I_a) = 0$ (this follows from Proposition 1 and 2 in Cuevas, Fraiman and Pateiro-López (2012) together with Proposition 2 in Cholaquidis et al. (2014)), then $\mathbb{P}(X \in int(I_a)) = P(X \in I_a) > 0$. By Lemma 5, choosing $\gamma = 0$, the set $\overline{I_a}$ is standard with respect to P_X restricted to $\overline{I_a}$, for $a = 0, 1$. By Lemma 4, with $Q = \overline{I_a}$, $\overline{I_a}$ is coverable; finally we get that H3 is satisfied.

To prove H5 i) observe that the connectedness of A_a^δ follows from that of I_a (H2 i) together with Lemma 6. For H5 ii), take δ small enough such that $\mathbb{P}(X \in A_a^\delta) > 0$, which should exist because of H2 ii). By (1) in Erdös (1945), using that $\partial A_a^\delta \subset \{x : d(x, \partial I_a) = \delta\}$, we get that $\mathbb{P}(X \in \partial A_a^\delta) = 0$. Finally to prove the covering stated in H5 first observe that, by Lemma 5, $\overline{A_a^\delta}$ is standard wrt P_X restricted to $\overline{A_a^\delta}$. Invoking Lemma 4 with $Q = \overline{A_a^\delta}$

and recalling that $\mathbb{P}(X \in \partial A_a^\delta) = 0$ we get the covering property stated in H5 iii).

Lastly the uniform convergence stated in H7 follows from Theorem 6 in Abdous and Theodorescu (1989), since f is uniformly continuous, assumptions (i)-(iii) hold for the uniform kernel and the bandwidth fulfills $lh_l^{2d}/\log(l) \to \infty$.

References

Abdous, B. and Theodorescu, R. (1989) On the Strong Uniform Consistency of a New Kernel Density Estimator. *Metrika* 11: 177–194.

Aaron, C., Cholaquidis, A., and Cuevas, A. (2017). Stochastic detection of some topological and geometric features. *Electronic Journal of Statistics* 11(2): 4596–4628. `http://dx.doi.org/10.1214/17-EJS1370`

Agrawala, A.K. (1970). Learning with a probabilistic teacher. *IEEE Transactions on Automatic Control* 19: 716–723

Asuncion, A., and Newman, D.J. (2007) UCI Machine Learning Repository, `http://www.ics.uci.edu/~mlearn/MLRepository.html`, *University of California, Irvine, School of Information and Computer Sciences*

Azizyan, M., Singh, A., and Wasserman, L. (2013). Density-sensitive semisupervised inference. *The Annals of Statistics* 41(2): 751–771.

Biau, G., and Devroye L. (2015). Lectures on the Nearest Neighbor Method. *Springer-Verlag.*

Belkin, M., and Niyogi, P. (2004). Semi-supervised learning on Riemannian manifolds. *Machine Learning* 56: 209–239.

Ben-David, S., Lu, T. and Pal, D.(2008). Does unlabeled data provably help? Worst-case analysis of the sample complexity of semi-supervised learning. In 21*st Annual Conference on Learning Theory (COLT)*. Available at `http://www.informatik.uni-trier.de/~ley/db/conf/colt/colt2008.html`.

Chapelle, O., Schölkopf, B., and Zien, A., eds. (2006) Semi-Supervised Learning. MIT Press.

Cholaquidis, A., Cuevas, A. and Fraiman, R. (2014) On Poincaré cone property. *Ann. Statist.*, **42**, 255–284.

Cuevas, A., and Fraiman, R. (1997). A plug-in approach to support estimation. *Annals of Statistics* 25: 2300–2312.

Cuevas, A., and Rodríguez-Casal, A. (2004). On boundary estimation. *Adv. in Appl. Probab.* 36: 340–354.

Cuevas, A., Fraiman, R. and Pateiro-López, B. (2012). On statistical properties of sets fulfilling rolling-type conditions. *Adv. in Appl. Probab.* **44** 311–329.

Erdös, P. (1945). Some remarks on the measurability of certain sets. *Bull. Amer. Math. Soc.* **51** 728–731.

Fanty, M., and Cole, R. (1991) Spoken letter recognition. In: R.P. Lippman, J. Moody, and D.S. Touretzky (Eds.), *Advances in Neural Information Processing Systems*, 3. Morgan Kaufmann, San Mateo, CA.

Federer, H. (1959) Curvature measures. *Trans. Amer. Math. Soc.* 93: 418–491.

Fralick, S.C. (1967) Learning to recognize patterns without a teacher. *IEEE Transactions on Information Theory* 13: 57–64.

Haffari, G. and Sarkar, A. (2007) Analysis of Semi-Supervised Learning with the Yarkowsky algoritm. *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence, UAI 2007. Vancouver, BC. July 19–22, 2007.*

Lafferty, J., and Wasserman, L. (2008). Statistical analysis of semi-supervised regression. Conference in *Advances in Neural Information Processing Systems*. 801–808.

Nadler, B., Srebro, N., and Zhou, X. (2009). Statistical analysis of semi-supervised learning: The limit of infinite unlabeled data. In *Advances in Neural Information Processing Systems* Vol. 22. MIT Press, pp. 1330–1338.

Niyogi, P. (2008). Manifold regularization and semi-supervised learning: Some theoretical analyses. Technical Report TR-2008-01, Computer Science Dept., Univ. of Chicago. Available at `http://people.cs.uchicago.edu/~niyogi/papersps/ssminimax2.pdf`.

Rigollet, P. (2007). Generalized error bound in semi-supervised classification under the cluster assumption. *J. Mach. Learn. Res.* 8: 1369–1392. MR2332435.

Scudder, H. J. (1965) Probability of error of some adaptive patter-recognition machines. *IEEE Transactions on Information Theory* 11: 363–371.

Singh, A., Nowak, R. D., and Zhu, X.(2008). Unlabeled data: Now it helps, now it doesn't. Technical report, ECE Dept., Univ. Wisconsin-Madison. Available at `www.cs.cmu.edu/~aarti/pubs/SSL_TR.pdf`.

Sinha, K., and Belkin, M. (2009). Semi-supervised learning using sparse eigenfunction bases. In *Advances in Neural Information Processing Systems* Vol. 22. Y. Bengio, D. Schuurmans, J. Lafferty, C.K.I. Williams and A. Culotta, eds. MIT Press, pp. 1687–1695.

Zhu, X. (2008) Semi-Supervised learning literature survey. `http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html`