HAL open science

Learning from Noisy Data using Hyperplane Sampling and Sample Averages

Guillaume Stempfel, Liva Ralaivola, François Denis

To cite this version:

Guillaume Stempfel, Liva Ralaivola, François Denis. Learning from Noisy Data using Hyperplane Sampling and Sample Averages. 2007. hal-00186391

HAL Id: hal-00186391

https://hal.science/hal-00186391

Preprint submitted on 8 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning from Noisy Data using Hyperplane Sampling and Sample Averages

Guillaume Stempfel, Liva Ralaivola, François Denis

Laboratoire d'Informatique de Marseille, UMR CNRS 6166
Université de Provence, 39, rue Joliot Curie, F-13013 Marseille, France
firstname.name@lif.univ-mrs.fr

Abstract. We present a new classification algorithm capable of learning from data corrupted by a class dependent uniform *classification noise*. The produced classifier is a linear classifier, and the algorithm works seamlessly when using kernels. The algorithm relies on the sampling of *random hyperplanes* that help the building of new training examples of which the correct classes are known; a linear classifier (e.g. an SVM) is learned from these examples and output by the algorithm. The produced examples are *sample averages* computed from the data at hand with respect to areas of the space defined by the random hyperplanes and the target hyperplane. A statistical analysis of the properties of these sample averages is provided as well as results from numerical simulations conducted on synthetic datasets. These simulations show that the linear and kernelized versions of our algorithm are effective for learning from both noise-free and noisy data.

1 Introduction

Learning from noisy data is a problem of interest both from the practical and theoretical points of view. In this paper, we focus on a particular noise setting in a binary classification framework where the noise process uniformly flips the label of an example to the opposite label with a probability that depends on each class. An instance of this classification setting might be that of automatic spam filtering where a user might erroneously label regular mails as spam and conversely; it is obvious in this example that the probability of mislabelling an email depends on its true nature (spam or non spam).

From the theoretical and algorithmic point of views, there are very few simple learning strategies that are guaranteed to output a reliable classifier from data corrupted by the classification noise process depicted above. It must be noted that despite soft-margin Support Vector Machines seem to be a viable strategy to learn from noisy data and that generalization bounds for SVM expressed in terms of margin and the values of slack variables do exist, there is *no* result, to our knowledge, about the characteristics of the solution to the SVM quadratic program when noisy data are involved.

Here, we propose a strategy to learn a large margin classifier from noisy data. The algorithm proposed relies on the sampling of *random hyperplanes* that help the building of new training examples of which the correct classes are known; a linear classifier (e.g. a perceptron) is learned from these examples and output by the algorithm. The produced examples are *sample averages* computed from the data at hand with respect to areas of the space defined by the random hyperplanes and the target hyperplane.

The paper is organized as follows. Section 2 formalizes the problem and introduces notation. Section 3 presents the different parts of the algorithms, i.e. the computation of the sample averages and their use as inputs to an SVM classifier. Numerical simulations are presented in Section 4: they show the behavior of our algorithm on linearly separable distributions and nonlinearly separable distributions in the noise-free and noisy contexts.

2 Notation and Definition of the Problem

$\mathcal{X}$ denotes the input space, assumed to be an *Hilbert space*, equipped with an inner product denoted by $\cdot$. We restrict our study to the binary classification problem and the target space $\mathcal{Y}$ is $\mathcal{Y}\{-1,+1\}$. Throughout the analysis, we additionally make the assumption of the existence of zero bias separating hyperplanes (i.e hyperplanes defined as $\mathbf{w}\cdot\mathbf{x}=0$ that pass through the origin of the space). These assumptions make our analysis seemlessly applicable when using kernels.

In order to simplify the definition and the writing of the proofs, we will consider normalized labeled examples, that is, for a pair $(\mathbf{x},y)$, we will consider $(\frac{\mathbf{x}}{\|\mathbf{x}\|},y)$. Note that the transformation does not change the difficulty of the problem.

Definition 1 (γ-separable distributions). *For $\gamma>0$, $\mathcal{D}^\gamma$ is the set of distributions on $\mathcal{X}$ such that for any $D\in\mathcal{D}^\gamma$, there exists a unit vector $\mathbf{w}^*\in\mathcal{X}$ such that*

$$\mathbb{P}_{(\mathbf{x},y)\sim D}\left[y(\mathbf{w}^*\cdot\mathbf{x})<\gamma\right]=0.$$

This means that given a vector $\mathbf{w}^*$, we consider a deterministic labelling $y(\mathbf{x})$ of $\mathbf{x}$ according to the sign of $\mathbf{w}^*\cdot\mathbf{x}$, i.e., $y(\mathbf{x})=$ sign $(\mathbf{w}^*\cdot\mathbf{x})$.

For a distribution D defined over the labelled space $\mathcal{X}\times\mathcal{Y}$, $D_{\mathcal{X}}$ denotes the distribution marginalized over $\mathcal{Y}$.

Definition 2 (Class-conditional classification noise process y^η). *Let $\eta^+,\eta^-\in[0,1)$ such that $\eta^++\eta^-<1$ and define $\boldsymbol{\eta}=[\eta^+\ \eta^-]$. Let the random process $y^{\boldsymbol{\eta}}:\mathcal{Y}\to\mathcal{Y}$ maps y as follows:*

$$y^{\boldsymbol{\eta}}(y)=\begin{cases}+1 \text{ with prob. } 1-\eta^+ & \text{if } y=+1\\ -1 \text{ with prob. } \quad\eta^+ & \text{if } y=+1\\ +1 \text{ with prob. } \quad\eta^- & \text{if } y=-1\\ -1 \text{ with prob. } 1-\eta^- & \text{if } y=-1\end{cases}$$

Let $\gamma>0$ and $\eta^+,\eta^-\in[0,1)$. For a distribution $D\in\mathcal{D}^\gamma$, D^η is the distribution over $\mathcal{X}\times\mathcal{Y}$ from which a labelled example is drawn according to the following process: (a) draw $(\mathbf{x},y)$ according to D and (b) return $(\mathbf{x},y^{\boldsymbol{\eta}}(y))$. The noisy version $\mathcal{S}^{\boldsymbol{\eta}}$ of a random set $\mathcal{S}=\{(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_m,y_m)\}$ is defined likewise, i.e. $\mathcal{S}^{\boldsymbol{\eta}}=\{(\mathbf{x}_1,y^{\boldsymbol{\eta}}(y_1)),\ldots,(\mathbf{x}_m,y^{\boldsymbol{\eta}}(y_m)\}$, which, from here on, will be shorthanded as $\mathcal{S}=\{(\mathbf{x}_1,y_1^{\boldsymbol{\eta}}),\ldots,(\mathbf{x}_m,y_m^{\boldsymbol{\eta}})\}$.

With the previous definitions at hand, the problem that we tackle is the learning from samples drawn from $D^{\boldsymbol{\eta}}$, for a noise-free distribution $D\in\mathcal{D}^\gamma$. We exhibit a learning algorithm that, given a finite sample from $D^{\boldsymbol{\eta}}$ (altered by class-conditional classification noise), outputs a linear classifier $\mathbf{w}$ that is targetted to a low error rate

on the noise free distribution D. If learning algorithms exist to learn from distributions that undergo a uniform classification noise process (with the same noise rate on both classes), as, for instance the perceptron-like approaches described in [1–3], or coreset-based strategies [4], they have not been actually specialized to the handling of class dependent classification noise. According to [5], these algorithms could be extended in a straightforward way to this setting, but we do think that it is of major importance to have a dedicated learning algorithm for this problem. In addition, we truly think that the strategy presented in this paper is amenable to a full theoretical analysis showing it is a Probably Approximately Correct strategy. This is the topic of future researches.

3 The Sample Average Machine

3.1 High Level Description

The new algorithm that we propose, called *Sample Average Machine* (SAM), implements a very simple yet effective two-step learning strategy. Given a noisy training sample $\mathcal{S}^{\boldsymbol{\eta}} = \{(\mathbf{x}_1, y_1^{\boldsymbol{\eta}}), \ldots, (\mathbf{x}_m, y_m^{\boldsymbol{\eta}})\}$ drawn according to $D^{\boldsymbol{\eta}}$ (the noisy version of D) the algorithm, depicted in Algorithm 1, works as follows.

1. SAM creates a new sample $\mathcal{S}^{\boldsymbol{\mu}}$ of examples of which the correct labels, according to the target hyperplane $\mathbf{w}^*$, are known with high probability. To produce these examples, SAM uniformly picks random vectors on the unit hypersphere defined in the subspace spanned by the vectors of the training set (this space is therefore of dimension at most m). For each random hyperplane $\mathbf{w}$, at least one new point $\boldsymbol{\mu}$ is added to $\mathcal{S}^{\boldsymbol{\mu}}$: $\boldsymbol{\mu}$ is a sample estimate of the mean vector of $D_{\mathcal{X}}$ restricted to one of the four subspaces delimited by $\mathbf{w}$ and $\mathbf{w}^*$ (see Fig. 1). Provided m is large enough, the correct label of $\boldsymbol{\mu}$ is known with high probability.
2. SAM learns a support vector classifier from $\mathcal{S}^{\boldsymbol{\mu}}$. The hyperplane returned by SAM is the one that will be used to classify new data.

The remaining of this section is largely devoted to the computation of the new examples the SVM is built upon. We give their definitions and provide their statistical properties. The proof heavily relies on concentration results, which makes it possible to have dimension independent results.

3.2 Generation of the Training Sample $\mathcal{S}^{\boldsymbol{\mu}}$

In the following, we consider a linearly separable distribution (with margin $\gamma > 0$) $D \in \mathcal{D}^{\gamma}$, of target hyperplane $\mathbf{w}^*$. The noise vector associated to the noise process is $\boldsymbol{\eta} = [\eta^+ \ \eta^-]$. Let $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$ a sample of m labelled examples drawn from D and $\mathcal{S}^{\boldsymbol{\eta}} = \{(\mathbf{x}_1, y_1^{\boldsymbol{\eta}}), \ldots, (\mathbf{x}_m, y_m^{\boldsymbol{\eta}})\}$ is the noisy version of S.

The first step to generate the new training points is the sampling of a vector $\mathbf{w}$ uniformly picked on the unit hypersphere defined in the space spanned by the $\mathbf{x}_1, \ldots, \mathbf{x}_m$. Given $\mathbf{w}$ and $\mathbf{w}^*$, $\mathcal{X}$ can be decomposed into four disjoint subspaces (as illustrated on Figure 1):

$$\begin{aligned} A(\mathbf{w}) &= \{\mathbf{x} \in \mathcal{X} | \mathbf{w} \cdot \mathbf{x} \geq 0, y(\mathbf{x}) = +1\} \\ B(\mathbf{w}) &= \{\mathbf{x} \in \mathcal{X} | \mathbf{w} \cdot \mathbf{x} < 0, y(\mathbf{x}) = +1\} \\ E(\mathbf{w}) &= \{\mathbf{x} \in \mathcal{X} | \mathbf{w} \cdot \mathbf{x} < 0, y(\mathbf{x}) = -1\} \\ F(\mathbf{w}) &= \{\mathbf{x} \in \mathcal{X} | \mathbf{w} \cdot \mathbf{x} \geq 0, y(\mathbf{x}) = -1\}\,. \end{aligned} \tag{1}$$

Algorithm 1 Sample Average Machine (see text for details)

Input: $m_0, n, \boldsymbol{\eta} = [\eta^+ \ \eta^-], \mathcal{S}^{\boldsymbol{\eta}} = \{(\mathbf{x}_1, y_1^{\boldsymbol{\eta}}), \ldots, (\mathbf{x}_m, y_m^{\boldsymbol{\eta}})\}$
Output: a linear classifier $\mathbf{w}$

```
/* building of the new (noise free) training sample */
S^μ = ∅
while |S^μ| < n do
    draw a random unit vector w ∈ span(x_1, ..., x_m)
    y_{A(w)} = y_{B(w)} = +1,  y_{E(w)} = y_{F(w)} = −1
    for Z ∈ {A(w), B(w), E(w), F(w)} do
        compute μ̂^η(Z) according to equation (3)
        compute P̂^η(Z) according to equation (4)
        if P̂^η(Z) ≥ m_0/m then
            S^μ = S^μ ∪ {(μ̂^η(Z), y_Z)}
        end if
    end for
    /* SVM learning */
    learn an SVM classifier w_SVM on S^μ
    return w_SVM
end while
```

Likewise, the samples $\mathcal{S}$ and $\mathcal{S}^{\boldsymbol{\eta}}$ are divided into four subsets each:

$$\begin{aligned}
\hat{A}^{(\boldsymbol{\eta})}(\mathbf{w}) &= \left\{\mathbf{x}_i \in \mathcal{S}^{(\boldsymbol{\eta})} | \mathbf{w} \cdot \mathbf{x}_i \geq 0, y_i^{(\boldsymbol{\eta})} = +1\right\} \\
\hat{B}^{(\boldsymbol{\eta})}(\mathbf{w}) &= \left\{\mathbf{x}_i \in \mathcal{S}^{(\boldsymbol{\eta})} | \mathbf{w} \cdot \mathbf{x}_i < 0, y_i^{(\boldsymbol{\eta})} = +1\right\} \\
\hat{E}^{(\boldsymbol{\eta})}(\mathbf{w}) &= \left\{\mathbf{x}_i \in \mathcal{S}^{(\boldsymbol{\eta})} | \mathbf{w} \cdot \mathbf{x}_i < 0, y_i^{(\boldsymbol{\eta})} = -1\right\} \\
\hat{F}^{(\boldsymbol{\eta})}(\mathbf{w}) &= \left\{\mathbf{x}_i \in \mathcal{S}^{(\boldsymbol{\eta})} | \mathbf{w} \cdot \mathbf{x}_i \geq 0, y_i^{(\boldsymbol{\eta})} = -1\right\}
\end{aligned} \tag{2}$$

where the superscript $(\boldsymbol{\eta})$ denotes an optional $\boldsymbol{\eta}$ in the definitions.

A few observations can be readily made: (a) the mean vectors of D restricted to each of the subspaces $A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})$ are of class $+1$, $+1$, -1, -1, respectively; (b) those mean vectors could be estimated by sample averages computed on the subsets $\hat{A}(\mathbf{w}), \hat{B}(\mathbf{w}), \hat{E}(\mathbf{w}), \hat{F}(\mathbf{w})$ if these sets were known; (c) in the framework of classification from noisy dataset however, the only set that is accessible is $\mathcal{S}^{\boldsymbol{\eta}}$ and its corresponding subsets and these sample averages cannot be performed directly.

It nevertheless turns out that it is possible to derive unbiased estimates of the mean vectors of each subspace from $\mathcal{S}^{\boldsymbol{\eta}}$. For a given $\mathbf{w}$, it is therefore possible to compute (at most four) vectors of which the correct classes are known with high probability, provided the size of $\mathcal{S}^{\boldsymbol{\eta}}$ is large enough.

Sample Estimates of the Mean Vectors. The four vectors of interest that we would like to approximate (respectively of class $+1$, $+1$, -1, and -1) are defined for Z in $\{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\}$ by $\boldsymbol{\mu}(Z) = \mathbb{E}_{\mathbf{x}\sim D}\left[\mathbf{x}\mathbb{1}_Z(\mathbf{x})\right]$ where $\mathbb{1}_Z(\mathbf{z}) = 1$ if $\mathbf{z} \in Z$ and 0 otherwise; we denote by $\hat{\boldsymbol{\mu}}(Z) = \frac{1}{m}\sum_{\mathbf{x}_i \in \hat{Z}} \mathbf{x}_i$ the sample estimate of $\boldsymbol{\mu}_Z$ (again, these sample estimates cannot be computed directly).

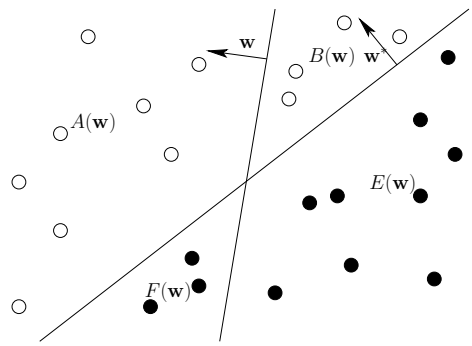

Fig. 1. The subspaces $A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})$ defined by a random vector $\mathbf{w}$ and the target vector $\mathbf{w}^*$; the corresponding subsets $\hat{A}(\mathbf{w}), \hat{B}(\mathbf{w}), \hat{E}(\mathbf{w}), \hat{F}(\mathbf{w})$ of a noise free set $\mathcal{S}$ (white/black disks are positive/negative examples) are automatically deduced.

Let $\beta = \frac{1}{1-\eta^+-\eta^-}$. Consider the following vectors, which can be computed from $\mathcal{S}^{\boldsymbol{\eta}}$:

$$\begin{aligned}
\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(A(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^-)\textstyle\sum_{\mathbf{x}_i\in\hat{A}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i - \eta^-\textstyle\sum_{\mathbf{x}_i\in\hat{F}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i\right)\\
\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(B(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^-)\textstyle\sum_{\mathbf{x}_i\in\hat{B}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i - \eta^-\textstyle\sum_{\mathbf{x}_i\in\hat{E}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i\right)\\
\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(E(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^+)\textstyle\sum_{\mathbf{x}_i\in\hat{E}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i - \eta^+\textstyle\sum_{\mathbf{x}_i\in\hat{B}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i\right)\\
\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(F(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^+)\textstyle\sum_{\mathbf{x}_i\in\hat{F}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i - \eta^+\textstyle\sum_{\mathbf{x}_i\in\hat{A}^{\boldsymbol{\eta}}(\mathbf{w})}\mathbf{x}_i\right).
\end{aligned} \tag{3}$$

Then, we have the following lemma.

Lemma 1. *$\forall D \in \mathcal{D}^0$, $\forall \mathbf{w} \in \mathcal{X}$, $\forall \eta^+, \eta^- \in [0,1)$ such that $\eta^+ + \eta^- < 1$, $\forall m \in \mathbb{N}$, $\forall \mathcal{S}^{\boldsymbol{\eta}} = \{(\mathbf{x}_1, y_1^{\boldsymbol{\eta}}), \ldots, (\mathbf{x}_m, y_m^{\boldsymbol{\eta}})\}$ drawn from $D^{\boldsymbol{\eta}}$, for Z in $\{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\}$ the following holds:*

$$\mathbb{E}_{\mathcal{S}^{\boldsymbol{\eta}}\sim D^{\boldsymbol{\eta}}}(\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z)) = \boldsymbol{\mu}(Z).$$

Proof. Let us focus on the case $Z = A(\mathbf{w})$ (the proof works similarly for the other three subspaces). In order to work out the proof of the equality, it suffices to observe that the random variable $\mathcal{S}^{\boldsymbol{\eta}}$ is driven by two random process. The first one is that of drawing random vectors $\mathbf{x}_1, \ldots, \mathbf{x}_m$ from $D_{\mathcal{X}}$ while the second is the class conditional random noise process (see Definition 2), which depends upon the first one through the labels of the $\mathbf{x}_i$'s. We therefore have:

$$\mathbb{E}_{\mathcal{S}^{\boldsymbol{\eta}}\sim D^{\boldsymbol{\eta}}}(\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(A(\mathbf{w}))) = \mathbb{E}_{\mathbf{x}_1,\ldots\mathbf{x}_m\sim D_{\mathcal{X}}}\left[\mathbb{E}_{y^{\boldsymbol{\eta}}(y_1),\ldots,y^{\boldsymbol{\eta}}(y_m)}\left[\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(A(\mathbf{w}))|\mathbf{x}_1,\ldots,\mathbf{x}_m\right]\right].$$

Focusing on the innermost expectation, droping , for sake of space, the fact that the $\mathbf{x}_1, \ldots, \mathbf{x}_m$ are fixed, and using the shorthand $\mathbf{y}^{\boldsymbol{\eta}}$ for $y^{\boldsymbol{\eta}}(y_1) \ldots y^{\boldsymbol{\eta}}(y_m)$, we have:

$$
\begin{aligned}
\mathbb{E}_{\mathbf{y}^{\boldsymbol{\eta}}}\left[\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(A(\mathbf{w}))\right] &= \mathbb{E}_{\mathbf{y}^{\boldsymbol{\eta}}}\left[\frac{1}{m}\beta\left((1-\eta^-)\sum_{\mathbf{x}_i \in \hat{A}^{\boldsymbol{\eta}}(\mathbf{w})} \mathbf{x}_i - \eta^- \sum_{\mathbf{x}_i \in \hat{F}^{\boldsymbol{\eta}}(\mathbf{w})} \mathbf{x}_i\right)\right] \\
&= \frac{\beta}{m}\mathbb{E}_{\mathbf{y}^{\boldsymbol{\eta}}}\left[(1-\eta^-)\sum_{\mathbf{w}\cdot\mathbf{x}_i \geq 0} \mathbf{x}_i \mathbf{1}_{\{+1\}}(y^{\boldsymbol{\eta}}(y_i)) - \eta^- \sum_{\mathbf{w}\cdot\mathbf{x}_i \geq 0} \mathbf{x}_i \mathbf{1}_{\{-1\}}(y^{\boldsymbol{\eta}}(y_i))\right] \\
&= \frac{\beta}{m}\mathbb{E}_{\mathbf{y}^{\boldsymbol{\eta}}}\left[(1-\eta^-)\left(\sum_{\mathbf{x}_i \in \hat{A}(\mathbf{w})} \mathbf{x}_i \mathbf{1}_{\{+1\}}(y^{\boldsymbol{\eta}}(+1)) + \sum_{\mathbf{x}_i \in \hat{F}(\mathbf{w})} \mathbf{x}_i \mathbf{1}_{\{+1\}}(y^{\boldsymbol{\eta}}(-1))\right)\right. \\
&\qquad \left. -\eta^-\left(\sum_{\mathbf{x}_i \in \hat{A}(\mathbf{w})} \mathbf{x}_i \mathbf{1}_{\{-1\}}(y^{\boldsymbol{\eta}}(+1)) + \sum_{\mathbf{x}_i \in \hat{F}(\mathbf{w})} \mathbf{x}_i \mathbf{1}_{\{-1\}}(y^{\boldsymbol{\eta}}(-1))\right)\right] \\
&= \frac{1}{m}\beta\left((1-\eta^-)\left(\sum_{\mathbf{x}_i \in \hat{A}(\mathbf{w})} \mathbf{x}_i(1-\eta^+) + \sum_{\mathbf{x}_i \in \hat{F}(\mathbf{w})} \mathbf{x}_i(\eta^-)\right)\right. \\
&\qquad \left. -\eta^-\left(\sum_{\mathbf{x}_i \in \hat{A}(\mathbf{w})} \mathbf{x}_i(\eta^+) + \sum_{\mathbf{x}_i \in \hat{F}(\mathbf{w})} \mathbf{x}_i(1-\eta^-)\right)\right) \\
&= \frac{1}{m}\sum_{\mathbf{x}_i \in \hat{A}(\mathbf{w})} \mathbf{x}_i \\
&= \hat{\boldsymbol{\mu}}(A(\mathbf{w})).
\end{aligned}
$$

Getting back to the full expectation, it is straightforward to check that:

$$
\begin{aligned}
\mathbb{E}_{\mathcal{S}^{\boldsymbol{\eta}} \sim D^{\boldsymbol{\eta}}}(\boldsymbol{\mu}^{\boldsymbol{\eta}}(A(\mathbf{w}))) &= \mathbb{E}_{\mathbf{x}_1,\ldots\mathbf{x}_m \sim D_{\mathcal{X}}}\left[\mathbb{E}_{y^{\boldsymbol{\eta}}(y_1),\ldots,y^{\boldsymbol{\eta}}(y_m)}\left[\boldsymbol{\mu}^{\eta}(A(\mathbf{w}))|\mathbf{x}_1,\ldots,\mathbf{x}_m\right]\right] \\
&= \mathbb{E}_{\mathbf{x}_1,\ldots\mathbf{x}_m \sim D_{\mathcal{X}}}\left[\hat{\boldsymbol{\mu}}(A(\mathbf{w}))\right] \\
&= \boldsymbol{\mu}_{A(\mathbf{w})}.
\end{aligned}
$$

We therefore have proved that, given a vector $\mathbf{w}$, the points computed in (3) are unbiased estimates of the mean of the subspaces delimited by $\mathbf{w}$ and $\mathbf{w}^*$.

Concentration of the Sample Estimates. We show that the distances between the sample estimates and their expected values can be bounded with high probability. To this end, we make use of McDiarmid's inequality:

Theorem 1 ([6]). *Let $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be independent random variables taking values in a set $\mathcal{X}$, and assume that $f : \mathcal{X}^n \to \mathbb{R}$ satisfies*

$$\sup_{\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{x}_i' \in \mathcal{X}} \left|f(\mathbf{x}_1,\ldots,\mathbf{x}_n) - f(\mathbf{x}_1,\ldots,\mathbf{x}_{i-1},\mathbf{x}_i',\mathbf{x}_{i+1},\ldots,\mathbf{x}_n)\right| \leq c_i$$

for every $1 \leq i \leq n$. Then, for every $t > 0$,

$$\mathbb{P}\left\{|f(\mathbf{x}_1,\ldots,\mathbf{x}_n) - \mathbf{E}f(\mathbf{x}_1,\ldots,\mathbf{x}_n)| \geq t\right\} \leq 2\exp\left(-\frac{2t^2}{\sum_{i=1}^n c_i^2}\right).$$

We have the following lemma.

Lemma 2. *$\forall\gamma > 0$, $\forall\delta > 0$, $\forall D \in \mathcal{D}^\gamma$, $\forall \mathbf{w} \in \mathcal{X}, \forall \mathcal{S}$ sample drawn from D,* if $|S| > \frac{128^2 \ln\left(\frac{8}{\delta}\right)}{\gamma^2}$ then, *for $Z \in \{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\}$,*

$$\mathbb{P}\left\{\|\boldsymbol{\mu}(Z) - \hat{\boldsymbol{\mu}}(Z)\| > \frac{\gamma}{32}\right\} < \frac{\delta}{4}.$$

Proof. Again let $Z = A(\mathbf{w})$. Consider $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_m, y_m)\}$ and $\mathcal{S}^{(i)} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}'_i, y'_i), \ldots, (\mathbf{x}_m, y_m)\}$ two samples of m examples drawn from D that only differs on their ith variable. It can be observed that for any i in $1, \ldots, m$:

$$\left|\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\| - \left\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}^{(i)}(A(\mathbf{w}))\right\|\right| \leq \left\|\hat{\boldsymbol{\mu}}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}^{(i)}(A(\mathbf{w}))\right\| \leq \frac{2}{m}$$

Hence, by Theorem 1, Lemma 1 and according to the value of m, we have

$$\begin{aligned}\mathbb{P}\left\{\left|\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\| - \mathbb{E}_{S\sim D}\left[\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\|\right]\right| > \frac{\gamma}{64}\right\} &< 2\exp\left(-\frac{\gamma^2 m}{8192}\right)\\ &< \frac{\delta}{4}.\end{aligned}$$

If $\boldsymbol{\sigma}$ is a Rademacher vector of size m, i.e. a random vector whose entries independently take the values $+1$ or -1 with probability 0.5, we observe that:

$$\begin{aligned}
\mathbb{E}_{S\sim D}\left[\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\|\right] &= \mathbb{E}_S\left[\left\|\mathbb{E}_{S'}\left[\hat{\boldsymbol{\mu}}'(A(\mathbf{w}))\right] - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\right\|\right]\\
&\leq \mathbb{E}_{SS'}\left[\left\|\hat{\boldsymbol{\mu}}'(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\right\|\right] && \text{(triangle ineq.)}\\
&= \mathbb{E}_{SS'}\left[\left\|\frac{1}{m}\sum\nolimits_{\mathbf{x}'_i \in \hat{A}'(\mathbf{w}))}\mathbf{x}'_i - \frac{1}{m}\sum\nolimits_{\mathbf{x}_i \in \hat{A}(\mathbf{w})}\mathbf{x}_i\right\|\right]\\
&= \mathbb{E}_{SS'\boldsymbol{\sigma}}\left[\frac{1}{m}\left\|\sum\nolimits_{i=1}^m \sigma_i\left(\mathbf{x}'_i\mathbf{1}_{\hat{A}'(\mathbf{w})}(\mathbf{x}'_i) - \mathbf{x}_i\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_i)\right)\right\|\right]\\
& && \text{($\mathcal{S}$ and $\mathcal{S}'$ are iid samples)}\\
&\leq 2\mathbb{E}_{S\boldsymbol{\sigma}}\left[\frac{1}{m}\left\|\sum\nolimits_{i=1}^m \sigma_i\mathbf{x}_i\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_i)\right\|\right]\\
&= \frac{2}{m}\mathbb{E}_{S\boldsymbol{\sigma}}\left[\left(\left(\sum\nolimits_{i=1}^m \sigma_i\mathbf{x}_i\mathbf{1}_{\hat{A}'(\mathbf{w})}(\mathbf{x}_i)\right)\cdot\left(\sum\nolimits_{j=1}^m \sigma_j\mathbf{x}_j\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_j)\right)\right)^{\frac{1}{2}}\right]\\
&\leq \frac{2}{m}\left(\mathbb{E}_{S\sigma}\left[\sum\nolimits_{i,j=1}^m \sigma_i\sigma_j\mathbf{x}_i\cdot\mathbf{x}_j\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_i)\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_j)\right]\right)^{\frac{1}{2}}\\
& && \text{(Jensen ineq.)}\\
&= \frac{2}{m}\left(\mathbb{E}_S\left[\sum\nolimits_{i=1}^m \mathbf{x}_i\cdot\mathbf{x}_i\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_i)\mathbf{1}_{\hat{A}(\mathbf{w})}(\mathbf{x}_i)\right]\right)^{\frac{1}{2}} && (\mathbb{E}_{\boldsymbol{\sigma}}[\sigma_i\sigma_j] = \delta_{ij})\\
&= \frac{2}{m}\sqrt{mD(A(\mathbf{w}))}\\
&\leq \frac{2}{\sqrt{m}}
\end{aligned}$$

According to the lower bound on m, we can conclude that

$$\mathbb{P}\left\{\|\boldsymbol{\mu}(A(\mathbf{w})) - \hat{\boldsymbol{\mu}}(A(\mathbf{w}))\| > \frac{\gamma}{32}\right\} < \frac{\delta}{4}.$$

The following lemma is very similar to the previous one except that the random process considered is the classification noise process.

Lemma 3. *Let* $\gamma > 0$. $\forall \eta^+, \eta^- \in [0,1)$ *such that* $\eta^+ + \eta^- < 1$, $\forall \delta > 0$, $\forall D \in \mathcal{D}^\gamma$, $\forall \mathbf{w} \in \mathcal{X}$, $\forall S^\eta$ *sample drawn from* $D^{\boldsymbol{\eta}}$, if $|S^{\boldsymbol{\eta}}| > \frac{128^2 \ln\left(\frac{8}{\delta}\right)}{\gamma^2(1-\eta^+-\eta^-)}$ then *for* $Z \in \{A(\mathbf{w}), B(\mathbf{w}), C(\mathbf{w}), D(\mathbf{w})\}$, *we have:*

$$\mathbb{P}\left\{\|\hat{\boldsymbol{\mu}}(Z) - \hat{\mu}^{\boldsymbol{\eta}}(Z)\| > \frac{\gamma}{32}\right\} < \frac{\delta}{4}$$

Proof. The lines of the proof are exactly the same as for Lemma 2 (even though the complete proof is little bit more tedious because of the way $\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z)$ is defined) and we leave it to the reader.

Finally, the next proposition readily follows.

Proposition 1. *Let* $\gamma > 0$. $\forall \eta^+, \eta^- \in [0,1)$ *such that* $\eta^+ + \eta^- < 1$, $\forall \delta > 0$, $\forall n \in \mathbb{N}$, $\forall D \in \mathcal{D}^\gamma$, $\forall W = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\}, \mathbf{w}_i \in \mathcal{X}$ *and* $\forall S^{\boldsymbol{\eta}}$ *drawn from* $D^{\boldsymbol{\eta}}$, if $|S^{\boldsymbol{\eta}}| > \frac{128^2 \ln\left(\frac{8n}{\delta}\right)}{\gamma^2(1-\eta^+-\eta^-)}$ then *the following holds:*

$$\forall Z \in \{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\},\ \mathbb{P}\left\{\|\boldsymbol{\mu}(Z) - \hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z)\| > \frac{\gamma}{16}\right\} \leq \frac{\delta}{2},\ \forall \mathbf{w} \in W.$$

Proof. Suppose that $Z = A(\mathbf{w})$. By Lemma 2 and Lemma 3, we know that for a given $\mathbf{w} \in \mathcal{X}$:

$$\mathbb{P}\left\{\mathbf{w} : \|\hat{\boldsymbol{\mu}}^{\eta}(A(\mathbf{w})) - \boldsymbol{\mu}(A(\mathbf{w}))\| > \frac{\gamma}{16}\right\} \leq \frac{\delta}{2n}.$$

A simple union bound argument concludes the proof:

$$\begin{aligned}\mathbb{P}\left\{\exists \mathbf{w} \in W : \left\|\hat{\mu}(S)_{A^\eta(\mathbf{w}_i)} - \mu_{A(\mathbf{w}_i)}\right\| > \frac{\gamma}{16}\right\} &\leq |W| \mathbb{P}\left\{\mathbf{w} : \|\hat{\boldsymbol{\mu}}^{\eta}(A(\mathbf{w})) - \boldsymbol{\mu}(A(\mathbf{w}))\| > \frac{\gamma}{16}\right\} \\ &\leq n \cdot \frac{\delta}{2n} = \frac{\delta}{2}.\end{aligned}$$

3.3 Margins for the Sample Estimates

In this subsection we show that it is possible to guarantee the (normalized) margins of the newly generated points (computed according to (3)), and therefore the correct labels, with a simple criterion based on the estimated sizes of the $\hat{Z}(\mathbf{w})$ (for $Z \in \{A(\mathbf{w}), B(\mathbf{w}), C(\mathbf{w}), D(\mathbf{w})\}$).

Proceeding in a way very similar to what we did previously for the estimation of the mean vectors, we introduce the following notation: for $Z \in \{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\}$, $\mathbb{P}(Z) = \mathbb{E}_{\mathbf{x} \sim D_{\mathcal{X}}} \mathbf{1}_{\mathbf{x} \in Z}$ and $\hat{\mathbb{P}}(Z(\mathbf{w})) = \frac{1}{m} \sum_{x_i \in \hat{Z}(\mathbf{w})} 1 = \frac{|Z(\mathbf{w})|}{m}$, which is the sample estimate of $\mathbb{P}(Z(\mathbf{w}))$.

We define the 'probability estimates' counterparts of the mean vector estimates defined in (3):

$$\begin{aligned}
\hat{\mathbb{P}}^{\boldsymbol{\eta}}(A(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^-)|\hat{A}^\eta(\mathbf{w})| - \eta^-|\hat{F}^\eta(\mathbf{w})|\right) \\
\hat{\mathbb{P}}^{\boldsymbol{\eta}}(B(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^-)|\hat{B}^\eta(\mathbf{w})| - \eta^-|\hat{E}^\eta(\mathbf{w})|\right) \\
\hat{\mathbb{P}}^{\boldsymbol{\eta}}(E(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^+)|\hat{E}^\eta(\mathbf{w})| - \eta^+|\hat{B}^\eta(\mathbf{w})|\right) \\
\hat{\mathbb{P}}^{\boldsymbol{\eta}}(F(\mathbf{w})) &= \tfrac{1}{m}\beta\left((1-\eta^+)|\hat{F}^\eta(\mathbf{w})| - \eta^+|\hat{A}^\eta(\mathbf{w})|\right)
\end{aligned} \tag{4}$$

With these definitions at hand it is easy to get the following proposition.

Proposition 2. *Let $\gamma > 0$. $\forall \eta^+, \eta^- \in [0,1)$ such that $\eta^+ + \eta^- < 1$, $\forall \delta > 0$, $\forall n \in \mathbb{N}$, $\forall D \in \mathcal{D}^\gamma$, $\forall W = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\}, \mathbf{w}_i \in \mathcal{X}$ and $\forall \mathcal{S}^{\boldsymbol{\eta}}$ drawn from $D^{\boldsymbol{\eta}}$,* if $|\mathcal{S}^{\boldsymbol{\eta}}| > \frac{128^2 \ln\left(\frac{8n}{\delta}\right)}{\gamma^2(1-\eta^+-\eta^-)}$ then *the following holds:*

$$\forall Z \in \{A(\mathbf{w}), B(\mathbf{w}), C(\mathbf{w}), D(\mathbf{w})\},\ \mathbb{P}\left\{\left|\mathbb{P}(A(\mathbf{w})) - \hat{\mathbb{P}}(A(\mathbf{w}))\right| > \frac{1}{16}\right\} \leq \frac{\delta}{2},\ \forall \mathbf{w} \in W.$$

Proof. The steps of the proof are *exactly* the same as those leading to Proposition 1. It suffices to observe that, for instance, $|\hat{A}^\eta(\mathbf{w})| = \sum_{\mathbf{x}_i \in \hat{A}^\eta} 1$ is used here instead of $\sum_{\mathbf{x}_i \in \hat{A}^\eta} \mathbf{x}_i$ (see (3)). The proof is left to the reader.

We finally have the following result, which shows that it is always possible with high probability to keep among the four points calculated in (3), at least one that has a positive margin with respect to the target hyperplane $\mathbf{w}^*$.

Theorem 2. *Let $\gamma > 0$. $\forall \eta^+, \eta^- \in [0,1)$ such that $\eta^+ - \eta^- < 1$, $\forall \delta > 0$, $\forall n \in \mathbb{N}$, $\forall D \in D^\gamma$, $\forall W = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\}, \mathbf{w}_i \in \mathcal{X}, \|\mathbf{w}_i\| = 1$, $\forall \mathcal{S}^{\boldsymbol{\eta}}$ drawn from $D^{\boldsymbol{\eta}}$,* if $|S^\eta| > \frac{128^2 \ln\left(\frac{8n}{\delta}\right)}{\gamma^2(1-\eta^+-\eta^-)}$ then *with probability at least $1-\delta$ we can generate at least n points having margin $\frac{\gamma}{17}$.*

Proof. Given a target hyperplane $\mathbf{w}^*$, any unit vector $\mathbf{w}$ defines 4 subspaces, and there consequently is at least, one of these subspaces $Z_0 \in \{A(\mathbf{w}), B(\mathbf{w}), E(\mathbf{w}), F(\mathbf{w})\}$ such that $\mathbb{P}(Z_0) \geq \frac{1}{4}$. By Proposition 2

$$\mathbb{P}\left\{\left|\mathbb{P}(Z_0) - \hat{\mathbb{P}}^{\boldsymbol{\eta}}(Z_0)\right| > \frac{1}{16}\right\} \leq \frac{\delta}{2},\ \forall \mathbf{w} \in W,$$

and the computed estimate $\hat{\mathbb{P}}^\eta(Z)$ of the probability for this area is at least $\frac{3}{16}$. By Proposition 2 again, a subspace with $\hat{\mathbb{P}}^\eta(Z_0) \geq \frac{3}{16}$ is such that $\mathbb{P}(Z_0) \geq \frac{1}{8}$.

Let y_0 the label of the points in Z_0 (y_0 is known for each of the subspace $A(\mathbf{w})$, $B(\mathbf{w})$, $E(\mathbf{w})$, $F(\mathbf{w})$). As for all $\mathbf{x} \in Z_0$, $y_0\,(\mathbf{w}^* \cdot \mathbf{x}) \geq \gamma$, we have $y_0 \mathbf{w}^* \cdot \boldsymbol{\mu}(Z_0) \geq \frac{\gamma}{8}$ and then, by Proposition 1, we can lower bound the normalized margin of $\hat{\mu}^{\boldsymbol{\eta}}(Z_0)$ with respect to $\mathbf{w}*$ for a given $\mathbf{w}$ (recall that Z_0 depends on $\mathbf{w}$):

$$\begin{aligned}
\frac{\mathbf{w}^* \cdot \hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z_0)}{\|\mathbf{w}^*\|\,\|\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z_0)\|} &\geq \frac{\mathbf{w}^* \cdot \boldsymbol{\mu}(Z_0) - \|\mathbf{w}^*\|\,\|\boldsymbol{\mu}(Z_0) - \hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z_0)\|}{\|\mathbf{w}^*\|\,(\|\boldsymbol{\mu}(Z_0)\| + \|\boldsymbol{\mu}(Z) - \hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z_0)\|)} \\
&\geq \frac{\frac{\gamma}{8} - \frac{\gamma}{16}}{1 + \frac{\gamma}{16}} \geq \frac{\gamma}{17} \qquad \text{(with probability } 1 - \tfrac{\delta}{n})
\end{aligned}$$

Henceforth, the margin of the newly computed point $\hat{\boldsymbol{\mu}}^{\boldsymbol{\eta}}(Z_0)$ is at least $\frac{\gamma}{17}$, with probability $1 - \frac{\delta}{n}$, provided $\hat{\mathbb{P}}^{\boldsymbol{\eta}}(Z_0) \geq \frac{3}{16}$. A union bound argument gives that we can generate, with probability $1-\delta$, a set of n hyperplanes makes it possible to generate at least n new points with a margin greater than $\frac{\gamma}{17}$.

Remark 1. Theorem 2 tells us that given a noisy training sample $\mathcal{S}^{\boldsymbol{\eta}}$ and using (3), we can generate a new linearly separable training set of size n with high probability provided that $\mathcal{S}^{\boldsymbol{\eta}}$ is large enough; in addition, if a margin of $\gamma/17$ is wanted for the points of the new set, it suffices to select the generated points $\boldsymbol{\mu}^{\boldsymbol{\eta}}(Z)$ computed on the subspaces Z such that $\hat{\mathbb{P}}^{\eta}(Z) \geq \frac{3}{16}$. In Algorithm 1, the lower bound on this probability estimate is an input of the learning procedure through parameter m_0.

4 Numerical Simulations

Hyperplanes in Various Dimensions For this set of experiments, we randomly generate hyperplanes in $\mathbb{R}^d$, with $d = 2, 10, 100$. For a given hyperplane $\mathbf{w}^*$, the data to be learned from and to be classified are uniformly distributed in the hypercube $[-1; +1]^d$, where a separation margin of $\gamma = 0.02$ is enforced (i.e. points such that $|\mathbf{w}^* \cdot \mathbf{x}| < 0.02$ are discarded). Noise is added to the training data according to several noise vectors $\boldsymbol{\eta}$.

A cross-validation procedure is performed in order to choose the soft-margin parameter C for the SVM and the value of m_0. The values tested for C are $1, 10, 100$, 1000, 10000, 100000 and $0, 5, 10, 20, 40, 80$ for m_0. The cross-validation procedure works as follows: given a noisy training set of size m, and a noisy test set of the same size, n new points are generated from the training set according to (3) and a soft-margin SVM is learned on these data; then n new points are also generated from the test set using (3) as well and the accuracy of the learned classifier is measured on this new set; the parameters (C, m_0) giving the lowest test error are selected for a validation procedure made on an independent validation set. The results provided in Table 1 are the error rates measured on these validation sets (10 hyperplanes are learned for each d).

It is important to note that in the cross-validation procedure, new points must be generated as it is not possible to assess the actual error rate of a classifier on a noisy sample if the noise rates for the classes at hand are not equal.

For $d = 2$, the size m of the noisy training and test sets is 400 and the size of the validation set is 1000. For the learning procedure, $4m$ points are generated from which we remove those which do not comply with the value of m_0; $4m$ points are generated as well for the test set and they are all kept. For $d = 10$, we use the same setting using $m = 1000$, and for $d = 1000$, m is set to 3000.

The striking feature of the results is the very good ability of SAM to handle the classification noise. Indeed, in dimension 2 and 10, the achieved error rates are very low. For $d = 100$ however, the performances are a little worse but still very good with regard to the amount of noise. One point that is worth noting is that because the SVM is learned on points that are averages of areas of the space, and not the points themselves, the quality of the learning might not be optimal when no noise is present.

Banana Banana is 2-dimensional nonlinearly separable classification problem, which, in addition, has a Bayes error around 10%. The data we use are those made available by Gunnar Rätsch[1], who provides 100 training of size $m = 400$ and test sets of size 4900. In order to tackle this nonlinearly separable problem, we make use of a Gaussian

[1] http://ida.first.fraunhofer.de/projects/bench/benchmarks.htm

Noise Rate	[0.0 , 0.0]	[0.0 , 0.2]	[0.0 , 0.4]	[0.0 , 0.6]	[0.2 , 0.2]	[0.2 , 0.4]
(C, m_0)	(100,20)	(10,20)	(1000,10)	(10,5)	(1000,40)	(100000,40)
Error Rate	0.3 ± 0.4	0.6 ± 0.8	0.5 ± 0.8	1.4 ± 1.6	2.6 ± 1.4	4.4 ± 3.6

Noise Rate	[0.0 , 0.0]	[0.0 , 0.2]	[0.0 , 0.4]	[0.0 , 0.6]	[0.2 , 0.2]	[0.2 , 0.4]
Error Rate	2.0 ± 0.7	3.9 ± 0.9	3.8 ± 1.1	5.6 ± 1.6	6.2 ± 1.7	6.4 ± 1.8
Error Rate	5.1 ± 0.7	6.6 ± 1.0	9.1 ± 1.0	12.6 ± 1.0	10.0 ± 1.0	15.9 ± 1.3

Table 1. Classification error rates (in %) together with standard deviation, obtained when learning noisy hyperplanes. The first table corresponds to $d = 2$ and the value for the pair (C, m_0) obtained by a cross-validation procedure is provided for each noise rate. The first error rates in the second table correspond to $d = 10$ while the second to $d = 100$; for these experiments, it turned out that the cross-validation procedure output $C = 1, m_0 = 0$.

Noise Rate	[0.0 , 0.0]	[0.0 , 0.2]	[0.0 , 0.4]	[0.0 , 0.6]	[0.2 , 0.0]
(C, m_0)	(10,20)	(100,40)	(100,20)	(100,0)	(10,10)
Error Rate	15.1 ± 1.1	14.0 ± 1.8	14.8 ± 1.3	16.6 ± 2.8	15.3 ± 1.4

Noise Rate	[0.2 , 0.2]	[0.2 , 0.4]	[0.4 , 0.0]	[0.4 , 0.2]	[0.6 , 0.0]
(C, m_0)	(100,40)	(100,20)	(1000,10)	(100,10)	(10000,5)
Error Rate	15.3 ± 1.7	19.2 ± 4.9	13.1 ± 1.5	18.2 ± 3.7	14.3 ± 1.8

Table 2. Banana error rates and standard deviation for various noise rates. The line labelled (C, m_0) specifies the values of (C, m_0) selected by a 2-fold cross-validation (see text).

kernel of width 1.0: we pick this value because according to [7], this seems to be the parameter allowing for the best results when used with SVM. The cross-validation procedure implemented for Banana is a mix of the cross-validation procedure described for learning hyperplanes (see above) and the one implemented in [7]: it is a 2-fold validation process based on the first two training sets corrupted by classification noise that works by learning with SAM one of the two sets and testing on a set of averaged vectors computed according to (3) on the other set. The parameters that are selected are those that give the lowest test error. The number of points created to perform the learning is $4m = 1600$.

Table 2 summarizes the learning results. It must be noted that only 10 train/test sets are used to compute the mean error rates and the standard deviations. Fig. 2 depicts the learned concepts together with the corresponding noisy samples. Once again, it is striking how SAM is capable of handling the noise, even for relatively high rates. Again, a very good insensitivity of the procedure can be observed.

5 Conclusion

We have proposed a new method, the Sample Average Machine or SAM, for the learning of data altered by class conditional classification noise. Based on a relatively intuitive idea, SAM generates from a noisy sample another labelled sample of data whose correct classes are known with high probability, and learns an SVM on this newly generated sample. The simulation results are quite satisfactory, in particular from the noise

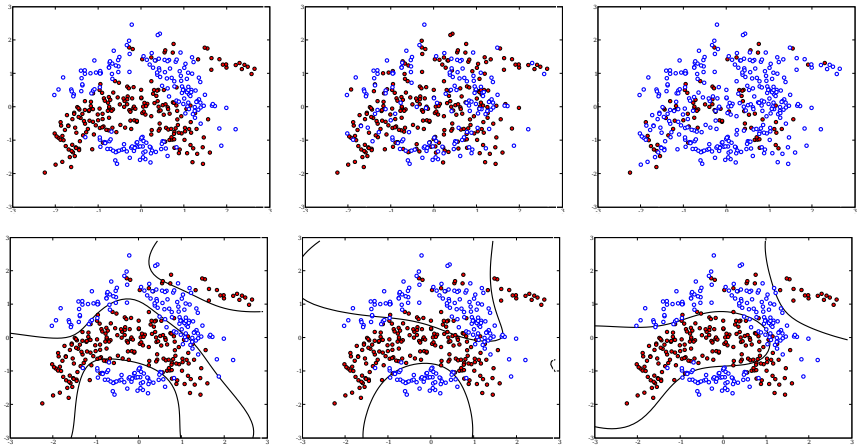

Fig. 2. Banana: the first row shows the noisy data (red/blue disks are positive/negative examples) with noises [0.0 0.0], [0.2 0.4] and [0.6 0], respectively; the second row shows the concept learned by SAM.

tolerance perspective. Moreover, we provide theoretical results which prove the good statistical properties of the new computed set and justify the learning on this sample.

The next step of our research on this problem consists in giving a formal proof that the learning on the generated set is equivalent to the learning on the training sample, directly drawn from the distribution, and, finally showing that SAM (or a derived algorithm) can be fitted into the PAC framework.

References

1. Bylander, T.: Learning Linear Threshold Functions in the Presence of Classification Noise. In: Proc. of 7th Annual Workshop on Computational Learning Theory, ACM Press, New York, NY, 1994 (1994) 340–347
2. Blum, A., Frieze, A.M., Kannan, R., Vempala, S.: A Polynomial-Time Algorithm for Learning Noisy Linear Threshold Functions. In: Proc. of 37th IEEE Symposium on Foundations of Computer Science. (1996) 330–338
3. Cohen, E.: Learning Noisy Perceptrons by a Perceptron in Polynomial Time. In: Proc. of 38th IEEE Symposium on Foundations of Computer Science. (1997) 514–523
4. S. Har-Peled, D. Roth, D.Z.: Maximum margin coresets for active and noise tolerant learning. In: Proc. of International Joint Conferences on Artificial Intelligence. (2007) 836–841
5. Ralaivola, L., Denis, F., Magnan, C.N.: CN=CPCN. In: Proc. of the 23rd Int. Conf. on Machine Learning. (2006)
6. McDiarmid, C.: On the method of bounded differences. In Surveys in Combinatorics (1989) 148–188
7. Rätsch, G., Onoda, T., Müller, K.R.: Soft Margins for AdaBoost. Machine Learning **42** (2001) 287–320