Probabilistic Approach to One-Class Support Vector Machine
Vincent Leclère, Edouard Grave, Laurent El Ghaoui

To cite this version:
Vincent Leclère, Edouard Grave, Laurent El Ghaoui. Probabilistic Approach to One-Class Support Vector Machine. 2016. hal-01404973

HAL Id: hal-01404973
https://enpc.hal.science/hal-01404973
Preprint submitted on 5 Dec 2016

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.
Probabilistic Approach to One-Class Support Vector Machine

Vincent Leclère
École des Ponts Paristech

Edouard Graves
UC Berkeley

Laurent El Ghaoui
UC Berkeley

Abstract

Classification is one of the main problem addressed by machine learning algorithms. Among them the Support Vector Machine (SVM) has attracted a lot of interest and shown success in the past decades. SVM are originally tailored for binary classification. If we have only a few example of negative dataset we can turn to one-class SVM.

In this paper we propose a probabilistic interpretation of the one-class SVM approach and an extension especially adapted in the case of highly imbalanced dataset. Indeed, we consider a binary classification problem where we represent the negative dataset by its two first moments, while still modeling the positive class by individual examples. The optimization problem is shown to have an equivalent formulation to a one-class SVM applied to the positive dataset after some preprocessing. The usual one-class SVM corresponding to the case where the negative class has mean 0 and identity variance.

We show empirically, on a protein classification task and a text classification task, that our approach achieves similar statistical performance than the two mainstream approaches to imbalanced classification problems, while being more computationally efficient.

1 Introduction

Binary classification consists in considering a set of data points \( \{ x_i \}_{i \in I} \) each associated with a label \( y_i \in \{-1, 1\} \), and trying to find a classifier function that associates to any new data point \( x \) a corresponding label \( y \).

Recently SVM has been an increasingly successful method to train such binary classifiers (see [34, 6, 33]). However, in some cases the classes can be highly imbalanced. Either because the number of negative example can be very small (novelty detection, see [29, 28, 21]), or because negative class is very large (e.g. text classification, object recognition).

For highly imbalanced datasets, a naive strategy that classifies all the examples as negative will achieve a very low classification error, since the vast majority of examples are indeed negative. Most classifiers that minimize the classification error thus lead to decision boundaries that are skewed with an important false negative rate. For most imbalanced problems, this is not acceptable, since the interesting class is the positive class. Another challenge comes from the huge number of examples from the negative class which is the bottleneck of the optimization algorithm. In both novelty detection and highly imbalanced classification a one-class SVM approach, that consider only the positive datapoints (see e.g. [28]) have shown very good results.

1.1 Related work

Many different approaches have been proposed to deal with imbalanced datasets, and the corresponding literature is too large to be summarized here. We invite the interested reader to look at the extensive review of the subject [14].

A first class of methods for imbalanced learning is based on sampling: the idea is to sample a balanced training set from the original unbalanced set of examples. Such methods are based on undersampling the negative class [17, 2], or on (synthetic) oversampling of positive examples [7, 2].

A second class of methods, referred to as cost sensitive learning, is based on assigning different misclassification costs to the negative and the positive examples [12, 8, 37].
Finally, closely related to our approach, one-class SVM were also considered by [26] in the case of extremely imbalanced datasets and for task of text classification by [20].

1.2 Contributions

In this paper we consider a probabilistic approach to the imbalanced classification problem where the negative class is only represented by its two first moments. We show that the optimization problem boils down to a one-class SVM with significant preprocessing of data exploiting informations of the negative class. In particular the usual one-class SVM consist in assuming that the negative class have mean 0 and identity for covariance matrix.

We make the following contributions:

- we propose a new formulation for the problem of imbalanced binary classification, inspired by the model of [18], where the negative class is represented by its mean and its variance, while the positive class is represented by individual examples (section 2.1);
- we show that our approach give a theoretical justification to one-class support vector machines, as introduced by [28] (section 2.2);
- we show how to incorporate moments informations on the negative class to linear one-class SVM, and propose an extension to kernel one-class SVM;
- we show that our approach is competitive with the two mainstream approaches to imbalanced classification problems (undersampling and asymmetric cost function) on two classification problems (sections 4.1 and 4.2).

1.3 Paper structure

The paper is organized as follow. § 2 present a probabilistic formulation of imbalanced binary classification and how it extend the linear one-class SVM formulation. § 3 discuss other elements that might be taken into account when looking for a classifier. § 4 presents two numerical applications of our approach.

1.4 Notations

Throughout the paper we use bold font to denote random variable and capital letter for matrices. Letter $x \in \mathbb{R}^p$ correspond to data, $y$ to the label, $w$ to a classifier. $I^+$ (resp. $I^-$) is the set of indexes of positively (resp. negatively) labeled datapoints.

2 A probabilistic approach to imbalanced classification

In this section, we propose a new formulation for solving the problem of imbalanced binary classification. In order to cope with the large number of negative examples we model the negative class by its distribution instead of a set of examples. We assume that each negative example is an independent realization of an unknown probability distribution with known expectation and covariance (this last assumption will be weakened in § 2.3). Note that we do not assume that the distribution is Gaussian, nonetheless we show that such an assumption leads to the same formulation. On the other hand, we still represent the positive class by all its examples.

2.1 A probabilistic formulation

Let $(x_i)_{i \in I^+}$ be a set of $n$ positive training examples and let $\bar{x}$ and $\Sigma$ be the mean and the covariance of the probability distribution of the negative class. In the following, we will always assume that the covariance matrix $\Sigma$ is positive definite. This is not a strong assumption, since we can always add a small regularization term $\lambda I_d$ to the covariance matrix (which can be interpreted as an uncertainty on the covariance matrix, see §2.3).

Our goal is to find the affine classifier $(w, b)$ such that all the positive examples are correctly classified while maximizing the probability of correctly classifying examples drawn from the negative distribution. As we know only the two first moments of the negative class, we take the worst possible probability among those with mean $\bar{x}$ and covariance $\Sigma$:

\[
\max_{w, b} \inf_{\mathbf{x} \sim (\bar{x}, \Sigma)} \mathbb{P}(w^\top \mathbf{x} - b \leq 0), \tag{1a}
\]

\[s.t. \quad w^\top x_i - b \geq 0, \quad i \in I^+. \tag{1b}
\]

where $\mathbf{x} \sim (\bar{x}, \Sigma)$, refers to the class of probability distributions with mean $\bar{x}$ and covariance $\Sigma$. In other words, our goal is to maximize the specificity of the separating hyperplane, while correctly classifying all the positive examples.

**Remark 1.** If the mean $\bar{x}$ and variance $\Sigma$ of the negative class is obtained from the empirical mean and variance of the negative set of datapoints, then the value of Problem (1) is a lower bound of the fraction of negative points $\{x_i, i \in I^-\}$ rightly classified by its solution.

Indeed, the uniform distribution over the set of negative points $\{x_i, i \in I^-\}$ is among the set of probability where the infimum is taken.

According to the following lemma from [18], which is a consequence of a theorem by [22], the minimum speci-
ficity over all probabilities with given mean and covariance has a geometric characterization:

**Lemma 1.** Let $\bar{x} \in \mathbb{R}^p$, $\Sigma \in \mathbb{R}^{p \times p}$ a positive definite matrix, $w \in \mathbb{R}^p$ such that $w \neq 0$, $b \in \mathbb{R}$ and $\alpha \in (0, 1)$. Then, the condition $\inf_{x \sim (\bar{x}, \Sigma)} P(b - x^\top w \geq 0) \geq \alpha$, holds if and only if $b - \bar{x}^\top w \geq \kappa(\alpha)\sqrt{w^\top \Sigma w}$, where $\kappa(\alpha) = \sqrt{\frac{\alpha}{1-\alpha}}$.

Using this result, the optimal hyperplane we are looking for is thus the optimal solution of the following optimization problem

$$
\max_{\alpha, w, b} \alpha \quad \text{(2a)}
$$

subject to

$$
b - \bar{x}^\top w \geq \kappa(\alpha)\sqrt{w^\top \Sigma w}, \quad \text{(2b)}
$$

$$
x_i^\top w - b \geq 0, \quad \forall i \in I^+. \quad \text{(2c)}
$$

Since the function $\kappa : \alpha \mapsto \sqrt{\frac{\alpha}{1-\alpha}}$ is increasing on $[0, 1]$, this problem is equivalent to

$$
\max_{\kappa > 0, w, b} \kappa \quad \text{(3a)}
$$

subject to

$$
b - \bar{x}^\top w \geq \kappa \sqrt{w^\top \Sigma w}, \quad \text{(3b)}
$$

$$
x_i^\top w - b \geq 0, \quad \forall i \in I^+. \quad \text{(3c)}
$$

Note that $\kappa$ being positive, $w = 0$ is the worst admissible solution. From now on we assume that there is a non-zero admissible solution, or in other words that $\bar{x}$ is not in the convex hull of positive points. This assumption can be relaxed through penalized slack variables (see §3.1).

Exploiting the homogeneity in the variable $w$, and the suboptimality of $w = 0$, we now impose that $\kappa \sqrt{w^\top \Sigma w} = 1$, and since the function $x \mapsto 1/\sqrt{x}$ is decreasing we obtain

$$
\min_{w} w^\top \Sigma w \quad \text{(4a)}
$$

subject to

$$
b - \bar{x}^\top w \geq 1, \quad \text{(4b)}
$$

$$
x_i^\top w - b \geq 0, \quad \forall i \in I^+. \quad \text{(4c)}
$$

This problem is a convex program (see [5]), which has an interesting geometric interpretation: the optimal $w$ is orthogonal – for the scalar product defined by the inverse covariance matrix – to the projection of $\bar{x}$ to the convex hull of the positive points. This interpretation is detailed in Annex A, and illustrated in Figure 1.

**Remark 2.** Let $v^\perp$ be the optimal value of Problem (3) and $w^$ an optimal solution. Then assuming that the first and second order moment of the negative class are exact, the negative misclassification probability $P(x^T w^ - b \geq 0)$ is bounded by $\frac{v^\perp}{1 + x^\top w^\perp}$.

By Remark 1, this bound is also an upper bound on the number of misclassified negative points.

**Remark 3.** If we assume that the distribution of the negative points is Gaussian with the given mean and variance, then we have $P(x^T w - b \leq 0) = P(N(0, 1) \leq \frac{b - \bar{x}^\top w}{\sqrt{\Sigma w}})$. Hence, we obtain Problem (2), with $\kappa(\alpha) = \Phi^{-1}(\alpha)$, where $\Phi$ is the cumulative distribution function of the standard normal Gaussian distribution. In particular, an assumption of normality of the negative class also leads to Problem (3).

The only improvement a normality assumption bring is an improved negative error estimation, as we obtain

$$
P\left(X^T w^\perp - b \geq 0\right) = 1 - \Phi\left(\frac{1}{\sqrt{v^\perp}}\right) \leq \frac{v^\perp}{1 + v^\perp}.
$$

In the next section we show how this formulation extend the one-class SVM formulation introduced by [28].

### 2.2 Relation to support vector machines

If we assume that the covariance matrix $\Sigma$ is equal to the identity matrix, then Program (3) is equivalent to hard margin SVM where the only negative point considered is the mean of the negative class. Furthermore, eliminating $b$ from Problem (3), yields

$$
\max_{w} w^\top \Sigma w \quad \text{(4a)}
$$

subject to

$$
(x_i - \bar{x})^\top w \geq 1, \quad \forall i \in I^+. \quad \text{(4b)}
$$

If the mean $\bar{x} = 0$, then we have a one-class SVM formulation. In other words, an hard-margin linear one-class SVM classifier is the classifier that truly classify every example while maximizing the probability of truly classifying a negative example assuming that the negative points are Gaussian centered in 0 with variance identity.

Formulation 4 allow to integrate first and second order information over the negative class. It is a one-class support vector machine formulation, where we minimize the Mahalanobis norm (corresponding to the covariance matrix of the negative class distribution) instead of the $\ell_2$ norm, and separate the positive points from the mean of the negative class instead of the origin.

In other words, our formulation consist in applying one-class SVM to the set of points $x_i = \Sigma^{-1/2}(x_i - \bar{x})$. If this point of view is useful for interpretation, it is not numerically efficient. Indeed, computing $\Sigma^{-1/2}$ can be challenging, and even the translation of the positive data might destroy some existing sparsity that could be numerically exploited.

Mahalanobis norm SVM has been used previously, especially in the case of one-class approach (see [32, 16,
we require \( \|\Sigma\| \leq \epsilon \|\Sigma\| \), with high probability, \( \bar{x}_i - x_i \leq \nu^2 \).

To sum up, asymptotically, with high probability, the empirical estimates lies in

\[
Q = \{(x, \Sigma) \mid (\bar{x} - \bar{x}_0)^T \Sigma^{-1} (\bar{x} - \bar{x}_0) \leq \nu^2, \|\Sigma - \Sigma_0\| \leq \rho\}.
\]

We can take a robust approach over this set in the sense that we consider the worst mean and covariance in the set. More precisely we want to solve

\[
\max_{w,b} \inf_{(x, \Sigma) \in Q} \inf_{x \sim (x, \Sigma)} P(w^T x - b \leq 0),
\]

\[
s.t. \ w^T x_i - b \geq 0, \quad \forall i \in I^+.
\]

This amount to replacing constraint (2b) by

\[
\inf_{(x, \Sigma) \in Q} b - \bar{x}^T w - \kappa(\alpha) \sqrt{w^T \Sigma w} \geq 0.
\]

For a given \( \Sigma \) and \( w \), we know that [18, Appendix B]

\[
\frac{\max}{\bar{x} \sim (\bar{x} - \bar{x}_0)^T \Sigma^{-1} (\bar{x} - \bar{x}_0) \leq \nu^2} x^T w = \bar{x}_0^T + \nu \sqrt{w^T \Sigma w}.
\]

Hence, constraint (6) is equivalent to

\[
\min_{\Sigma: \|\Sigma - \Sigma_0\| \leq \rho} b - \hat{x}^T w - (\kappa(\alpha) + \nu) \sqrt{w^T \Sigma w} \geq 0.
\]

Furthermore we know that (see [18, Appendix C], replacing the Frobenius norm by the spectral norm if necessary)

\[
\max \{ w^T \Sigma w \mid \|\Sigma - \Sigma_0\| \leq \rho \} = w^T (\Sigma_0 + \rho I) w.
\]

Finally, constraint (6) can be written

\[
b - \hat{x}^T w \geq (\kappa(\alpha) + \nu) \sqrt{w^T (\Sigma_0 + \rho I) w}.
\]

Hence solving the robust formulation (5) is equivalent to solving

\[
\max_{w} \min_{\Sigma: \|\Sigma - \Sigma_0\| \leq \rho} w^T (\Sigma_0 + \rho I) w
\]

s.t.

\[
\begin{align*}
& b - \hat{x}^T w \geq 1, \\
& x_i^T w - b \geq 0, \quad \forall i \in I^+.
\end{align*}
\]
Remark 4. It is remarkable that the uncertainty in the mean does not affect the solution of the problem whereas the uncertainty in the variance yield an $\ell_2$ regularization of the optimization problem.

However, the uncertainty in the mean will affect the probability of negative error estimate

$$P\left( x^T w^* - b \geq 0 \right) \leq \kappa^{-1}\left( \frac{1}{\sqrt{D}} - \nu \right),$$

(8)

where $w^*$ (resp. $v^\delta$) is the optimal solution (resp. value) of Problem (7).

3 Other objectives

Our approach consists in maximizing the specificity of the separating hyperplane, while correctly classifying all the positive examples. Other elements might be taken into account when looking for a classifier. In this section we address some of these elements: soft margin in §3.1, factor model representation of the covariance matrix in §3.2 and non-linear kernel in §3.3.

3.1 Soft-margin

Similarly to SVM, the constraint that all positive examples should be correctly classified might be unrealistic in practice. We thus propose to relax these constraints by penalizing slack variables.

More precisely two formulation are readily available, one can be derived from Problem (3) giving

$$\min_{w, \xi, \eta} w^T \Sigma w + \frac{1}{\nu^+ |I^+|} \sum_{i \in I^+} \xi_i + \frac{\eta}{\nu^-}$$

(9a)

s.t.  $b - \bar{x}^T w \geq 1 - \eta$,

(9b)

$$x_i^T w - b \geq \xi_i, \quad \forall i \in I^+,$$

(9c)

$$\xi \geq 0, \quad \eta \geq 0.$$  

(9d)

This formulation is very close to differential cost SVM, $\nu^+$ and $\nu^-$ being parameters in $(0, 1]$.

The second formulation is derived from Problem (4) and gives

$$\max_{\kappa, w} w^T \Sigma w + \frac{1}{\nu^+ |I^+|} \sum_{i \in I^+} \xi_i$$

(10a)

$$\quad \left( x_i - \bar{x} \right)^T w \geq 1 - \xi_i, \quad \forall i \in I^+.$$  

(10b)

This formulation is equivalent to soft-margin one-class linear SVM applied to the preprocessed positive data (see §2.2 for more details). We call it Moment-based imbalanced binary classifier of MIBC, and this is the formulation used in the numerical experiments of §4.

3.2 Factor model of the variance matrix

A factor model of the covariance matrix $\Sigma$ is a couple of matrices $(D, F)$ where $D$ is diagonal and $F \in \mathbb{R}^{n \times k}$ is tall, such that $\Sigma \approx D + FF^T$. Finding the best factor model is a well studied problem, with numerous possible approaches (see [9, 10] and references therein).

We sketch a way of finding a good factor model from a set of negative datapoints. Let $X = [x_1, \ldots, x_{|I^-|}]$ be the data matrix where each column is a negative data point; $\bar{x}$ be the empirical mean of the negative points, and $\bar{X} = [x_1 - \bar{x}, \ldots, x_{|I^-|} - \bar{x}]$ the centered version of $X$. Without loss of generality we assume that each feature (row of $X$) has variance equal to 1. Consider the singular vector decomposition (svd) of $\bar{X}$, $\bar{X} = USV^T$, where $U$ and $V$ are unitary matrices, and $S$ is diagonal with decreasing positive values. Then we have $\Sigma = 1/|I^-| \bar{X} \bar{X}^T = 1/|I^-| US^2 U^T$. From this equality it is easy to see that

$$1/|I^-| U_k S_k^2 U_k \leq \Sigma \leq 1/|I^-| (U_k S_k^2 U_k + s_{k+1}^2 I),$$

where $U_k$ is composed of the $k$ first left singular vectors of $\bar{X}$ (column of $U$), $S_k$ is the diagonal matrix of the $k$ first singular values (values of $S$), and $s_{k+1}$ is the $k + 1$th singular value. The inequalities being understood with respect to the cone of positive semidefinite matrices.

From this analysis we suggest the following factor model

$$D = \frac{s_{k+1}^2}{|I^-|} I, \quad F = \frac{1}{|I^-|} U_k S_k$$

(11)

and we have, for any vector $w$

$$w^T FF^T w \leq w^T w \leq w^T (D + FF^T) w.$$ 

Hence, the optimal value of

$$\min_{w, z} w^T Dw + z^T z$$

s.t.  $w^T (x_i - \bar{x}) \geq 1, \quad \forall i \in I^+$

$$F^T w = z$$

is an upper bound to the optimal value of Problem (4), leading to an upper bound on the best negative error probability we could achieve see Remark 2.

Remark 5. The $k$ first singular vectors and values of $\bar{X}$ can be obtained in a number of way. In particular if $X$ is sparse (as, for example, in text analysis) it is important to keep this feature in mind when computing

In the general case scale each row of $X$ by its standard deviation, apply the described procedure, and multiply the value of $D$ by the corresponding variance, and the row of $F$ by the corresponding semideviation.
3.3 Kernelization

Our original problem (1) considered a linear classifier \( w^T x - b \). However, it is sensible to look for non-linear classifiers. The standard method in the SVM community is known as the kernel trick.

The kernel trick consist in extending the feature space and look for a linear classifier in the extended space. More precisely we consider a feature function \( \varphi : \mathbb{R}^n \mapsto \Phi \), and we apply the linear methodology to the points \( \{ \varphi(x_i) \}_{i \in I} \). The second part of the trick consist in remarking that the optimal solution \( w \) is a linear combination of the data points \( w = \sum_{i \in I} \alpha_i \varphi(x_i) \). Hence, in the optimization problem we have to solve \( \varphi(x_i) \) only appear as a scalar product \( \varphi(x_i)^\top \varphi(x_j) \). So if we choose a feature function \( \varphi \) such that \( K(x,y) = \varphi(x)^\top \varphi(y) \) is easy to compute the linear SVM optimization problem in the extended space is as easy to solve as the linear SVM optimization problem in the original input space.

We have seen in Section 2.2 that Problem (1) (or its robust version) is equivalent to solving a one-class linear SVM for the preprocessed positive class

\[
\hat{x}_i = \Sigma^{-1/2}(x_i - \bar{x}), \quad i \in I^+.
\]

Hence, it is natural to try a Kernelized one-class SVM to the preprocessed points \( \{ \hat{x}_i \}_{i \in I^+} \). Preprocessing the positive points could be time consuming and lead to loss of any sparsity of original data. However, it is easy to see that for the most classical kernels \( K \) (polynomial, Gaussian and sigmoidal) we have \( K(\hat{x}_i, \hat{x}_j) = \tilde{K}(x_i, x_j) \) where \( \tilde{K} \) is an easy to compute function. In other word applying classical to the preprocessed positive class is equivalent to apply a customized kernel to the original points. You find in Table 2 the definition of such customized kernel matrix. As usual \( \bar{x} \) represent the mean of the negative class, and \( \Sigma \) its covariance matrix (plus a scalar times the identity matrix in a robust version taking into account uncertainty on the covariance matrix).

Remark 7. One could think of replacing \( x_i \) by \( \varphi(x_i) \) directly in Problem (1). However, following the same steps as in the linear case yield an optimization problem with as many variable as positive and negative points, negating one of the main interest of our approach: reducing the computational cost of the problem by reducing the number of variables.

4 Numerical experiments

4.1 Small scale experiment: protein classification

In this section, we present experiments performed on small dimensional datasets. We compare our moment-based imbalanced binary classifier (MIBC) with two standard strategies for imbalanced classification problems, using SVMs: undersampling the negative class and using different costs for the negative and positive examples. We will use the liblinear [11] implementation of linear support vector machines, and an implementation of our approach using Mosek\(^2\). It is thus important to note that our implementation is based on a general purpose quadratic programming solver.

A custom implementation, for example based on the SMO algorithm, should greatly improve the efficiency of our moment-based imbalanced binary classifier.

<table>
<thead>
<tr>
<th>Dataset</th>
<th># positive</th>
<th># negative</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhosS</td>
<td>613</td>
<td>10,798</td>
<td>17</td>
</tr>
<tr>
<td>PhosT</td>
<td>140</td>
<td>9,051</td>
<td>64</td>
</tr>
<tr>
<td>PhosY</td>
<td>136</td>
<td>5,103</td>
<td>37</td>
</tr>
<tr>
<td>CAM</td>
<td>942</td>
<td>17,974</td>
<td>19</td>
</tr>
</tbody>
</table>

Table 1: Basic statistics about the different datasets.

4.1.1 Datasets

We evaluated the different methods on four datasets introduced by [25], which are publicly available\(^3\). These datasets correspond to protein classification problems, such as predicting protein phosphorylation sites (PhosT, PhosS, PhosY) or predicting binding regions (CAM). Following the approach proposed by [25], we keep the 150 features which are the most correlated to the class labels. The ratio of negative examples to positive ones varies from 17.6 to 64.6 on the different datasets. Basic statistics about those are given in Table 1.

4.1.2 Methodology

For each dataset, we use 50% of the examples as training set, 20% as validation set and 30% as test set. For all methods, we chose \( C \) in the set \( \{10^3, ..., 10^{-4}\} \). When undersampling the negative class, we keep as many negative examples as positive examples. For

\(^2\)www.mosek.com

\(^3\)www.informatics.indiana.edu/predrag/publications.htm
the asymmetric cost function method, we consider the following ratios between the weights of the positive and negative examples: \{1.0, r/4, r/2, r, 2r\}, where \(r\) is the ratio of number of negative examples to the number positive ones. We replicated the experiments over twenty random splits of the data.

\subsection*{4.1.3 Discussion}

We report areas under the ROC curve for the four datasets in Table 2, computational times in Table 3 and ROC curves for two datasets in Figure 3 (PhosT and PhosY). We performed a paired samples \(t\)-test to determine if our results are statistically significant.

First, we observe that our moment-based imbalanced binary classifier always outperforms the undersampling approach, while performing at least as well as the cost sensitive method. Second, the two datasets on which our method outperforms the asymmetric cost function SVM (PhosT and PhosY) correspond to the highest ratio of number of negative to positive examples (64 and 37 respectively). This seems to indicate

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
 & \(K(\hat{x}_i, \hat{x}_j)\) & \(K(x_i, x_j)\) \\
\hline
Polynomial & \((\hat{x}_i^\top \hat{x}_j + c)^p\) & \((x_i^\top \Sigma^{-1} x_j - (\Sigma^{-1} \bar{x})^\top (x_i + x_j) + c + \bar{x}^\top \Sigma^{-1} \bar{x})^p\) \\
Gaussian & \(e^{-\gamma||\hat{x}_i - \hat{x}_j||^2}\) & \(\exp\left(-\gamma(x_i - x_j)^\top \Sigma^{-1} (x_i - x_j)\right)\) \\
Sigmoidal & \(\tanh(\alpha \hat{x}_i^\top \hat{x}_j + c)\) & \(\tanh(\alpha x_i^\top \Sigma^{-1} x_j - (\alpha \Sigma^{-1} \bar{x})^\top (x_i + x_j) + c + \bar{x}^\top \Sigma^{-1} \bar{x})\) \\
\hline
\end{tabular}
\end{table}

Figure 2: Standard kernel applied to preprocessed positive datapoints \(\hat{x}_i\) is equivalent to modified kernel \(\hat{K}\) applied to original positive datapoints.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
 & This work & Cost-sensitive & Sampling \\
\hline
PhosS & 77.2\(^\dagger\) ± 0.7 & 76.8 ± 0.8 & 74.3 ± 1.1 \\
PhosT & 77.4\(^\dagger\) ± 1.7 & 73.0 ± 2.0 & 72.0 ± 1.5 \\
PhosY & 76.2\(^\dagger\) ± 1.5 & 72.8 ± 1.7 & 70.1 ± 2.1 \\
CAM & 78.2 ± 0.5 & 78.1 ± 0.5 & 75.3 ± 0.4 \\
\hline
\end{tabular}
\end{table}

Table 2: Areas under the ROC curve (with confidence intervals), averaged over twenty experiments. \(^\dagger\) indicates that our method is significantly better than the two others, (with \(p\)-value \(p < 0.01\)).

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
 & This work & Cost-sensitive & Speed-up \\
\hline
PhosS & 146 & 325 & 2.2\times \\
PhosT & 23 & 112 & 4.8\times \\
PhosY & 19 & 41 & 2.1\times \\
CAM & 425 & 605 & 1.4\times \\
\hline
\end{tabular}
\end{table}

Table 3: Computational times, in milliseconds, required to solve one problem, averaged over twenty experiments.
that our method is particularly adapted to highly imbalanced datasets. Finally, our method is computationally more efficient, leading to speed-up between 1.4 and 4.8 over cost-sensitive SVM, while obtaining as good or even better statistical performances. We remind our reader that we implemented our method using Python and Mosek, and it is thus certainly possible to get much better performances.

4.2 Large scale experiment: text analysis

In this section, we report experiments performed on the task of text classification. We will follow the same methodology as described in the section 4.1.2. Since bag-of-words representations of textual documents live in high dimensional spaces, we propose to replace the full covariance matrix of the negative class by its diagonal.

4.2.1 Dataset

We use the Reuters rcv1 dataset, introduced by [19], which is a classical test bed for text classification methods. Each document of the corpus is tagged with respect to three different category sets: topics, industries and regions. We consider classification problems that consist in classifying documents that are labeled with a given topic label v.s. the rest of the documents. There are 104 different topics, and we will thus consider only a subset of the 104 possible classification tasks. Since we want to focus on highly imbalanced classification problems, we set the ratio of negative examples to positive examples to 1,000.

4.2.2 Discussion

We report areas under the ROC curve in Table 4, computational times in Table 5. We performed a paired samples t-test to determine if our results are statistically significant.

<table>
<thead>
<tr>
<th>Topic</th>
<th>This work</th>
<th>Cost-sensitive</th>
<th>Sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>89.7 ± 1.0</td>
<td>89.9 ± 1.4</td>
<td>87.7 ± 1.2</td>
</tr>
<tr>
<td>9</td>
<td>96.1 ± 0.7</td>
<td>96.3 ± 0.8</td>
<td>94.1 ± 1.3</td>
</tr>
<tr>
<td>25</td>
<td>95.1 ± 0.8</td>
<td>94.3 ± 1.6</td>
<td>93.7 ± 1.2</td>
</tr>
<tr>
<td>33</td>
<td>96.0 ± 0.4</td>
<td>95.7 ± 0.6</td>
<td>93.9 ± 0.7</td>
</tr>
<tr>
<td>59</td>
<td>96.1 ± 0.4</td>
<td>95.9 ± 1.4</td>
<td>95.0 ± 0.6</td>
</tr>
<tr>
<td>84</td>
<td>96.9 ± 0.8</td>
<td>96.4 ± 1.5</td>
<td>96.3 ± 0.9</td>
</tr>
</tbody>
</table>

Table 4: Areas under the ROC curve (with confidence intervals), averaged over ten experiments. Differences between our moment-based imbalanced binary classifier and subsampling results are statistically significant (with p-value $p < 0.01$).

<table>
<thead>
<tr>
<th>Topic</th>
<th>This work</th>
<th>Cost-sensitive</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>33</td>
<td>1088</td>
<td>33×</td>
</tr>
<tr>
<td>9</td>
<td>49</td>
<td>1451</td>
<td>29×</td>
</tr>
<tr>
<td>25</td>
<td>56</td>
<td>1211</td>
<td>21×</td>
</tr>
<tr>
<td>33</td>
<td>74</td>
<td>1788</td>
<td>24×</td>
</tr>
<tr>
<td>59</td>
<td>62</td>
<td>1299</td>
<td>21×</td>
</tr>
<tr>
<td>84</td>
<td>56</td>
<td>2056</td>
<td>36×</td>
</tr>
</tbody>
</table>

Table 5: Computational times, in milliseconds, required to solve one problem, averaged over ten experiments.

We observe that our moment-based imbalanced binary classifier achieves similar statistical performances than the cost-sensitive method, while generally outperforming the undersampling approach. Finally, our approach to imbalanced classification is much more computationally efficient than a SVM with asymmetric costs, leading to speed-up between 21 and 36.

References


