Non-smooth classification model based on new smoothing technique

S. LYAQINI\textsuperscript{a}, M. NACHAOUI\textsuperscript{a}, M. QUAFAFOU\textsuperscript{b}

\textsuperscript{a} Laboratoire de Mathématiques et Application, FST de Béni Mellal Université Sultan Moulay Slimane.B.P. 523, Béni Mellal. \\
\textsuperscript{b} LSIS - CNRS UMR 6168 Université Aix-Marseille, France.

E-mail: lyaqini.soufiane@gmail.com

Abstract. This work describes a framework for solving support vector machine with kernel (SVMK). Recently, it has been proved that the use of non-smooth loss function for supervised learning problem gives more efficient results [1]. This gives the idea of solving the SVMK problem based on hinge loss function. However, the hinge loss function is non-differentiable (we can’t use the standard optimization methods to minimize the empirical risk). To overcome this difficulty, a special smoothing technique for the hinge loss is proposed. Thus, the obtained smooth problem combined with Tikhonov regularization is solved using a stochastic gradient descent method. Finally, some numerical experiments on academic and real-life datasets are presented to show the efficiency of the proposed approach.

Keywords: Stochastic gradient descent, Supervised learning, Non-smooth loss function, Optimization, Smooth approximation, Non parametric classification

1. Introduction
Support vector machine (SVM) is a high-efficiency statistical learning method based on the structural risk minimization principle. This classification model is now the most commonly applied machine learning technique which successfully used in many engineering related applications [2]. The (SVM) is a supervised learning method which aim is to find a function in a multidimensional space that is able to separate training data with known class labels. Concretely, this technique consists to project nonlinear separable samples onto another higher dimensional space by using different types of kernel functions. Roughly speaking, the (SVM) is formally described as an optimization problem of minimizing loss functional over some training sets. Recently, the use of non-smooth loss function has proved to be a successful approach for supervised machine learning and (SVM) [1, 3]. To circumvent the difficulty due to non-differentiability of the loss function, some studies have focused on introducing the smooth support vector machine (SVM) [3, 4]. However, this studies has been done in the particular case, where the hypothesis space is linear one. In this work we consider a support vector machine based on on hinge loss function and kernels. As it’s well known the (SVM) problem is one of highly ill-posed problem. Therefore, regularization methods should be employed for obtaining stable approximate solutions. This can be done, generally, by added a regularization term [5, 6] or developpes some algorithms which has a regularizing effect [7, 8, 9, 10]. Thus, we use a smoothing technique firstly introduce in [11], which transforms (SVM) problem into smooth
one that is twice differentiable and convex. Then the obtained problem is solved by using Tikhonov regularization and stochastic gradient descent method. In this case, we prove that when the smoothing parameter $\gamma$ go to infinity the smooth empirical risk $R_{\gamma \text{emp}}$ converges to the empirical one $R_{\text{emp}}$. Finally, we present some numerical validations of the proposed computational methodology. Most notably, our experimental results, which cover academic and real-life data, indicate that the proposed approach is an efficient and helpful tool in machine learning.

We briefly summarize the contents of the paper. In Section 2, we present the setting of the classification problem and its reformulation as a minimization problem of empirical risk, using hinge loss function. Section 3 is concerned by smooth approximation of the hinge loss function. Then, we show that when the smoothing parameter $\gamma$ go to infinity the smooth empirical risk $R_{\gamma \text{emp}}$ converges to the empirical one $R_{\text{emp}}$. Furthermore, we present a numerical algorithm based on stochastic gradient descent method. In Section 4, we evaluate the efficiency of the proposed approach using academic and real-life experiences.

2. Formulating the Classification Problem

In classification problem, we consider a training data set $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$, where $x_i \in \mathcal{X} \subset \mathbb{R}^d$ called the input space and $y_i \in \mathcal{Y} = \{-1, 1\}$ is the output space. The relation between the input $x \in \mathcal{X}$ and the output $y \in \mathcal{Y}$ is described by a probability distribution $p(x, y)$ defined on $\mathcal{X} \times \mathcal{Y}$. The distribution $p$ is known only on $\mathcal{D}$. Concretely, based on the dataset $\mathcal{D}$, the classification problem investigates how to find a function $f : \mathcal{X} \to \mathcal{Y}$ that can precisely predicts the output $y_i$ of new input $x_i$. The ability of $f$ to describe the distribution $p$ is measured by its expected risk as follows:

$$R[f] = \int_{\mathcal{X} \times \mathcal{Y}} L(f(x), y)d\rho(x, y),$$

where $L$ is a non negative function, called loss function, which measures the error between $y$ and its prediction $f(x)$. The most common loss function is the square loss defined by $L(\omega, y) = (1 - \omega y)^2$ chosen for its good properties in particular the differentiability. There are some others kind of loss function which do not have this property such as the hinge loss $L(\omega, y) = \max(1 - \omega y, 0)$. It was proved that the non-smooth loss function lead to better convergence rate than the classic square loss [1]. This empowers us to say that hinge loss give efficient learning algorithm [12]. In the sequel, we use the hinge loss function.

![Figure 1: Various loss functions used in classification.](image-url)
The ideal estimator $f^\dagger$ is the minimizer of

$$\min_{f \in \mathcal{H}} \mathcal{R}[f],$$

where $\mathcal{H}$ is the space of measurable functions for which $\mathcal{R}[f]$ is well defined, named hypotheses space, which we will discuss in the next subsection.

### 2.1. Reproducing kernel Hilbert space (RKHS)

In last decades, the Reproducing kernel Hilbert Spaces have played an important role in a number of successful applications [13, 1]. We begin by providing some background on RKHS [13]. Let $\mathcal{H}$ be a Hilbert space of real valued functions defined on $\mathcal{X}$ and equipped with an inner product $<.,.>_{\mathcal{H}}$.

**Definition 2.1.** A function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a reproducing kernel of $\mathcal{H}$ if it verifies the following properties:

- For every fixed $x \in \mathcal{X}$, $K(x,y) \in \mathcal{H}$, $\forall y \in \mathcal{X}$,
- $f(x) = <f,K_x>_{\mathcal{H}}$, $\forall f \in \mathcal{H}$, where $K_x(.) := K(x,.)$.

Such Hilbert space $\mathcal{H}$ is called a reproducing kernel Hilbert space (RKHS). It is also known that a kernel $K(.,.)$ is a reproducing kernel if and only if it is positive definite, that is:

$$\sum_{i,j=1}^{n} a_i a_j K(x_i,x_j) \geq 0,$$

for any $n \in \mathbb{N}, x_1,\ldots,x_n \in \mathcal{X}$ and $a_1,\ldots,a_n \in \mathbb{R}$.

Also, this function $K$ is such that the space $\mathcal{H}$ can be read

$$\mathcal{H} = \text{span}\{K_x(.) \mid x \in \mathcal{X}\},$$

as the closure of the subspace of all linear combinations of $K_x(.)$.

### 2.2. The empirical risk

While the probability distribution $p(x,y)$ is unknown, the ideal estimator $f^\dagger$ cannot be found directly. In practice, the training set $\mathcal{D}$ is used to find an approximation of the expected risk, called the empirical risk, given by

$$\mathcal{R}_{\text{emp}}[f] = \frac{1}{n} \sum_{i=1}^{n} \max(1 - f(x)y, 0).$$

The minimizer $f^*_D$

$$\min_{f \in \mathcal{H}} \mathcal{R}_{\text{emp}}[f],$$

(1)

can be seen as a rough approximation of $f^\dagger$. Rosasco et al in [12], has showed that the minimizer $f^*_D$ of the empirical risk with the hinge loss function mimics the behavior of $f^\dagger$ better than the square loss.
3. Smooth hinge loss

From the figure 1, we observe that, the hinge loss functions is non differentiable. This precludes the use of standard optimization which required the assumption of differentiability of the cost function. This is why, we opt here for the so called smooth approximation [14, 11] for approximating the hinge loss function with a continuous and differentiable one. This in fact allowing us to formulate the gradient kinds method to solve the resulting optimization problems.

The hinge loss function can be accurately approximated by a smooth function which is twice differentiable and convex defined below. For all \( u \in \mathbb{R} \), the max function is approximated by a smooth function,

\[
\max_{\gamma}(u, 0) = \left( u + \frac{1}{\gamma} \log(1 + \exp(-\gamma u)) \right), \quad \forall \gamma > 0.
\]

We first notice that the loss function is always a true function of only one variable \( u \), with \( u = 1 - \omega y \).

Let us denote by \( L_\gamma(u) \) the smoothed loss function with parameter \( \gamma \) of \( L(u) = \max(u, 0) \) defined by

\[
L_\gamma(u) = \max_{\gamma}(u, 0), \quad \forall \gamma > 0.
\]

We have that as \( \gamma \) tends to \(+\infty\), the smoothed hinge loss \( L_\gamma \) approaches the hinge loss \( L \), which is clear in the figure 2.

![Smoothed hinge loss](image)

**Figure 2: Smoothed hinges loss.**

The \( L_\gamma \) is used here to replace the loss function of (2.2) to obtain our smooth empirical risk, given by

\[
R_{\text{emp}}^\gamma[f] = \frac{1}{n} \sum_{i=1}^{n} L_\gamma(1 - f(x_i)y_i),
\]

which is twice differentiable and convex.

In the sequel, we shall show that the smooth empirical risk \( R_{\text{emp}}^\gamma \) converges to the empirical risk \( R_{\text{emp}} \) as the smoothing parameter \( \gamma \) go to infinity. For this, we introduce the following lemma, which bounds the difference between hinge loss function \( L \) and its smooth approximation \( L_\gamma \).
Lemma 3.1. For all $u \in \mathbb{R}$, then

$$|\mathcal{L}_\gamma(u) - \mathcal{L}(u)| \leq \frac{\log(2)}{\gamma}$$  \hspace{1cm} (2)

Proof. The proof will be split into two cases, which are $u > 0$ and $u < 0$.

For $u > 0$, we have that $\mathcal{L}(u) = u$, then

$$|\mathcal{L}_\gamma(u) - \mathcal{L}(u)| = |u + \frac{1}{\gamma} \log [1 + \exp (-\gamma u)] - u|$$

$$= \frac{1}{\gamma} \log [1 + \exp (-\gamma u)]$$

$$\leq \frac{\log(2)}{\gamma}.$$ 

Since $u \mapsto \frac{1}{\gamma} \log [1 + \exp (-\gamma u)]$ is a decreasing function and $u \geq 0$.

For $u \leq 0$, we have that $\mathcal{L}(u) = 0$, then

$$|\mathcal{L}_\gamma(u) - \mathcal{L}(u)| = |\mathcal{L}_\gamma(u)|$$

$$\leq \mathcal{L}_\gamma(0) = \frac{\log(2)}{\gamma}.$$ 

Since $\mathcal{L}_\gamma$ is an increasing function and $u \leq 0$, then $\mathcal{L}_\gamma(u) \leq \mathcal{L}_\gamma(0)$. \hfill \Box

According to lemma 3.1 we have the following result

Lemma 3.2. For all $f \in \mathcal{H}$, then

$$|\mathcal{R}_{emp}^\gamma[f] - \mathcal{R}_{emp}[f]| \leq \frac{\log(2)}{\gamma}.$$ \hspace{1cm} (3)

Proof. For all $f \in \mathcal{H}$, we have

$$\mathcal{L}_\gamma(1 - f(x_i)y_i) - \mathcal{L}(1 - f(x_i)y_i) \leq \frac{\log(2)}{\gamma}, \quad \forall i = 1, \ldots, n.$$ 

Then

$$\frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_\gamma(1 - f(x_i)y_i) - \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(1 - f(x_i)y_i) \leq \frac{1}{n} \sum_{i=1}^{n} \frac{\log(2)}{\gamma}.$$ 

Thus

$$0 \leq \mathcal{R}_{emp}^\gamma[f] - \mathcal{R}_{emp}[f] \leq \frac{\log(2)}{\gamma},$$

since $\mathcal{R}_{emp}^\gamma[f] \geq \mathcal{R}_{emp}[f]$.

Therefore

$$|\mathcal{R}_{emp}^\gamma[f] - \mathcal{R}_{emp}[f]| \leq \frac{\log(2)}{\gamma}.$$ \hfill \Box
In the sequel, we consider the approximate optimization problem of (1) given by
\[
\min_{f \in \mathcal{H}} \mathcal{R}_{\text{emp}}[f].
\]
This problem is a strongly convex minimization problem. Moreover, the cost function in (4) is differentiable, thus we can use a stochastic gradient descent method to solve the problem.

The problem (4) is an ill-posed problem [15, 16]. A standard approach to imposing well-posed to a procedure is via the principle of regularization. The conceptual approach of regularization is to look for approximate solutions by setting appropriate smoothness constraints on the hypothesis space \( \mathcal{H} \). In particular, we use the Tikhonov regularization, which consists to replace the minimization problem (4) with the following one.
\[
\min_{f \in \mathcal{H}} \mathcal{R}_{\text{emp}}[f] + \lambda \| f \|_{\mathcal{H}}^2,
\]
where \( \lambda > 0 \) is the regularization parameter.

According to the representer theorem [17], The solution \( f_{\gamma}^{\lambda} \) of the problem (5), can be written as a finite linear combination of kernel evaluations in the data, namely
\[
f_{\gamma}^{\lambda}(x) = \sum_{i=1}^{n} \alpha_i^{\gamma} K(x, x_i),
\]
where \( \alpha_i^{\gamma} \in \mathbb{R}, i = 1, ..., n \) and \( K \) the reproducing kernel of \( \mathcal{H} \). Therefore, the solution to the possibly infinite dimensional optimization problem (5) resides in the \( n \)-dimensional span of the functions \( K_{x_i}(\cdot), i = 1, ..., n \). To compute the coefficients, we simply have to solve the problem
\[
\min_{\alpha \in \mathbb{R}^n} \frac{1}{n} \sum_{i=1}^{n} \left( 1 - \sum_{j=1}^{n} \alpha_j^{\gamma} K(x_i, x_j) y_i + \frac{1}{\gamma} \log \left( 1 + \exp\left( -\frac{\gamma}{\gamma} \left( 1 - \sum_{j=1}^{n} \alpha_j^{\gamma} K(x_i, x_j) y_i \right) \right) \right) \right) + \lambda \| \sum_{j=1}^{n} \alpha_j^{\gamma} K(\cdot, x_j) \|^2.
\]
(7)

Denoting
\[
K = \begin{pmatrix}
K(x_1, x_1) & \cdots & K(x_1, x_n) \\
\vdots & \ddots & \vdots \\
K(x_n, x_1) & \cdots & K(x_n, x_n)
\end{pmatrix}, \quad K_i = \begin{pmatrix}
K(x_i, x_1) \\
\vdots \\
K(x_i, x_n)
\end{pmatrix}, \quad \text{and } y = \begin{pmatrix}
y_1 \\
\vdots \\
y_n
\end{pmatrix}.
\]

Then the objective function is
\[
J_{\gamma}^{\lambda}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} \left( 1 - \alpha^\top K_i y_i + \frac{1}{\gamma} \log \left( 1 + \exp\left( -\frac{\gamma}{\gamma} (1 - \alpha^\top K_i y_i) \right) \right) \right) + \lambda \alpha^\top K \alpha.
\]

Thanks to the differentiability of the objective function of (7), we can use a stochastic gradient descent method to solve this reformulation [18, 19, 20]. The basic version of gradient descent algorithm, is described via the following rule:

1. **Input** initial initial iterate \( \alpha_0 \), step size \( \tau_0 > 0 \).
2. **For** \( t=1, \ldots \) do
3. **Choose** \( i \in \{1, 2, \ldots, n\} \).
4. \( \alpha_{k+1} = \alpha_k - \tau_k \nabla J_{\gamma}^{\lambda,i}(\alpha_k) \).
5. **and For**
6. **return** \( \alpha_k \).
Where $J_{\lambda,i}^\gamma = \left(1 - \alpha^\top K_i y_i + \frac{1}{\gamma} \log \left(1 + \exp(-\gamma(1 - \alpha^\top K_i y_i))\right)\right) + \lambda \alpha^\top K \alpha$.

4. Experimental results

In this section, we present some experimental results showing the efficiency of the proposed approach. In this case, the validation of the numerical experiment is made based on synthetic and real-life data from the UCI machine learning repository.

In sequel, for all considered examples, the Gaussian kernel, given by

$$K(u, v) = \exp \left(-\frac{1}{2\delta^2} \sum_{i=1}^{\ell} (u_i - v_i)^2\right),$$

as basis of the space $\mathcal{H}$, where $\delta$ is a free parameter and $n$ is the size of training set.

First, we choose we validate the effectiveness of the proposed method through academic example. In this example we are interest in approximating a function, defined as follows

$$f(x) = \begin{cases} 
1 & \text{if } x = 0, \\
\text{sign}(x) & \text{otherwise}
\end{cases}$$

First of all, we construct the data sets using a discretization of $[-5, 5]$ by

$$x_j = -5 + j \frac{1}{100}, \quad \text{and} \quad y_j = f(x_j), \quad x_j \in [-5, 5], \quad \text{for } j = 0, \ldots, N \quad \text{and}. \quad (8)$$

We construct our training set by taking 800 random observations from all data sets.

![Figure 3: a) cost function, b) The optimality gap.](image-url)
Moreover, the obtained results in figures 3-4 show that, the proposed method outperforms and the predicted class are really close to the real one with accuracy equal 0.99. These results guarantee us the effectiveness of the proposed method.

In the sequel, we investigate the effectiveness of the proposed method in real world example, available from the UCI Machine Learning Repository called banknote authentication Data Set [21]. The learning data set consists of 1372 objects (“banknotes”) grouped in the two mentioned classes. Data were extracted from images that were taken from genuine and forged banknote-like specimens. For digitization, an industrial camera usually used for print inspection was used. The final images have 400x 400 pixels. Due to the object lens and distance to the investigated object gray-scale pictures with a resolution of about 660 dpi were gained. Wavelet Transform tool were used to extract features from images [22, 23].

For the implementation, we construct our training set by taking 1000 random observations form all data sets. Then by using the proposed approach, we generate the turbulence model. As it can be seen, from Figures 5 and 6, the smoothed hinge loss function predicted the model with a good accuracy compared to square loss function functions. This allow us to say that the smoothed hinge loss function is a good choice for classifications problems.

![Figure 4: Real and predicted class, for \( \delta = 0.5, \gamma = 10 \) and \( \lambda = 10^{-5} \).](image)

Figure 4: Real and predicted class, for \( \delta = 0.5, \gamma = 10 \) and \( \lambda = 10^{-5} \).

![Figure 5: a) cost function, b) The optimality gap.](image)

Figure 5: a) cost function, b) The optimality gap.
5. Conclusion
In this paper we have presented a framework for solving support vector machine with kernel (SVMK). Based on hinge loss function combined with a special smoothing technique the original problem is formulated into equivalent smoothing one. Thus the obtained smooth problem combined with Tikhonov regularization is solved using a stochastic gradient descent method. Finally, some numerical experiments on academic and real-life datasets is presented to show the efficiency of the proposed approach. As perspective, we use other smooth technique, for both regression and classification problems.

References
[1] S. Lyaqini, M. Quafafou, M. Nachaoui, A. Chakib, Supervised learning as an inverse problem based on non-smooth loss function, Knowledge and Information Systems (2020) 1–20.
[2] W. Shen, Z. Wei, Q. Li, H. Zhang, D. Miao, Three-way decisions based blocking reduction models in hierarchical classification, Inform. Sci. 523 (2020) 63–76.
[3] Y.-J. Lee, W.-F. Hsieh, C.-M. Huang, epsilon-svr: a smooth support vector machine for epsilon-insensitive regression, IEEE Transactions on Knowledge & Data Engineering (5) (2005) 678–685.
[4] J. Hajewski, S. Oliveira, D. Stewart, Smoothed hinge loss and $\ell^1$ support vector machines, in: 2018 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2018, pp. 1217–1223.
[5] M. Nachaoui, Parameter learning for combined first and second order total variation for image reconstruction, Advanced Mathematical Models & Applications 5 (1) (2020) 53–69.
[6] S. Lu, S. V. Pereverzev, Regularization theory for ill-posed problems, Vol. 58 of Inverse and Ill-posed Problems Series, De Gruyter, Berlin, 2013, selected topics.
[7] K. A. Berdawood, A. Nachaoui, R. Saeed, M. Nachaoui, F. Aboud, An alternating procedure with dynamic relaxation for cauchy problems governed by the modified helmholtz equation, Advanced Mathematical Models & Applications 5 (1) (2020) 131–139.

[8] P. Dvalishvili, A. Nachaoui, M. Nachaoui, T. Tadumadze, On the well-posedness of the cauchy problem for a class of differential equations with distributed delay and the continuous initial condition, Vol. 43, Inst Mathematics & Mechanics, Natl Acad Sciences Azerbaijan 9 B Vahabzdeh , 2017, pp. 146–160.

[9] A. Nachaoui, M. Nachaoui, A. Chakib, M. A. Hilal, Some novel numerical techniques for an inverse Cauchy problem, J. Comput. Appl. Math. 381 (2021) 113030, 21.

[10] A. Nachaoui, M. Nachaoui, Iterative methods for forward and inverse bioelectric field problem, in: International Conference on Applied Mathematics, Modeling and Life Sciences, Icamls 18, 2018.

[11] C. Chen, O. L. Mangasarian, A class of smoothing functions for nonlinear and mixed complementarity problems, Computational Optimization and Applications 5 (2) (1996) 97–138.

[12] L. Rosasco, E. De Vito, A. Caponnetto, M. Piana, A. Verri, Are loss functions all the same?, Neural Comput. 16 (2004) 1063–1076.

[13] N. Aronszajn, Theory of reproducing kernels, Transactions of the American Mathematical Society 68 (1950) 337–404.

[14] R. Rosales, M. Schmidt, G. Fung, Fast optimization methods for l1 regularization: A comparative study and two new approaches.

[15] J. Hadamard, Lectures on Cauchy’s problem in linear partial differential equations, New Haven: Yale University Press, 1923.

[16] F. Girosi, M. Jones, T. Poggio, Regularization theory and neural networks architectures, Neural computation 7 (2) (1995) 219–269.

[17] B. Schölkopf, R. Herbrich, A. J. Smola, A generalized representer theorem, in: International conference on computational learning theory, Springer, 2001, pp. 416–426.

[18] S. Mandt, M. D. Hoffman, D. M. Blei, Stochastic gradient descent as approximate bayesian inference, The Journal of Machine Learning Research 18 (1) (2017) 4873–4907.

[19] E. Yazan, M. F. Talu, Comparison of the stochastic gradient descent based optimization techniques (2017) 1–5.

[20] H. Kasai, Sgdlibrary: A matlab library for stochastic optimization algorithms, JMLR, Journal of Machine Learning Research 19.

[21] http://archive.ics.uci.edu/ml/datasets/yacht2bhydrodynamics.

[22] E. Kaya, A. Yasar, I. Saritas, Banknote classification using artificial neural network approach, training 10 (2015) 1.

[23] J. Jdrzejowicz, P. Jdrzejowicz, Gene expression programming ensemble for classifying big datasets (2017) 3–12.