# Borderline kernel based over-sampling

M. Pérez-Ortiz[1], P.A. Gutiérrez[1] and C. Hervás-Martínez[1] *

University of Córdoba, Dept. of Computer Science and Numerical Analysis
Rabanales Campus, Albert Einstein building, 14071 - Córdoba, Spain

**Abstract.** Nowadays, the imbalanced nature of some real-world data is receiving a lot of attention from the pattern recognition and machine learning communities in both theoretical and practical aspects, giving rise to different promising approaches to handling it. However, preprocessing methods operate in the original input space, presenting distortions when combined with kernel classifiers, that operate in the feature space induced by a kernel function. This paper explores the notion of empirical feature space (a Euclidean space which is isomorphic to the feature space and therefore preserves its structure) to derive a kernel-based synthetic over-sampling technique based on borderline instances which are considered as crucial for establishing the decision boundary. Therefore, the proposed methodology would maintain the main properties of the kernel mapping while reinforcing the decision boundaries induced by a kernel machine. The results show that the proposed method achieves better results than the same borderline over- sampling method applied in the original input space.

## 1 Introduction

Imbalanced classification is one of the current challenges for machine learning [1, 2], since it has been shown to hinder the learning performance of classification algorithms. Imbalanced classification problems are very common in many real-world domains, such as medical diagnosis, text categorization, fraud detection or information retrieval, contexts where usually the minority class happens to be more interesting than the majority one, but also more difficult to model due to the low number of available patterns. Since most traditional learning systems have been designed to work on balanced data, they will usually be focused on improving overall performance and be biased towards the majority class, consequently harming the minority one [3]. To cope with this issue, several algorithms have been designed over the years to over-sample minority samples and to under-sample the majority ones, the Synthetic Minority Over-sampling Technique [1] (SMOTE) being one of the most representatives for the first group, among others.

* This work has been partially subsidized by the TIN2011-22794 project of the Spanish Ministerial Commission of Science and Technology (MICYT), FEDER funds and the P08-TIC-3745 project of the "Junta de Andalucía" (Spain).

In the context of kernel classifiers [4], since the very first introduction of the support vector machine paradigm (Support Vector Classifier or SVC), we have witnessed a huge development in theory and methodologies of what is known as kernel-based methods: advances in performance theory, different variants of kernel classifiers and regressors, algorithms for feature selection and extraction, all that accompanied by countless successful applications. Moreover, the success of kernel methods can be attributed to the joint use of a robust classification procedure (such as the large margin hyperplane principle) and a convenient and versatile way of preprocessing the patterns (the kernel trick). However, very little has been done in the context of imbalanced classification, and more specifically, concerning over-sampling in the feature space. This is essentially the main aim of this paper because when these classifiers are combined with other preprocessing techniques which operate in the input space, some obvious distortions are found, given that they operate in different spaces. The ideal approach would be preprocess the training patterns in the feature space, although this is not possible since the only information available is the dot products of their images. To deal with this issue, this paper makes use of the notion of empirical feature space [5, 6], which preserves the geometrical structure of the original feature space, given that distances and angles in the feature space are uniquely determined by dot products and that the dot products of the corresponding images are the original kernel values. This empirical feature space is Euclidean, so it provides a tractable framework to study the spatial distribution of the mapping function $\Phi(\cdot)$ [7], to measure class separability [6] and to optimize the kernel [6, 8]. Besides, the notion of empirical kernel feature space has been used for the kernelization of all kinds of linear classifiers [9, 10], with the advantage that the algorithm does not need to be formulated to deal with dot products.

Therefore, the main aim of this paper is to check whether the empirical feature space provides a more suitable space than the input space for performing over-sampling. This Euclidean space is isomorphic to the feature space, hence we hypothesize that the synthetic patterns generated will be better adapted to the kernel machine classifier. Borderline over-sampling [11] has been chosen for the experimentation since we consider that borderline examples are more informative for a large margin based classifier such as SVM (this borderline area is more crucial for establishing the decision boundary) and also most prone to be misclassified. Indeed, performing over-sampling on this area has been demonstrated to make more benefit than performing it on the whole minority class [11, 12]. For this purpose, an efficient way of selecting informative instances from the pool of samples is also needed, this step being usually computed in the input space, rather than in the feature one, which is also one of the hypotheses of the paper: that, for a kernel machine, borderline patterns will be better chosen in the feature space that in the input space, given that the kernel machine operates in this feature space.

The idea of over-sampling in the feature space have been also researched in [13], where synthetic instances were generated by using the geometric interpretation of the dot products in the kernel matrix, and the pre-images of these

synthetic instances were approximated based on a distance relation between the feature space and the input one, since inverse mapping $\Phi(\cdot)^{-1}$ from the feature space to input space is not available. Finally, the approximation of these pre-images are appended to the original dataset to train a SVM. Note that in our case, the over-sampling is performed in the empirical feature space, thus our methodology is free of the computational cost and assumptions of this inverse mapping approximation.

The paper is organized as follows: Section 2 shows a description of the methodology used; Section 3 describes the experimental study and analyses the results obtained; and finally, Section 4 outlines some conclusions.

## 2 Methodology

The goal in binary classification could be said to assign an input vector $\mathbf{x} \in \mathbb{R}^d$ to one of the classes $\mathcal{C}_+$ and $\mathcal{C}_-$ (corresponding this labelling to the output space $\mathcal{Y}$). The objective is to find a prediction rule $f : \mathcal{X} \to \mathcal{Y}$ by using an i.i.d. training sample $D = \{\mathbf{x}_i, y_i\}_{i=1}^m \in \mathcal{X} \times \mathcal{Y}$. The methodology here proposed is based on performing over-sampling in the empirical feature space using the patterns on the boundary of the minority class. Consequently, the notion of empirical feature space is firstly described. Then, we describe how to extend the borderline SMOTE algorithm to better handle imbalanced datasets when applying kernel classifiers.

### 2.1 Empirical feature space

In this section, the empirical feature space spanned by the training data is defined. Let $\mathcal{H}$ denote a high-dimensional or infinite-dimensional Hilbert space. Then, for any mapping of patterns $\Phi : \mathcal{X} \to \mathcal{H}$, the inner product $\mathcal{K}(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle_{\mathcal{H}}$ of the mapped inputs is known as a kernel function, giving rise to a symmetric and positive semidefinite matrix (known as Gram or kernel matrix $\mathbf{K}$) from a given input set $\mathcal{X}$. By definition, these matrices can be diagonalised as follows:

$$\mathbf{K}_{(m\times m)} = \mathbf{P}_{(m\times r)} \cdot \mathbf{M}_{(r\times r)} \cdot \mathbf{P}^{\mathrm{T}}_{(r\times m)}, \tag{1}$$

where $(\cdot)^{\mathrm{T}}$ is the transpose operation, $\mathbf{M}$ is a diagonal matrix containing the $r$ positive eigenvalues of $\mathbf{K}$ in decreasing order, and $\mathbf{P}$ consists of the eigenvectors associated to those $r$ eigenvalues. The empirical feature space is a Euclidean space preserving the dot product information about $\mathcal{H}$ contained in $\mathbf{K}$. The mapping from the input space to a $r$-dimensional empirical feature space can be defined as $\Phi_r^e : \mathcal{X} \to \mathbb{R}^r$, where $r$ is the rank of $\mathbf{K}$. This space is isomorphic to the embedded feature space $\mathcal{H}$, but presents all the advantages of being Euclidean:

$$\Phi_r^e : \mathbf{x}_i \to \mathbf{M}^{-1/2} \cdot \mathbf{P}^{\mathrm{T}} \cdot (\mathcal{K}(\mathbf{x}_i, \mathbf{x}_1), \ldots, \mathcal{K}(\mathbf{x}_i, \mathbf{x}_m))^{\mathrm{T}}. \tag{2}$$

It is easy to check that the kernel matrix of the training images obtained by this transformation is $\mathbf{K}$, when considering the standard dot product [5, 6]. Note that

this transformation corresponds to the principal component analysis *whitening* step [14], although applied to the kernel matrix, instead of the covariance matrix. Although the whole set of all $r$ positive eigenvalues could be considered, a smaller set (in this case, for simplicity, a 10-dimensional set) has been chosen in this paper by choosing the $p$ dominant eigenvalues and their associated eigenvectors. The choice of this smaller set limits the dimensionality of the empirical feature space and make more robust the process of over-sampling by simplifying the space, given the concentration of spectral measures.

Fig. 1 shows the case of a synthetic dataset concerning a non-linearly separable classification task and its transformation to the two-dimensional empirical feature space induced by the well-known standard Gaussian kernel, which is linearly separable.

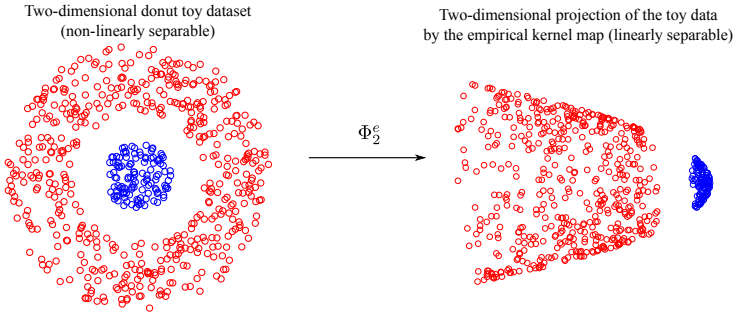


**Fig. 1.** Synthetic two-dimensional dataset representing a non-linearly separable classification problem and its transformation to the 2 dominant dimensions of the empirical feature space induced by the Gaussian kernel function (linearly separable problem).

### 2.2 Borderline over-sampling in the empirical feature space

The main idea for the proposed method is to use the empirical feature space to apply preprocessing algorithms, because preprocessed patterns would better suit the kernel machine classifier later considered. In this paper, the borderline SMOTE algorithm was selected to decrease the problems caused by imbalanced datasets when applying a kernel classifier.

Borderline over-sampling [11] is based on the idea of generating new synthetic patterns on the borderline between different classes, as these patterns are considered as being more probable to be misclassified. Thus, the first step corresponds to the identification of these patterns that are "in danger" of being misclassified, which is usually done by examining the neighborhood of the pattern considered, e.g. if all the nearest neighbors correspond to the minority class, the pattern is not considered as a borderline example, however, if half of the nearest neighbors belong to the minority class and the other half to the majority one, the pattern can be considered as a borderline one. Finally, borderline examples are the ones

considered for generating new synthetic patterns by means of the well-known SMOTE technique [1]. Therefore, when considering the empirical feature space rather than the original input one, not only the process of generating new examples change as the space used is different, but also the patterns chosen as borderline.

Concerning the proposed method, first of all, the empirical feature space induced by a kernel function $\mathcal{K}$ in the training set is computed. Formally, $\mathbf{T}^e_{(m\times r)}$ is the matrix generated by applying the empirical kernel map $\Phi^e_r$ (see (2)) to the training patterns. Then, the standard borderline SMOTE algorithm [11] is applied over the class images of this $\mathbf{T}^e$ matrix, resulting in the generation of $n$ new synthetic images of patterns, arranged in the matrix $\mathbf{S}^e_{(n\times r)}$ (note that all these new patterns will belong to the minority class). The new synthetic examples will be used to complete the kernel matrix, obtaining their dot product with respect to the rest of training patterns, i.e. $\mathbf{KS}^e_{i,j} = \mathbf{T}^e_i \cdot \mathbf{S}^e_j, 1 \leq i \leq m, 1 \leq j \leq n$, and with respect to themselves $\mathbf{SS}^e_{i,j} = \mathbf{S}^e_i \cdot \mathbf{S}^e_j, 1 \leq i, j \leq n$, where $\mathbf{T}^e_i$ is the empirical space representation of the $i$-th training pattern, and $\mathbf{S}^e_i$ is the $i$-th synthetic sample previously generated. Using these matrices, the over-sampled training Gram matrix $\mathbf{K}^*$ will be composed as follows:

$$\mathbf{K}^*_{(m+n)\times(m+n)} = \begin{pmatrix} \mathbf{K}_{(m\times m)} & \mathbf{KS}^e_{(m\times n)} \\ \left(\mathbf{KS}^e_{(m\times n)}\right)^{\mathrm{T}} & \mathbf{SS}_{(n\times n)} \end{pmatrix}, \tag{3}$$

where $\mathbf{K}$ is the original kernel matrix. For the generalization phase, the same steps are considered to complete the test kernel matrix, taking into account that the empirical feature space images of the test patterns are derived using the same $\Phi^e_r$ transformation (considering only the training data). Fig. 2 shows the main steps of the proposed algorithm: Borderline Kernel SMOTE (BKS).

**Algorithm BKS**

- **Input**: Training patterns (**Tr**) and training targets (**Trg**).
- **Output**: Over-sampled training kernel matrix ($\mathbf{K}^*$).
  1. Compute kernel matrix $\mathbf{K}$ for training patterns.
  2. Compute the empirical kernel map $\Phi^e_r$ via $\mathbf{K}$.
  3. Map training patterns to the empirical feature space using $\Phi^e_r$ ($\mathbf{T}^e$).
  4. Apply borderline SMOTE with the new representation $\mathbf{T}^e$ of the training patterns and obtain a new set $\mathbf{S}^e$ of synthetic data.
  5. Complete the over-sampled kernel matrix $\mathbf{K}^*$ with the new synthetic patterns and their dot product according to (3).

**Fig. 2.** Different steps for the kernel over-sampling algorithm proposed.

Given that the over-sampling technique operates in $r$ dimensions (kernel matrix rank), instead of $d$ (dimensionality of the input space), what is noteworthy is the applicability of the proposed method to bioinformatics datasets where

the number of features tend to be much higher than the number of samples ($r << d$), and where imbalanced datasets are commonly found. Additionally, as an advantage of the method, there is no need to treat the data attributes differently (taking into account their nature) since all of them are real, unlike in the original SMOTE.

As a final remark, in order to clarify the usefulness of performing the over-sampling in the feature space, let us analyze the case presented in Fig. 3, where a toy non-linearly separable dataset has been represented. The top part of the figure corresponds to the synthetic dataset created and its transformation via the empirical kernel map, while the bottom part includes information about the 5-nearest neighbors for each pattern. From this figure, one can appreciate that despite the fact that $k$-nearest neighbors is a nonlinear methodology, it is very sensitive to the correct choice of $k$, in such a way that we could be generating new synthetic patterns in an inappropriate region (as the bottom left plot where the over-sampling is generated in the input space). However, if we consider the empirical feature space instead (as in the right part of the figure), the over-sampling is less sensitive to the choice of $k$, since, in this space, the separation between the patterns is easier (ideally, linearly separable), which is one of the main characteristics of the kernel trick. Note that the representation of the empirical feature space plotted in the right part of the figure is only a two-dimensional approximation, thus we are obviating useful information.

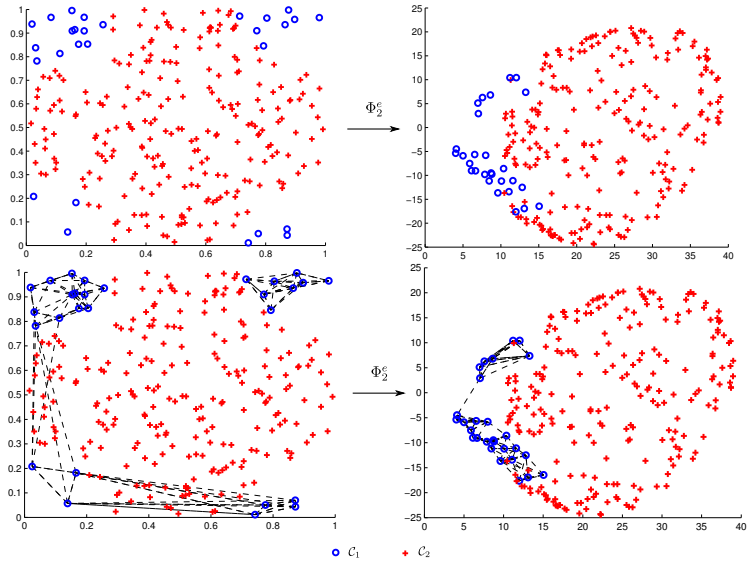


**Fig. 3.** Toy two-dimensional non-linearly separable dataset and the transformation to the 2 dominant dimensions of the empirical feature space induced by the Gaussian kernel function. Dashed lines represent the 5-nearest neighbors of each pattern belonging to the minority class.

**Table 1.** Characteristics of the benchmark datasets used for the experimentation (number of patterns, features and imbalance ratio (IR)).

| Dataset | Patterns | Features | IR |
|---|---|---|---|
| liver | 345 | 6 | 1.38 |
| bands | 365 | 19 | 1.70 |
| vehicle1 | 846 | 18 | 2.90 |
| ecoli1 | 336 | 7 | 3.36 |
| ecoli2 | 336 | 7 | 5.46 |
| glass6 | 214 | 9 | 6.38 |
| yeast0359-78 | 506 | 8 | 9.12 |
| vowel0 | 988 | 13 | 9.98 |
| yeast1-7 | 459 | 8 | 14.30 |
| yeast1289-7 | 947 | 8 | 30.57 |

All nominal variables are transformed into binary ones

## 3 Experimental results

The proposed method has been tested considering the Support Vector Classifier (SVC) [15] and the well-known borderline SMOTE [11]. Our methodology (Borderline Kernel SMOTE, BKS) is compared to the original borderline SMOTE in the input space (BS), and to the results without over-sampling. 10 binary benchmark datasets from the UCI repository [16] with different imbalance ratios (proportion of majority patterns with respect to minority ones) have been tested to analyze the performance of the methods in different situations. The characteristics of these datasets can be seen in Table 1. As done in other over-sampling state-of-the-art works [3], some multiclass datasets have also been considered by grouping some classes, e.g. ecoli1 represents the ecoli dataset when considering class 1 versus the rest, and yeast0359-78 is the yeast dataset when grouping classes 0, 3, 5, and 9 versus classes 7 and 8 in order to obtain higher imbalance ratio values.

A stratified 5-fold technique was performed to divide the data and the results are taken as mean and standard deviation of the selected measures. The Gaussian kernel was used. The kernel width and the cost parameter of SVC was selected within the values $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$, by means of a nested 5-fold method applied to the training set. The number of synthetic patterns generated was that needed to balance the distributions, i.e. after applying the over-sampling process, the number of majority and minority patterns were the same.

The results have been reported in terms of three metrics, two of them specially designed to deal with imbalanced datasets:

1. The well-known Accuracy metric ($Acc$), which corresponds to the ratio of correctly classified patterns and measures overall performance.
2. The Geometric Mean of the sensitivities ($GM = \sqrt{S_{\mathrm{p}} \cdot S_{\mathrm{n}}} \cdot 100$), where $S_{\mathrm{p}}$ is the sensitivity for the positive class (ratio of correctly classified patterns considering only this class) and $S_{\mathrm{n}}$ is the sensitivity for the negative one.
3. The Minimum Sensitivity [17] ($MS = \min\{S_p, S_n\} \cdot 100$), which can be defined as the minimum value of the sensitivities for each class.

**Table 2.** Results achieved by the three methods considered for the different metrics.

| Dataset | Algorithm | $Acc(\%)$ | $GM(\%)$ | $MS(\%)$ |
|---|---|---|---|---|
| liver | SVC | *71.03 ± 8.05* | *68.28 ± 8.59* | 58.57 ± 10.41 |
| | BS+SVC | 69.86 ± 6.68 | 68.27 ± 6.22 | *61.88 ± 6.43* |
| | BKS+SVC | **71.30 ± 9.28** | **70.06 ± 9.45** | **64.21 ± 10.62** |
| bands | SVC | **71.76 ± 4.61** | *66.46 ± 8.15* | *55.49 ± 14.17* |
| | BS+SVC | *71.49 ± 5.50* | 65.95 ± 10.25 | 55.33 ± 16.50 |
| | BKS+SVC | 70.11 ± 6.85 | **68.63 ± 9.53** | **61.56 ± 12.94** |
| vehicle1 | SVC | *85.34 ± 4.13* | 80.35 ± 7.85 | 72.32 ± 12.68 |
| | BS+SVC | **86.05 ± 1.72** | *83.48 ± 1.70* | *78.58 ± 3.99* |
| | BKS+SVC | 83.10 ± 2.09 | **84.48 ± 2.52** | **81.55 ± 1.94** |
| ecoli1 | SVC | **90.20 ± 4.94** | *85.38 ± 5.82* | *77.75 ± 8.95* |
| | BS+SVC | 87.52 ± 4.08 | 84.52 ± 6.75 | 77.15 ± 11.51 |
| | BKS+SVC | *90.18 ± 2.92* | **86.45 ± 3.76** | **80.38 ± 7.45** |
| ecoli2 | SVC | *94.95 ± 2.23* | 90.55 ± 3.07 | 84.73 ± 4.87 |
| | BS+SVC | 94.94 ± 2.26 | *93.03 ± 5.22* | *89.14 ± 7.98* |
| | BKS+SVC | **97.02 ± 2.11** | **95.11 ± 3.98** | **91.84 ± 6.94** |
| glass6 | SVC | *95.32 ± 2.88* | 85.73 ± 9.39 | 75.33 ± 16.26 |
| | BS+SVC | 93.44 ± 5.58 | *86.33 ± 12.02* | *78.13 ± 18.96* |
| | BKS+SVC | **95.82 ± 6.31** | **92.48 ± 14.52** | **87.78 ± 21.88** |
| yeast0359-78 | SVC | **87.54 ± 5.81** | 50.88 ± 13.05 | 30.00 ± 15.81 |
| | BS+SVC | *79.05 ± 3.56* | *64.18 ± 11.21* | *49.38 ± 15.99* |
| | BKS+SVC | 70.93 ± 10.35 | **66.72 ± 7.24** | **57.74 ± 11.86** |
| vowel0 | SVC | **100.00 ± 0.00** | **100.00 ± 0.00** | **100.00 ± 0.00** |
| | BS+SVC | *99.90 ± 0.23* | *99.94 ± 0.12* | *99.89 ± 0.25* |
| | BKS+SVC | **100.00 ± 0.00** | **100.00 ± 0.00** | **100.00 ± 0.00** |
| yeast1-7 | SVC | **94.12 ± 1.81** | 48.30 ± 27.42 | 30.00 ± 18.26 |
| | BS+SVC | *84.99 ± 5.44* | *69.14 ± 26.45* | *59.61 ± 24.73* |
| | BKS+SVC | 81.94 ± 7.44 | **77.07 ± 12.66** | **67.98 ± 14.47** |
| yeast1289-7 | SVC | **97.25 ± 0.69** | 45.30 ± 27.43 | 26.67 ± 19.00 |
| | BS+SVC | *81.62 ± 5.60* | *63.55 ± 16.73* | *51.53 ± 25.12* |
| | BKS+SVC | 79.39 ± 7.89 | **69.71 ± 10.66** | **60.60 ± 18.80** |

The best method is in **bold** face and the second one in *italics*

**Table 3.** Mean and ranking values obtained for each methodology and measure.

| | SVC | | BS+SVC | | BKS+SVC | |
|---|---|---|---|---|---|---|
| Measure | Mean | Rank | Mean | Rank | Mean | Rank |
| $Acc$ | **88.75** | **1.45** | *84.88* | 2.4 | 83.98 | *2.15* |
| $GM$ | 72.12 | 2.55 | *77.84* | *2.4* | **81.07** | **1.05** |
| $MS$ | 61.08 | 2.65 | *70.06* | *2.3* | **75.36** | **1.05** |

The best method is in **bold** face and the second one in *italics*

The measure considered during the hyperparameter selection was $GM$, given its robustness for imbalanced datasets. All the test results of these experiments can be seen in Table 2 and the mean and rankings of these results in Table 3.

From the results obtained, several conclusions can be drawn. Firstly, the good performance of the proposed method can be seen analyzing $GM$ and $MS$ measures, where it can be seen that the application of the over-sampling technique in the empirical feature space outperforms the results achieved when applying it in the original input space. Indeed, the ranking of these measures for the SVC and BS+SVC algorithms are similar, indicating that the use of an over-sampling technique in the original input space may not incorporate enough useful information for a kernel machine. Furthermore, although standard deviations corresponding

to $GM$ and $MS$ are high, due to the drastic nature of these measures, in most of the cases, standard deviations of BKS are lower than the ones associated with BS. Concerning $Acc$, the proposed method achieves comparable results to those obtained by the other methods (especially for low IR values). However, one can appreciate that in some cases deteriorating the classification of the majority class (and therefore the overall performance) is needed in order to classify correctly the minority one (this is the case of the datasets yeast0359-78, yeast1-7 and yeast1289-7). With concern to very low IR values (the case of liver and bands datasets), the over-sampling proposed algorithm do not deteriorate the SVC solution and is even able to obtain better values for $GM$ and $MS$. Finally, for the vowel0 dataset, it can be seen that the application of BS is not successful, since the original SVC obtains an optimal solution that is not found when performing the over-sampling in the input space. However, when performing the over-sampling in the feature space induced by the kernel, the performance of the classifier is not deteriorated.

To quantify whether a statistical difference exists among the algorithms compared, the non-parametric Friedman's test [18] (with $\alpha = 0.05$) has been applied to the mean rankings for the three measures considered, rejecting the null-hypothesis that all algorithms perform similarly for $GM$ and $MS$, and accepting it for $Acc$. The confidence interval was $C_0 = (0, F_{(\alpha=0.05)} = 3.55)$, and the corresponding F-value was $2.88 \in C_0$, $19.35 \notin C_0$ and $21.77 \notin C_0$ for $Acc$, $GM$ and $MS$, respectively. Furthermore, the Nemenyi test has also been applied concluding that there are statistically significant differences for $\alpha = 0.05$ in $GM$ and $MS$ (the Nemenyi critical difference being 1.04782) when comparing BKS+SVC with SVC (with ranking differences of 1.5 and 1.6, respectively) and with BS+SVC (with ranking differences of 1.35 and 1.25, respectively).

## 4 Conclusions and future work

This paper explores the idea of performing over-sampling in the class boundary of the empirical feature space related to a kernel function. We focus on the imbalanced binary classification paradigm, and the proposed method has been tested with the standard Support Vector Classifier and the borderline SMOTE algorithm, achieving better results than when applying the same preprocessing in the original input space, specially for metrics designed for imbalanced classification. As future work, the performance of different kernel functions for performing kernel over-sampling could be studied to analyze the kernel function to use according to the nature of the data. Furthermore, in the same vein as this paper, an analytical methodology [19] could be used to compute the number of relevant dimensions for the empirical feature space (note that in our case this value was prefixed for the sake of simplicity).

## References

1. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research **16**

(2002) 321–357
2. Tang, Y., Zhang, Y.Q., Chawla, N.V., Krasser, S.: SVMs modeling for highly imbalanced classification. IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics **39**(1) (February 2009) 281–288
3. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **42**(4) (2012) 463–484
4. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press (2001)
5. Schölkopf, B., Mika, S., Burges, C.J.C., Knirsch, P., Müller, K.R., Rätsch, G., Smola, A.J.: Input space versus feature space in kernel-based methods. IEEE Transactions on Neural Networks **10** (1999) 1000–1017
6. Xiong, H., Swamy, M.N.S., Ahmad, M.O.: Optimizing the kernel in the empirical feature space. IEEE Transactions on Neural Networks **16**(2) (2005) 460–474
7. Yan, F., Mikolajczyk, K., Kittler, J., Tahir, M.A.: Combining multiple kernels by augmenting the kernel matrix. In: Proc. of the 9th International Workshop on Multiple Classifier Systems (MCS). Volume 5997., Springer (2010) 175–184
8. Xiong, H., Swamy, M.N.S., Ahmad, M.O.: Learning with the optimized data-dependent kernel. In: Proc. of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW). Volume 6., IEEE Computer Society (2004) 95–
9. Abe, S., Onishi, K.: Sparse least squares support vector regressors trained in the reduced empirical feature space. In: Proc. of the 17th international conference on Artificial neural networks. ICANN, Springer-Verlag (2007) 527–536
10. Xiong, H.: A unified framework for kernelization: The empirical kernel feature space. In: Chinese Conference on Pattern Recognition (CCPR). (nov. 2009) 1 –5
11. Han, H., Wang, W.Y., Mao, B.H.: Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. Advances in Intelligent Computing (2005) 878–887
12. Wang, H.Y.: Combination approach of smote and biased-svm for imbalanced datasets. (2008)
13. Zeng, Z.Q., Gao, J.: Improving svm classification with imbalance data set. In: Proc. of the 16th International Conference on Neural Information Processing: Part I. ICONIP '09, Berlin, Heidelberg, Springer-Verlag (2009) 389–398
14. Schölkopf, B., Smola, A., Müller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation **10**(5) (1998) 460–474
15. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning **20**(3) (1995) 273–297
16. Asuncion, A., Newman, D.: UCI machine learning repository (2007)
17. Fernández-Caballero, J.C., Martínez-Estudillo, F.J., Hervás-Martínez, C., Gutiérrez, P.A.: Sensitivity versus accuracy in multiclass problems using memetic pareto evolutionary neural networks. IEEE Transactions on Neural Networks **21**(5) (2010) 750–770
18. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7** (2006) 1–30
19. Braun, M.L., Buhmann, J.M., Müller, K.R.: On relevant dimensions in kernel feature spaces. J. Mach. Learn. Res. **9** (June 2008) 1875–1908