Pertanika J. Sci. & Technol. 27 (1): 527 - 545 (2019)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Performance of SVM with Multiple Kernel Learning for Classification Tasks of Imbalanced Datasets

Sana Saeed[1]* and Hong Choon Ong[2]

[1]College of Statistical and Actuarial Sciences, University of the Punjab, 54590, Lahore, Pakistan
[2]School of Mathematical Sciences, Institute of Post Graduate Studies, University Sains Malaysia, 11800 USM, Gelugor, Penang, Malaysia

ABSTRACT

Support vector machine (SVM) is one of the most popular algorithms in machine learning and data mining. However, its reduced efficiency is usually observed for imbalanced datasets. To improve the performance of SVM for binary imbalanced datasets, a new scheme based on oversampling and the hybrid algorithm were introduced. Besides the use of a single kernel function, SVM was applied with multiple kernel learning (MKL). A weighted linear combination was defined based on the linear kernel function, radial basis function (RBF kernel), and sigmoid kernel function for MKL. By generating the synthetic samples in the minority class, searching the best choices of the SVM parameters and identifying the weights of MKL by minimizing the objective function, the improved performance of SVM was observed. To prove the strength of the proposed scheme, an experimental study, including noisy borderline and real imbalanced datasets was conducted. SVM was applied with linear kernel function, RBF kernel, sigmoid kernel function and MKL on all datasets. The performance of SVM with all kernel functions was evaluated by using sensitivity, G Mean, and F measure. A significantly improved performance of SVM with MKL was observed by applying the proposed scheme.

Keywords: Hybrid algorithm, imbalanced datasets, multiple kernel learning, oversampling algorithm, support vector machine

ARTICLE INFO

Article history:
Received: 21 March 2018
Accepted: 10 July 2018
Published: 24 January 2019

E-mail addresses:
sana.aqeel2010@gmail.com (Sana Saeed)
hcong@usm.my (Hong Choon Ong)
* Corresponding author

INTRODUCTION

SVM is a popular supervised machine learning algorithm, successfully handling classification tasks in many real-world applications: for example, credit scoring, text classification and bankruptcy prediction (Chaudhuri & De, 2011; Shin et al., 2005;

ISSN: 0128-7680
e-ISSN: 2231-8526

Sun et al., 2009; Huang et al., 2007). SVM has a strong theoretical and mathematical background and a high generalization capability of finding the global and non-linear classification solutions (Ben-Hur & Weston, 2010). However, its performance becomes limited for the classification task of imbalanced noisy and borderline datasets (Imam et al., 2006; Eitrich & Lang, 2006).

As the optimal performance of SVM is based on the decent choices of its parameters with kernel settings, the model selection problem essentially includes the search for the best values of the slack variable penalty weight (C) and kernel parameters, which are supposed to be used for the classification task. Traditionally, a grid search selection is adopted. Nevertheless, this method is time-consuming and does not produce the desired results (Hsu & Lin, 2002; Hsu et al., 2003). The present study is based on this prevailing issue of SVM. The current research work will address and discuss the issues of SVM for binary imbalanced datasets. Besides the use of a single kernel function, the current study will examine the performance of SVM with MKL. This study aims to present a new scheme for the said task.

Primary research works in MKL from the perspective of optimization techniques, can be seen from an interesting and brief research work presented by Lanckriet et al. (2004). The authors proposed an idea of using semidefinite programming (SDP) for kernel learning. Since SDP is a convex form of optimization and avoids being trapped in local optima, a transductive algorithm was offered. Furthermore, a learning method of SVM parameters was discussed. Afterward, an idea of using semi-infinite programming for the conic combination of kernels was proposed by Sonnenburg et al. (2006). The authors suggested the use of evolutionary approach, genetic algorithm (GA) in determining the weights of combined kernels of SVM. The authors gave a new direction to kernel learning by embedding the metaheuristic algorithms in it. Zhang (2006) proposed kernel optimization for SVM using the Levenberg-Marquardt (LM) algorithm instead of using the gradient descent approach to test the protein location data from yeast. Linear combinations of kernels for SVM in regards to speaker verification were used by Dehak et al. (2008). The combination weights were speaker dependent as compared to the universal weights on score level fusion. Another comprehensive study on the linear and nonlinear combinations of kernels was conducted by Cortes et al. (1995).

Cao et al. (2013) proposed an idea of optimized cost-sensitive SVM. An effective wrapper framework incorporating the area under the curve (AUC) and G Mean into the objective function of SVM was introduced to gain a better performance of SVM. A subset of feature selection, parameters, and misclassification cost were simultaneously optimized. Jiang et al. (2014) presented an idea for the optimal selection of SVM parameters by using three metrics namely AUC, accuracy and balanced accuracy using computational data. The authors engaged different levels of separability, different levels of imbalances and different levels of training sets in the study.

To enhance the performance of SVM for imbalanced datasets, different resampling approaches were also proposed by researchers. Most of them suggested the use of the oversampling technique, synthetic minority oversampling technique (SMOTE) in combination with SVM. Different kinds of sampling techniques were proposed; for example, a combined sampling approach using SMOTE and Tomek link with SVM for binary classification, a hybrid sampling approach using under and oversampling, and an ensemble method i.e. bagging of extrapolation borderline SMOTE (BEBS), which all can be studied from the available literature (Sain & Purnami, 2015; Wang, 2014; Wang et al., 2017).

Due to the emerging use of metaheuristic techniques for optimization problems, the use of these techniques for SVM can also be justified. For example, GA based feature selection and parameter optimization procedure for SVM can be found in the literature (Wang et al., 2011). GA and particle swarm optimization (PSO) for SVM and ant colony optimization (ACO) for SVM model selection can also be seen in the available studies on SVM optimization (Alwan & Ku-Mahamud, 2013; Ren & Bai, 2010; Blondin & Saad, 2010). An efficient memetic algorithm based on PSO and pattern search (PS) was proposed for SVM parameter optimization. PSO was used for the exploration purpose while PS was applied for exploitation (Bao et al., 2013). Another study proposed a combination of optimization and classification algorithms for SVM by using SMOTE and PSO (Cervantes et al., 2017). Wu et al. (2017) applied two-phase sequential minimal optimization (TSMO) and differential learning particle swarm optimization (DPSO) for SVM.

This article is organized as follows: material and methods are provided in Section 2. In Section 3, the proposed scheme is presented. Results of the experimental studies are given in Section 4. The conclusion is discussed in Section 5.

MATERIAL AND METHODS

According to Abe (2005), SVM as a nonlinear classifier can offer a better precision in many real-world applications. The process of making linear classifiers become nonlinear is to map the data from input space *X* to feature space *F* by using a nonlinear function $\varphi : X \rightarrow F$. In the feature space *F,* the discriminant function can be written as:

$$g(x) = \theta^T \varphi(x) + \theta_0 \qquad (1)$$

where θ is known as the weight vector and θ_0 represents the bias. Kernel methods provide the best way of tackling this problem of mapping data to the high dimensional feature space instead of computing their dot products. Suppose that the weight vector may be expressed as a linear combination of training examples (T_r) as follows:

$$\theta = \sum_{i=1}^{T_r} \beta_i x_i \quad (2)$$

Therefore, in terms of a discriminant function, it can be written as:

$$g(x) = \sum_{i=1}^{T_r} \beta_i x_i^T x_i + \theta_0 \quad (3)$$

In the feature space, Equation (1) can be written as:

$$g(x) = \sum_{i=1}^{T_r} \beta_i \varphi(x_i)^T \varphi(x_i) + \theta_0 \quad (4)$$

This representation in terms of the variables β_i is called the dual representation of the decision boundary (Ben-Hur & Weston, 2010). According to Scholkopf and Smola (2001), a kernel function is a function that returns the dot product of the vectors by taking vectors as inputs in the original space. Mathematically, for data $x, x_1 \in X$, a kernel function is defined by $k(x, x_1) = \left\langle \varphi(x)^T, \varphi(x_1) \right\rangle$, where φ is a kernel function. In terms of the kernel function, Equation (4) can be rewritten as:

$$g(x) = \sum_{i=1}^{T_r} \beta_i k(x, x_i) + \theta_0 \quad (5)$$

Kernel-based methods such as SVM have been proven as an effective technique for data analysis in different fields of life. These methods employ the kernel functions that can compute the similarity between two vectors x and x_1 (Sonnenburg et al., 2006). Since different kernels correspond to different designs of similarity. Therefore, forming a combination of different kernels may lead to a better solution to the problem.

Multiple Kernel Learning

MKL is a set of machine learning strategies that use the predefined set of kernels. The predefined set of kernels may or may not be linear but its optimality is always demanding. Instead of creating a new kernel, MKL is an efficient way to combine the existing kernels. In MKL, it is assumed that for T_r training data point (x_i, y_i), $i = 1, 2, ..., T_r$ where $x_i \in X$ for some input space *X* and $y_i \in \{-1, 1\}$, there are *M* kernel matrices that are assumed to be symmetric and positive semi-definite (PSD). The problem is to find the best linear combination of the kernel $\sum_{l=1}^{M} \gamma_l k_l$ with non-negative weights i.e. $\gamma_l \geq 0$ and $\sum_{l=1}^{M} \gamma_l = 1$ for $l = 1, 2, ..., M$ (Bach et al., 2004; Shawe-Taylor & Cristianini, 2004). In this study, three kernel functions are engaged: linear kernel function, RBF kernel and sigmoid kernel

function (Scholkopf & Smola, 2001). The linear kernel function is computed by using $k(x, x_1) = x'x_1$, RBF kernel can be defined by $k(x, x_1) = \exp(\nu \left\| x - x_1 \right\|^2)$ where ν is the positive parameter of RBF kernel for controlling its radius, and sigmoid kernel function can be stated as $k(x, x_1) = \tanh(\alpha_s x'x_1 + c_0)$ where $\alpha_s > 0$ is the scaling parameter and $c_0 \leq 0$ is the shifting parameter (Wang & Xu, 2017). The MKL practices different learning methods to combine kernels. The comprehensive details of these methods can be studied from Gonen and Alpaydin's (2011) work. The current study will apply an optimization approach for MKL to combine the kernel functions for the classification tasks of binary imbalanced datasets. The linear weighted combination of these three kernel functions is cast-off to learn them in terms of MKL.

$$MKL = \sum_{l=1}^{3} \gamma_l k_l = \gamma_1 k_1 + \gamma_2 k_2 + \gamma_3 k_3 \qquad (6)$$

where $\gamma_l \geq 0$, $\sum_{l=1}^{3} \gamma_l = 1$, and k_1, k_2, and k_3 are linear, RBF, and sigmoid kernel function respectively. γ_1, γ_2, and are the weights of the respective kernels.

Proposed Hybrid Algorithm

A hybrid algorithm is proposed during our research work for the optimization of continuous and nonlinear test functions. The bi-objective version of this hybrid algorithm was discussed by Saeed & Ong (2018). This hybrid algorithm grabbed the advantages of evolution strategies (ES) and swarm intelligence (SI). Covariance matrix adaptation evolution strategy (CMA-ES) and cuckoo search (CS) are combined for this task. As an application, this hybrid algorithm is engaged in this study.

Covariance Matrix Adaptation Evolution Strategy. CMA-ES is one of the most powerful evolutionary strategy proposed by Hansen et al. (1997). The key idea of CMA-ES lies in its invariance properties, which can be achieved by carefully planned variation, selection operators and efficient self-adaptation of mutation distribution (Igel et al., 2007). CMA-ES works with three operations: (1) Sampling from the multivariate normal distribution (2) Selection and recombination and (3) Adaptation of the covariance matrix.

Cuckoo Search. CS is one of the most popular nature-inspired metaheuristic algorithms proposed by Yang & Deb (2009), for the continuous nonlinear optimization problems. This algorithm is inspired by the cuckoos, the fascinating birds not only due to their sounds but also because of their hostile reproductive approach (Yang and Deb, 2009). Based on the egg-laying behavior of cuckoos, CS algorithm has the following three rules (1) each cuckoo lay one egg at a time and dumps it in a randomly chosen nest. (2) For the next generation,

the best nest with high-quality eggs are approved only. (3) The number of host nests (n) is fixed and the egg laid by a cuckoo is discovered by the host bird with a probability Pa. In this case, the host bird has two choices either to get rid of the egg or simply abandon the nest to build a completely new nest.

For the hybrid algorithm, after setting the objective function in CS, and after generating the initial solution, the best solution (X_{cs}^{s}) are produced at s^{th} iteration. Then with the recombination operator of CMA-ES, the weighted means m^s are produced. Before moving to the next iteration, in order to produce the new solution the best solution obtained from CS and the weighted mean from CMA-ES are plugged in into this new solution with the help of this following equation:

$$X^s = X_{CS}^s + m^s \tag{7}$$

For the next iteration (s + 1), X^s is used to get new solutions. Then the procedure of discovery and randomization are completed. All details are provided in the pseudocode of algorithm (see Algorithm 1).

Synthetic Minority Oversampling Technique

SMOTE is an oversampling algorithm for imbalanced datasets proposed by Chawla et al. (2002). This sampling technique uses oversampling of the minority class by creating synthetic samples. Subject to the amount of oversampling requirement, neighbors from the k nearest are selected. For example, if the amount of oversampling needed 200 percent then only two of the five nearest neighbors are selected and produce one sample in the direction of each. For synthetic samples following steps are applied:

1. Compute the difference between nearest neighbor and feature vector (sample).
2. Generate a random number between 0 and 1, multiply the difference by this random number.
3. Add it, to the feature vector under consideration. This will originate the selection of random point beside the line segment between two specific features.

The implementation of this algorithm requires five nearest neighbors (Han et al., 2005; Blagus & Lusa., 2013).

Proposed Scheme

The given classification scheme is proposed to study the performance of SVM with MKL (SVM+MKL) for binary imbalanced datasets. The proposed scheme is based on the above mentioned oversampling and hybrid algorithm. An oversampling algorithm (SMOTE) is applied to overcome the imbalance problem of the datasets. Each imbalanced dataset is partitioned into training, test, and validation sets using 60:20:20 ratios. As the performance of SVM is highly based on the appropriate choices of its parameters. Consequently, the

parameters of SVM including the parameters of the kernel functions along with the linear combination weights γ_l are optimized by using the proposed hybrid algorithm.

Algorithm 1
Hybrid Algorithm

Begin
1. *Setting the initial parameters, number of nests n and number of solutions N_d.*
2. *Setting the objective function f(x), $x = (x_1, x_2, ..., x_d)$, adjusting the lower and upper bounds of the test function, and constraints (if any).*
3. *Initialize CS by generating the random initial solution of n host nests.*
4. *Find the best solution (X^s_{cs}) from CS at s^{th} iteration. Initialize CMA-ES algorithm, and generate the m^s weighted means at s^{th} iteration with the help of recombination operators.*
5. *Generate the new solution X^s at s^{th} iteration using Equation (7).*
6. *Set the number of iterations and a maximum number of iterations.*
7. ***While** (number of iteration < Maximum iteration) or (stop criterion).*
8. *Produce the new solution at $(s+1)^{th}$ iteration by levy flights.*
9. *Evaluate its quality or fitness.*
10. *Choose a nest among n say (j) randomly.*
11. ***If** ($f_i > f_j$) then*
12. *Replace j with the new solution.*
    ***End** if*
13. *Abandoned the new nest using Pa and new ones are built.*
14. *Keep the best solution.*
15. *Rank the solution and find the current best.*
    ***End** while*
16. *Post process results and visualization.*

End

The objective function to be minimized is the misclassification error of the minority (positive). This optimization procedure is completed on the training dataset. Optimized parameters obtained from the training process are engaged with the test sets, where after completing the classification task, all standard evaluation measures are computed. Three established evaluation measures for imbalanced datasets namely, sensitivity, G Mean, and F measure are computed. G Mean is the geometric mean of the two prediction accuracies i.e. sensitivity: accuracy on the positive examples (minority class) and specificity: accuracy on the negative examples (majority class). It can be calculated with the help of the given below formula:

$$\text{G Mean} = \sqrt{Sensitivity \times Specificity}$$

Sensitivity and specificity can be defined as:

$$\text{Sensitivity} = \frac{Tp}{Tp + Fn}$$

$$\text{Specificity} = \frac{Tn}{Tn + Fp}$$

where Tp represents true positive examples, Fp is used for false positive, Tn for true negative and Fn is used to show false negative examples. The second evaluation measure, F measure, the harmonic mean of the precision and sensitivity can be calculated with the help of the following formula:

$$\text{F measure} = \frac{2 \times Sensitivity \times Precision}{Sensitivity + Precision}$$

and precision can be defined as:

$$\text{Precision} = \frac{Tp}{Tp + Fp}$$

The complete details of these evaluation measures can be studied from Bekkar et al. (2013). The values of G Mean and F measure varies between 0 and 1. The values near to 1 reflect the good performances by the classifiers and weak performances of the classifiers can be assessed by the low values (values near to 0). On the other hand, the high values of sensitivities reflect the good performances of the classifiers on the positive examples (minority class) only. The complete proposed scheme for the classification task is provided in a flowchart in Figure 1.

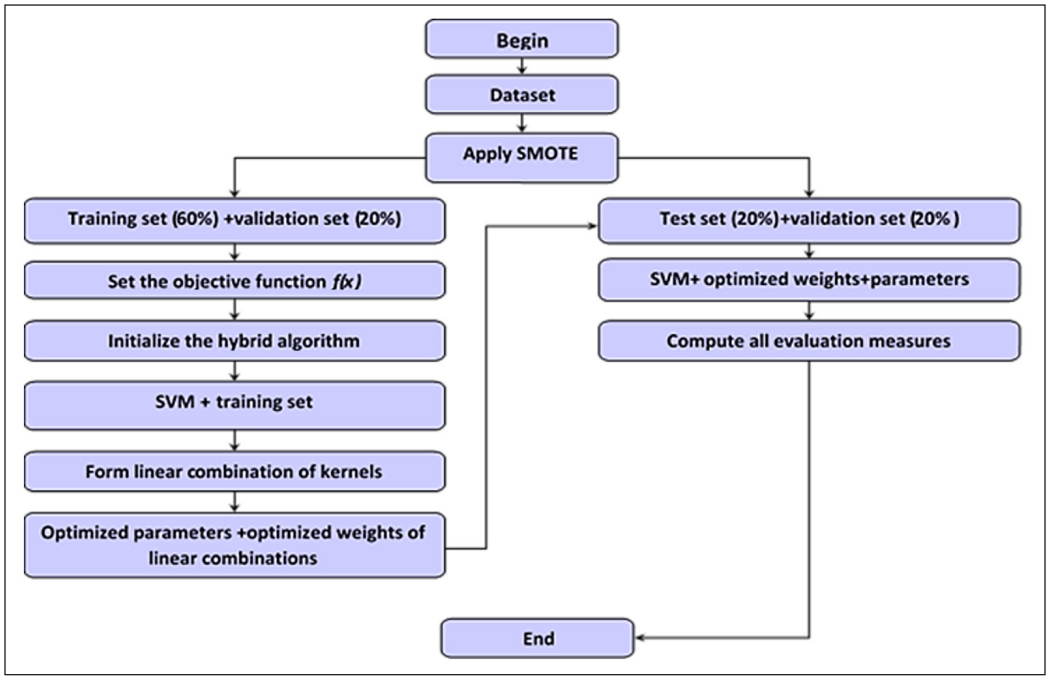

Figure 1. Flowchart of the proposed scheme for SVM+MKL

RESULTS AND DISCUSSIONS

An experimental study was conducted by engaging two types of datasets: (1) Noisy borderline imbalanced datasets and (2) Real imbalanced datasets. These datasets are taken from a well-known datasets repository KEEL (Alcala-Fdez et al., 2011). SVM is applied to each preprocessed dataset by using different kernel functions: SVM+linear, SVM+RBF, SVM+sigmoid, and SVM+MKL.

Noisy Borderline Imbalanced Datasets

Six noisy borderline imbalanced datasets namely Clove0, Clove30, Paw0, Paw30, Subclus0, and Subclus30 are engaged. Each dataset is classified by using SVM+linear, SVM+RBF, and SVM+MKL. For all noisy borderline datasets, by adding sigmoid kernel function in the linear combinations of kernels, the condition of PSD could not be satisfied. As a result, the sigmoid kernel function is excluded from the linear combinations of kernel functions. The MKL for noisy borderline imbalanced datasets is studied only by using the linear weighted combinations of the linear kernel function and RBF kernel, which can be defined as follows:

$$MKL = \gamma_1 k_1 + \gamma_2 k_2 \quad (8)$$

where k_1 represents the linear kernel function, k_2 represents the RBF kernel and γ_1, γ_2 are the weighted coefficients of these kernel function respectively. For the individual applications of the linear kernel function and RBF kernel with SVM, parameters are selected by the grid search methods. For SVM+MKL, parameters including the kernel parameter (ν) and slack variable (C) and the weights of kernel functions (γ_1 and γ_2) are optimized by using the proposed hybrid algorithm (see Table 1).

Three above mentioned evaluation measures, sensitivity represented by Sen, G Mean by G, and F measure by F are computed. The obtained results are provided in Table 2. Starting from first dataset (Clove0) to the last dataset (Subclus30), SVM+linear and

Table 1
Optimized parameters and weights of MKL for noisy borderline imbalanced datasets

Datasets	Parameters	MKL= $\gamma_1 k_1 + \gamma_2 k_2$
Clove0	ν =51.950, C =827.34	MKL= $0.20k_1 + 0.80k_2$
Clove30	ν =84.53, C =628.62	MKL= $0.50k_1 + 0.50k_2$
Paw0	ν =49.493, C =0.794	MKL= $0.183k_1 + 0.817k_2$
Paw30	ν =381.07, C =436.63	MKL= $0.2k_1 + 0.8k_2$
Subclus0	ν =101.29, C =403.58	MKL= $0.425k_1 + 0.575k_2$
Subclus30	ν =182.16, C =219.70	MKL= $0.42k_1 + 0.58k_2$

SVM+RBF performed well in terms of the good values of evaluation measures. However, SVM+RBF performed comparatively better than SVM+linear resulting in good values of all evaluation measures on all datasets. But SVM+RBF took longer testing time than SVM+linear. An outstanding performance of SVM+MKL is observed by using the proposed scheme. The proposed scheme based on the combined effect of oversampling and hybrid algorithm showed a very remarkable performance on noisy borderline datasets by using the linear combination of kernel functions with SVM. For all datasets, the maximum values of evaluation measures obtained by SVM+MKL are provided and highlighted in Table 2. A significant point in all experiments is the least testing time of SVM+MKL with the proposed scheme for all datasets.

Table 2
Performance of SVM using all kernel function on noisy borderline imbalanced datasets

Evaluation measures	Sen	G	F	Time (sec)	Sen	G	F	Time (sec)
Datasets		Clover0				Clover30		
SVM+linear	0.708	0.586	0.754	7.209	0.629	0.546	0.705	7.216
SVM+RBF	0.950	0.975	0.974	8.593	0.872	0.934	0.931	7.616
SVM+MKL	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**4.168**
		Paw0				Paw30		
SVM+linear	0.775	0.586	0.810	5.722	0.868	0.631	0.853	6.012
SVM+RBF	0.717	0.847	0.835	7.823	0.815	0.903	0.898	7.610
SVM+MKL	**1.000**	**1.000**	**1.000**	**3.105**	**1.000**	**1.000**	**1.000**	**3.420**
		Subclus0				Subclus30		
SVM+linear	0.957	0.664	0.897	6.193	0.586	0.563	0.684	5.818
SVM+RBF	0.838	0.911	0.907	8.724	0.725	0.851	0.840	7.850
SVM+MKL	**1.000**	**1.000**	**1.000**	3.046	**1.000**	**1.000**	**1.000**	**3.612**

Real Imbalanced Datasets

Another experimental study is conducted on six real imbalanced datasets to observe the role of MKL in the performance of SVM by applying the proposed scheme. A brief detail of these datasets is provided in Table 3. For MKL, the linear weighted combination is defined by using three kernel functions as follows:

$$MKL = \gamma_1 k_1 + \gamma_2 k_2 + \gamma_3 k_3 \qquad (9)$$

where k_1 represents the linear kernel function, k_2 shows the RBF kernel and k_3 represents the sigmoid kernel function, γ_1 γ_2 and γ_3 are the respective weights of these kernel functions. During the formation of the weighted linear combination of kernel functions for all datasets, the condition of PSD is carefully satisfied. For sigmoid kernel

function, only one parameter (α_s) is optimized whereas the other parameter is taken as fixed (c_0 =-1) to maintain the simplicity of the optimization process. All optimized parameters and weights of MKL are provided in Table 4.

Table 3
Datasets description

Datasets	Imbalance ratio (IR)	Total instances
Pima	1.87	768
Haberman	2.78	306
Thyroid	5.14	215
Yeast	9.08	514
Cleveland	12.62	177
Wine	29.17	1599

Table 4
Optimized parameters and weights of MKL for real imbalanced datasets

Datasets	Parameters	MKL= $\gamma_1 k_1 + \gamma_2 k_2 + \gamma_3 k_3$
Pima	ν =26.902, C =100.0503, α_s =3.2877	MKL=0.57519 k_1 +0.14168 k_2 +0.28313 k_3
Haberman	ν =69.244, C =471.96, α_s =2.5022	MKL=0.22969 k_1 +0.56385 k_2 +0.2064 k_3
Thyroid	ν =2.4396, C =6.7346, α_s =2.6732	MKL=0.10047 k_1 +0.083496 k_2 +0.81604 k_3
Yeast	ν =60.837, k_1 =867.40	MKL=0.20840 k_1 +0.79160 k_2
Cleveland	ν =50.046, C =855.44, α_s =0.95289	MKL=0.200 k_1 +0.71 k_2 +0.09 k_3
Wine	ν =59.949, C =870.621, α_s =0.90236	MKL=0.11414 k_1 +0.76884 k_2 +0.0.11702 k_3

For yeast dataset, by adding the sigmoid kernel function, the condition could not be satisfied. Therefore, for this dataset, a weighted linear combination is formed by using only two kernel functions, linear kernel function, and RBF kernel. To study the performance of SVM with MKL by applying the proposed scheme, all real imbalanced datasets are classified by using SVM+linear, SVM+RBF, SVM+sigmoid, and SVM+MKL. For SVM+MKL, all parameters and their respective weights are optimized by applying the proposed hybrid algorithm.

The obtained results of all evaluation measures namely Sen, G, and F from all datasets are provided in Table 5. For the first dataset (Pima), with imbalance ratio (IR=1.87), RBF and sigmoid kernel performed very well resulting in maximum values of evaluation measures. SVM+linear also performed well. Nevertheless, its performance is not better than SVM+RBF and SVM+sigmoid. SVM+MKL showed maximum sensitivity for this dataset with minimum testing time.

For the second dataset, Haberman, again RBF and sigmoid kernels performed well but their performances on the minority class are less admirable than SVM+MKL (Sen=1.000). For the Thyroid dataset with IR=5.14, an outstanding performance is observed by SVM+linear and SVM+MKL. This time although SVM+RBF and SVM+sigmoid performed well their performances are less as compared to SVM+linear and SVM+MKL. On the other hand, the minimum testing time for SVM+MKL is observed for this dataset. For the fourth dataset (Yeast) with IR= 9.08, the average performances are observed by SVM+linear, SVM+RBF, and SVM+sigmoid. SVM+MKL remained successful in producing the maximum value of G (0.764) in minimum testing time. Approximately the same performances of RBF and sigmoid kernel functions are observed on Cleveland dataset. For Cleveland (IR=12.62) and Wine (IR=29.17), an outstanding performance of SVM+MKL can be seen resulting in maximum values of all evaluation measures. Though, for these two datasets, the testing time taken by SVM+MKL is longer than the other methods. Out of six datasets, the outstanding performance of SVM+MKL in terms of the maximum values of all evaluation measures (Sen, G, and F) is observed for three datasets (Thyroid, Cleveland, and Wine). All maximum values of evaluation measures obtained from SVM+MKL are highlighted in Table 5. The proposed scheme for SVM+MKL based on the oversampling and hybrid algorithm successfully handled imbalanced datasets.

Table 5
Performance of SVM using all kernel functions on real imbalanced datasets

Evaluation measures	Sen	G	F	Time (sec)	Sen	G	F	Time (sec)
Datasets		Pima				Haberman		
SVM+linear	0.844	0.880	0.913	6.797	0.574	0.732	0.728	6.029
SVM+RBF	1.000	1.000	1.000	7.230	0.984	0.992	0.992	5.601
SVM+sigmoid	1.000	1.000	1.000	6.854	0.991	0.995	0.995	5.095
SVM+MKL	1.000	0.727	0.678	**1.745**	**1.000**	0.845	0.879	**1.861**
		Thyroid				Yeast		
SVM+linear	1.000	1.000	1.000	1.947	0.676	0.737	0.780	3.375
SVM+RBF	0.813	0.901	0.897	1.825	0.407	0.638	0.579	3.492
SVM+sigmoid	0.872	0.934	0.932	1.899	0.395	0.628	0.566	3.342
SVM+MKL	1.000	1.000	1.000	**1.508**	0.653	**0.764**	0.731	**2.126**
		Cleveland				Wine		
SVM+linear	1.000	0.990	0.996	1.162	0.758	0.711	0.745	0.834
SVM+RBF	0.631	0.794	0.774	1.808	0.129	0.359	0.229	0.938
SVM+sigmoid	0.631	0.794	0.774	1.663	0.646	0.804	0.785	0.828
SVM+MKL	1.000	**1.000**	**1.000**	1.591	**1.000**	**1.000**	**1.000**	1.553

Statistical Ranks to SVM's with all Kernel Functions

Statistical ranks to SVM's with all kernel functions are assigned by using two well-known statistical non-parametric rank test: Friedman test and Quade test. These rank tests are applied to both types of datasets: noisy borderline datasets and real imbalanced datasets. The details of these tests can be found in Demsar (2006), Garcia et al. (2007), Trawinski et al. (2012) and Pohlert (2014). The null hypothesis to be tested, in both tests, is that on the average the performances of SVM's with all kernel functions are equal. Friedman's test follows an approximately Chi-square distribution for a large number of blocks (datasets) and treatments (SVM with all kernel functions). As we have a small number of treatments. Therefore, the critical values are derived from their tables. For noisy borderline imbalanced datasets, as the number of blocks is 6 and number of treatment is 3. Hence, 7.00 and 4.10 are the critical values of Friedman and Quade test respectively. All obtained results are provided in Table 6. The minimum ranks in Table 6 (bold ranks) are justifying the leading position of SVM+MKL. Graphically, these ranks are presented in Figure 2. For six real imbalanced datasets, as the number of treatments is 4 and number of blocks is also 6. Thus, the critical values for Friedman and Quade tests are 7.6 and 3.29 respectively. For these datasets, both tests showed insignificant results for most of the cases (see Table 7). On the other hand, the minimum ranks obtained by these tests justify the leading position of SVM+MKL. The obtained ranks for SVM+MKL are bold and shown in Table 7. The graphical representation of all ranks for all cases is shown in Figure 3.

Table 6
The average ranking of SVM performances by using all kernel functions on noisy borderline imbalanced datasets

	With respect to sensitivity		With respect to G Mean	
Methods	Friedman test	Quade test	Friedman test	Quade test
SVM+linear	2.5	2.7143	3	3
SVM+RBF	2.5	2.2857	2	2
SVM+MKL	**1**	**1**	**1**	**1**
Test statistics	9	9	12	21
Critical values	7	4.10	7	4.10
Decision (5%)	significant	significant	significant	significant
	With respect to F measure		With respect to testing time	
Methods	Friedman test	Quade test	Friedman test	Quade test
SVM+linear	3	3	2	2
SVM+RBF	2	2	3	3
SVM+MKL	**1**	**1**	**1**	**1**
Test statistics	12	21	12	21
Critical values	7	4.10	7	4.10
Decision (5%)	significant	significant	significant	significant

CONCLUSION

A study was conducted to observe the performance of SVM with MKL for binary imbalanced datasets including noisy borderline and real imbalanced datasets. For this given task, a new scheme based SMOTE and hybrid algorithm had proposed. An experimental study was conducted to justify the validity of the proposed scheme, by engaging the noisy borderline and real imbalanced datasets. For MKL, the weighted linear combinations of kernel functions were formed after satisfying the condition of PSD. By applying SMOTE and optimizing the parameters of SVM along with the weights of the kernel functions on the training datasets, these optimized parameters were engaged with testing datasets to fulfill the classification tasks. SVM is applied by using SVM+linear, SVM+RBF, SVM+sigmoid, and SVM+MKL to all datasets. Three evaluation measures (Sen, G, and F) were observed. An outstanding performance of the proposed scheme for SVM+MKL was observed for noisy borderline datasets. Though, an average performance was observed for real imbalanced datasets. In all, it can be said that our proposed scheme for SVM+MKL showed an enhanced performance for the classification of binary imbalanced datasets.

Table 7
The average ranking of SVM performances by using all kernel functions on real imbalanced datasets

	With respect to sensitivity		With respect to G Mean	
Methods	Friedman test	Quade test	Friedman test	Quade test
SVM+linear	2.1667	2.2381	2.5833	2.8810
SVM+RBF	3.25	3.4286	3	2.8571
SVM+sigmoid	2.9167	2.9524	2.5	2.1429
SVM+MKL	**1**.5	**1.333**	**1.9167**	**2.1190**
Test statistics	3.65	3.38	2.15	0.48
Critical values	7.6	3.29	7.6	3.29
Decision (5%)	**i**nsignificant	s**i**gnificant	**i**nsignificant	insignificant
	With respect to F measure		With respect to testing time	
Methods	Friedman test	Quade test	Friedman test	Quade test
SVM+linear	2.4167	2.7857	2.667	2.6667
SVM+RBF	3.00	2.8571	3.333	3.5238
SVM+sigmoid	2.50	2.1429	2.333	2.2857
SVM+MKL	**2.0833**	**2.2143**	**1.6667**	**1.5238**
Test statistics	1.55	0.36	5.20	2.52
Critical values	7.6	3.29	7.6	3.29
Decision (5%)	insignificant	insignificant	insignificant	insignificant

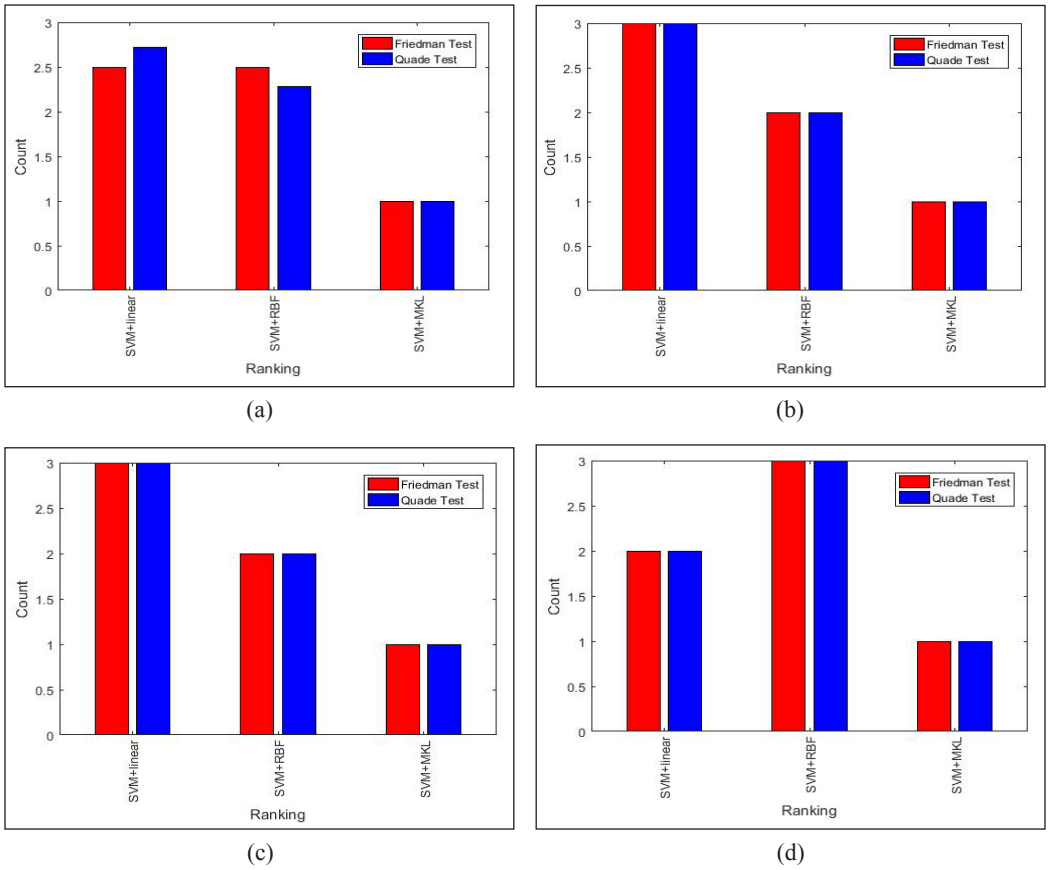

Figure 2. Average ranks of SVM's with all kernel functions on noisy borderline imbalanced datasets: (a) with respect to sensitivity; (b) with respect to G Mean; (c) with respect to F measure; and (d) with respect to testing time

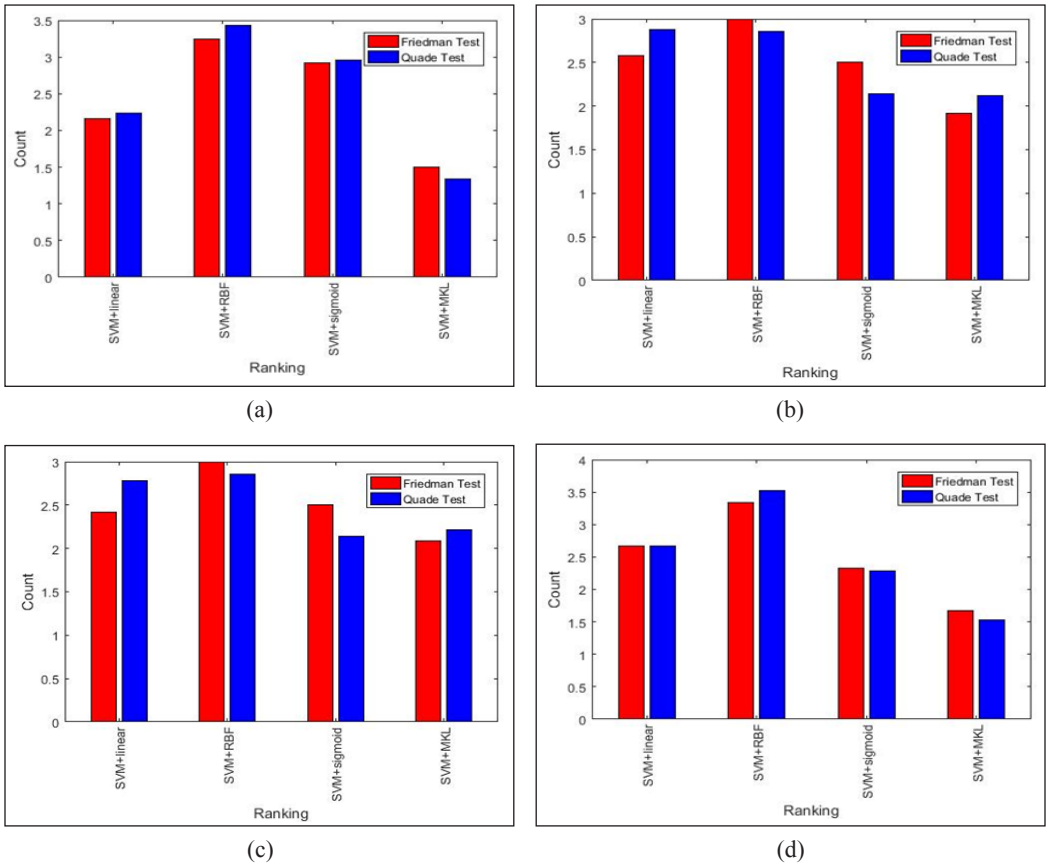

Figure 3. Average ranks of SVM's with all kernel functions on real imbalanced datasets: (a) with respect to sensitivity; (b) with respect to G Mean; (c) with respect to F measure; and (d) with respect to testing time (sec)

REFERENCES

Abe, S. (2005). *Support vector machines for pattern classification* (2nd Ed.). London: Springer.

Alcala-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., Garcia, S., Sanchez, L., & Herrera, F. (2011). Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing, 17*, 255-287.

Alwan, H. B., & Ku-Mahamud, K. R. (2013). Solving svm model selection problem using acor and iacor. *WSEAS Transactions on Computers, 12*(9), 277-288.

Bach, F. R., Lanckriet, G. R., & Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the Twenty-First International Conference on Machine Learning* (p. 6). Banff, Alberta, Canada.

Bao, Y., Hu, Z., & Xiong, T. (2013). A pso and pattern search based memetic algorithm for svms parameters optimization. *Neurocomputing, 117*, 98-106.

Bekkar, M., Djemaa, H. K., & Alitouche, T. A. (2013). Evaluation measure for models assessment over imbalanced data sets. *Journal of Information Engineering and Applications, 3*(10), 27-38. Retrieved from https://eva.fing.edu.uy/mod/resource/view.php? id=33977

Ben-Hur, A., & Weston, J. (2010). A user' guide to support vector machines. *Data Mining Techniques for the Life Sciences, 609*, 223-239. doi: 10.1007/978-1-60327-24/-4_13

Blagus, R., & Lusa, L. (2013). Smote for high-dimensional class-imbalanced data. *BMC Bioinformatics, 14*, 106-121.

Blondin, J., & Saad, A. (2010). Metaheuristic techniques for support vector machine model selection. In *Proceedings of the 10th International Conference on Hybrid Intelligent Systems* (pp. 197-200). Atlanta, GA, USA.

Cao, P., Zhao, D., & Zaiane, O. (2013). An optimized cost-sensitive svm for imbalanced data learning. In *Proceedings of the 17th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, Pakdd* (pp. 280-292). Berlin, Heidelberg.

Cervantes, J., Garcia, L. F., Rodriguez, L., Lopez, A., Castilla, J. R., & Trueba, A. (2017). Pso-based method for svm classification on skewed data set. *Neurocomputing, 228*, 187-197.

Chaudhuri, A., & De, K. (2011). Fuzzy support vector machine for bankruptcy prediction. *Applied Soft Computing, 11*, 2472-2486.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321-357.

Cortes, C., & Vapnik, V. (1995). Support-vector network. *Machine Learning, 20*(3), 273-297.

Dehak, R., Dehak, N., Kenny, P., & Dumouchel, P. (2008). Kernel combination for svm speaker verification. In *Odyssey: The Speaker and Language Recognition Workshop Stellenbosch* (p. 21). South Africa.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1-30.

Eitrich, T., & Lang, B. (2006). Efficient optimization of support vector machine learning parameters for unbalanced data sets. *Journal of Computational and Applied Mathematics, 196*(2), 425-436.

Garcia, S. A., Bentez, A. D., Herrera, F., & Fernandez, A. (2007). Statistical comparisons by means of non-parametric tests: a case study on genetic based machine learning. In *Proceedings of EI Congreso Espanol de Informática (CEDI 2007)*. Zaragoza, Spain.

Gonen, M., & Alpaydin, E. (2011). Multiple kernel learning algorithms. *Journal of Machine Learning Research, 12*, 2211-2268.

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline-smote: a new oversampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing* (pp. 878-887). Berlin, Heidelberg.

Hansen, N., & Ostermeier, A. (1997). Convergence properties of evolution strategies with derandomized covariance matrix adaptation: The ($u \setminus u_1, \lambda$) cma-es. In *Proceedings of the 5th European Congress on Intelligent Techniques and Soft Computing* (pp. 650-654). Berlin, Germany. doi: 10.1.130.648

Hsu, C. W., & Chang, C. C., & Lin, C. J. (2003). *A practical guide to support vector classification*. Department of Computer Science, National Taiwan University. Retrieved February 19, 2018, from www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

Hsu, C., & Lin, C. J. (2002). A simple decomposition method for support vector machines. *Machine Learning, 46*(1), 291-314.

Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications, 33*, 847-856.

Igel, C., Hansen, N., & Roth, S. (2007). Covariance matric adaptation for multi-objective optimization. *Evolution Computation, 15*(1), 1-28.

Imam, T., Ting, K. M., & Kamruzzaman, J. (2006). Z-svm: an svm for improved classification of imbalanced data. In *Proceedings of the Australian Conference on Artificial Intelligence* (pp. 264-273). Berlin, Heidelberg.

Jiang, P., Missoum, S., & Chen, Z. (2014). Optimal svm parameter selection for non-separable and unbalanced datasets. *Structural and Multidisciplinary Optimization, 50*(4), 523-535.

Lanckriet, G. R., Cristianini, N., Bartlett, P., Ghaoui, L. E., & Jordan, M. I. (2004). Leaning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research, 5*, 27-72.

Pohlert, T. (2014). *The pairwise multiple comparison of means ranks package (pmcmr)*. Retrieved February 19, 2018, from http://CRANR-project.org /package=PMCMR.

Ren, Y., & Bai, G. (2010). Determination of optimal svm parameters by using ga/pso. *Journal of Computers, 5*, 1160-1168.

Saeed, S., & Ong, H. C. (2018). A bi-objective hybrid algorithm for the classification of imbalanced noisy and borderline data sets. *Pattern Analysis and Applications, 2018,* 1-20. doi: org/10.1007/s10044-018-0693-4

Sain, H., & Purnami, S. W. (2015). Combine sampling support vector machine for imbalanced data classification. *Procedia Computer Science, 72*, 59-66.

Scholkopf, B., & Smola, A. J. (2001). *Learning with kernels: support vector machines, regularization, optimization and beyond*. MA: MIT press.

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel methods for pattern analysis*. Cambridge: Cambridge University Press.

Shin, K. S., Lee, T. S., & Kim, H. J. (2005). An application of support vector machines in bankruptcy prediction model. *Expert Systems with Applications, 28*, 127-135.

Sonnenburg, S., Ratsch, G., Schafer, C., & Scholkopf, B. (2006). Large Scale multiple kernel learning. *Journal of Machine Learning Research, 7*, 1531-1565.

Sun, A., Lim, E. P., & Liu, Y. (2009). On strategies for imbalanced text classification using svm: a comparative study. *Decision Support Systems, 48*(1), 191-201.

Trawinski, B., Smetek, M., Telec, Z., & Lasota, T. (2012). Nonparametric statistical analysis for multiple comparisons of machine learning regression algorithms. *International Journal of Applied Mathematics and Computer Science, 22*(4), 867-881.

Wang, H., & Xu, D. (2017). Parameter selection method for support vector regression based on adaptive fusion of the mixed kernel function. *Journal of Control Science and Engineering, 2017*, 1-12. doi:10.1155/2017/3614790

Wang, L., Xu, G., Wang, J., Yang, S., Guo, L., & Yan, W. (2011). Ga-svm based feature selection and parameters optimization for bci research. In *Proceedings of the Seventh International Conference on Natural Computation* (pp. 580-583). Shanghai, China. doi: 10.1109/ICNC.2011.6022083

Wang, Q. (2014). A hybrid sampling svm approach to imbalanced data classification. In *Abstract and Applied Analysis* (Vol. 2014, pp. 1-7). Cairo: Hindawi Publishing Corporation. doi: 10.1155/2014/972786

Wang, Q., Luo, Z., Huang, J., Feng, Y., & Liu, Z. (2017). A novel ensemble method for imbalanced data learning: bagging of extrapolation-smote svm. *Computational Intelligence and Neuroscience, 2017,* 1-11. doi: org/10.1155/2017/1827016

Wu, S. J., Pham, V. H., & Nguyen, T. N. (2017). Two phase optimization for support vectors and parameter selection of support vector machines: two-class classification. *Applied Soft Computing, 59*, 129-142.

Yang, X. S. (2010). *Nature inspired metaheuristic algorithms* (2nd Ed.). Frome: Luniver press.

Yang, X. S., & Deb, S. (2009). Cuckoo search via levy flights. In *Proceedings of the World Congress on Nature and Biologically Inspired Computing* (pp. 210-214). Coimbatore, India. doi: 10.1109/NABIC.2009.5393690

Zhang, B. (2006). *Svm kernel optimization: an example in yeast protein subcellular localization prediction*. Retrieved February 19, 2018, from http://www.cs.cmu.edu/~epxing/class/10701-06f/project-reports/buck-zhang.pdf.