Regularized Least Squares: A Useful (Forgotten) Tool for Supervised and Semi-Supervised Learning

S. Andrew Gadsden, Dhafar Mohammed, and Mohammad Al-Shabi
Dept. of Mechanical Engineering, McMaster University
Hamilton, Ontario, Canada, L8S 4L7

ABSTRACT

This paper discusses the undervalued importance of Regularized Least Squares, and its continued usefulness in solving supervised and semi-supervised learning problems. The common two moon classification problem was used to study and compare the effectiveness of three methods: the Support Vector Machine (Radial Basis Functions and 6th Order Polynomials), Laplacian Regularized Least Squares, and K-Means with Regularized Least Squares (KM-RLS). The latter approach, which appears to be novel, is shown to be a very strong candidate for designing the hidden layer part of the classifier. As shown in the detailed results section, the KM-RLS method yields excellent results, and is computationally faster and less complex, when compared with the commonly used Support Vector Machine.

Keywords: Regularized Least Squares, Support Vector Machine, Laplacian, K-Means, and classification problems.

1. INTRODUCTION

In neural networks, the process of classifying data involves labeling each data point to a certain group or class. The machine learning process is commonly utilized to perform the classification. Machine learning involves computer algorithms and techniques that enable computers to appear to 'learn' or perform a set of iterations with increasing accuracy. The three most common types of machine learning are: unsupervised, semi-supervised, and supervised.

In the case of unsupervised learning, labeled examples are not available; hence the learning process is termed unsupervised. It is important to note that the use of unlabeled examples is not computationally expensive. However, it may lead to less accurate results when compared to semi-supervised and supervised learning. Semi-supervised learning combines both labeled and unlabeled data. Supervised learning functions—generally created through training—map inputs directly to the desired outputs. Essentially, the supervised learner is "to predict the value of the function for any valid input object after having seen a number of training examples" [1].

The most common classification methods include: the Support Vector Machine (SVM), Self Organizing Maps (SOM), Linear Vector Quantization (LVQ), and feedforward neural networks such as the basic perceptron. It is proposed that clustering techniques such as K-Means (KM) may be used in conjunction with classification methods such as Regularized Least Squares (RLS) to enhance the quality and accuracy of the classification.

2. WHY REGURALIZED LEAST SQUARES?

Inverse problems are typically systems where the model parameters are obtained from a set of observed data. In terms defined by Hadamard, inverse problems are usually ill-posed [2]. Hadamard stated that well-posed problems are mathematical models of physical systems where a solution exists, is unique, and depends continuously on the data [3]. To solve inverse problems numerically, "one must introduce some additional information about the solution, such as an assumption on the smoothness or a bound on the norm" [4]. This process is typically referred to as regularization.

A simple form (and the most popular) is Tikhonov regularization. Tikhonov introduced the idea in [2] when he considered a system of linear algebraic equations. Assuming that the system has no exact solution, a common approach to finding an approximate solution is to use the method of linear least squares [2]. As stated by Rifkin and Lippert, if one combines Tikhonov regularization in a Reproducing Kernel Hilbert Space (RKHS) using the square loss method, one obtains the Regularized Least Squares (RLS) classification algorithm [5].

Generally speaking, a RKHS is "a function space in which pointwise evaluation is a continuous linear functional" [6]. As defined in [5], the Tikhonov minimization problem with a square loss may be written as follows:

$$\min_{f \in H} \frac{1}{2} \sum_{i=1}^{n} (f(X_i) - Y_i)^2 + \frac{\lambda}{2} \|f\|_K^2 \tag{1}$$

$$f = \sum_{i=1}^{n} c_i k(X_i, \cdot) \tag{2}$$

Where X and Y are the data points, λ is the regularization parameter, K is the kernel matrix (which may be linear, polynomial, or Gaussian), f is the solution to Eq. (1), and c exists and is unique. Rifkin and Lippert [5] used some basic properties of the RKHS to rewrite Eq. (1) into the following:

$$\min_{c \in \Re^n} \frac{1}{2} \|Y - Kc\|_2^2 + \frac{\lambda}{2} c^t Kc \tag{3}$$

$$(K + \lambda I)c = Y \tag{4}$$

Differentiating Eq. (3) with respect to c, one obtains Eq. (4). Note that I is an identity matrix. It is important to note that a "regularized least squares classification problem may now be solved by a single system of linear equations" [5].

The RLS algorithm allows the training process to be much simpler, as the required solution is in the form of linear equations (see Eq. (4)). Unlike the SVM, which requires solving a quadratic equation, the RLS has the potential for yielding accurate results at a quicker pace since it is computationally less expensive. The computational time of the SVM is dependent on the amount of computer power and memory available, the number of data points, and the number of Support Vectors generated. The steps to solving a general RLS problem include computing the K matrix, and solving the corresponding linear system. As such, the RLS algorithm is simpler to understand when compared with the SVM. Combining the RLS method with the K-Means algorithm yields interesting results. Although the Regularized Least Squares classification method is not a new method, it becomes apparent that it is a useful tool for supervised and semi-supervised learning.

3. THE K-MEANS ALGORITHM: CANDIDATE FOR THE HIDDEN LAYER

The K-Means (KM) algorithm is used to cluster objects or points into partitions, based on specific attributes. Essentially, the main goal of KM is to minimize the total intra-cluster variance [7]. The cluster variance may be calculated using the following squared error function [7]:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} \left| x_j - \mu_i \right|^2 \tag{5}$$

Where k refers to the number of clusters (S), x refers to the data points, and μ is the corresponding centroid.

The first step is to partition the inputs into initial sets, and then calculate the mean (or the centroid). Next, the algorithm creates new partitions by relating each point with the closest mean [7]. The centroids are then recalculated, and the algorithm repeats itself until convergence. In general, these steps are completed rather quickly. Hence, this method remains rather popular due to its relatively fast computational time. However, note that the final solution depends on the initial set of clusters. The quality of the global optimum can therefore vary between iterations. A common way to avoid this includes running the algorithm a few times and returning the best cluster found [7]. Including the KM clustering method with a classification tool such as RLS proves to be quite useful.

4. COMPARATIVE EVALUATION: TWO MOONS PROBLEM

This section outlines the main results of implementing the SVM, the Laplacian RLS (LapRLS), and KM-RLS techniques on the two moons problem. Note that the following results were obtained using: Windows XP (Professional Edition) Operating System (SP 2), Intel Dual Core 2.5 GHz, 1.0 GB of RAM, and 3.0 GB of Virtual Memory. MATLAB 7.4.0 (R2007a) was used to run the code and generate the figures.

This problem involves a two moons (or half moons) pattern, as shown in Figure 1. The points belonging to the top and bottom moon are referred to as belonging to Class A and B, respectively. The moons are extremely close to each other, where the centers have been shifted 0.985 (normalized) vertically. Four different datasets were used, each with varying noise amplitudes (0, 0.05, 0.15, and 0.25) for 3,258 points. Simply put, a variety of classification techniques were used to separate and identify the points belonging to both moons. The techniques were then compared and evaluated based on the quality of the classification, and the computational difficulty and running time.

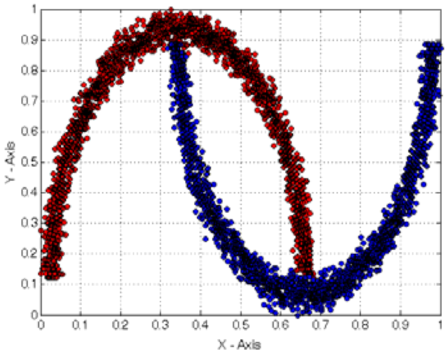

Figure 1: Two Moons Problem

LapRLS Results

The following table lists the quantitative results obtained from running the LapRLS simulation.

Table 1: Laplacian RLS Results

Data-set	**Results**				
	Class A Points	Class B Points	Misclassified Points	Classification Percentage	Run Time
1	1,638	1,620	53 (27, 36)	98.37	154 sec
2	1,602	1,656	45 (36, 9)	98.62	155 sec
3	1,635	1,623	52 (23, 29)	98.40	155 sec
4	1,621	1,637	68 (38, 30)	97.91	155 sec

SVM Results

The following table lists the quantitative results obtained from running the SVM simulation.

Table 2: Support Vector Machine Results

Data set	**Method**	**Results**				
		Class A Points	Class B Points	Misclassified Points	Classification Percentage	Run Time
1	RBF	818	810	36	97.79	51 min
	Poly 6	814	814	0	100	75 min
2	RBF	823	805	35	97.85	52 min
	Poly 6	814	814	0	100	73 min
3	RBF	815	813	35	97.85	54 min
	Poly 6	820	808	6	99.63	73 min
4	RBF	807	821	39	97.60	52 min
	Poly 6	811	817	13	99.20	76 min

KM-RLS Results

The following table lists the quantitative results obtained from running the KM-RLS simulation.

Table 3: K-Means RLS Results

Data set	**Results**					
	Clusters Total (Each)	Class A Points	Class B Points	Misclassified Points	Classification Percentage	Run Time
1	42 (21)	1,629	1,629	0 (0, 0)	100	≈ 8 sec
2	34 (17)	1,627	1,631	2 (2, 0)	99.94	≈ 5 sec
3	36 (18)	1,620	1,638	13 (11, 2)	99.60	≈ 6 sec
4	78 (39)	1,614	1,644	17 (16, 1)	99.48	≈ 28 sec

The following figure illustrates the results of the KM-RLS simulation. Note how the clusters are well formed within the data points. The classification regions (different shades) are well defined and classify the data points within a reasonable amount of error. Figure 3 (next page) zooms in on the upper region of Figure 2 for more detail.

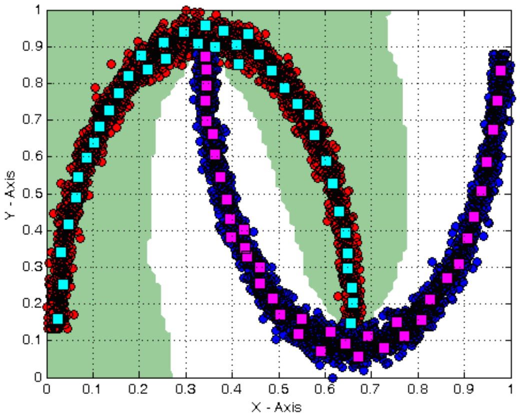

Figure 2: KM-RLS Results

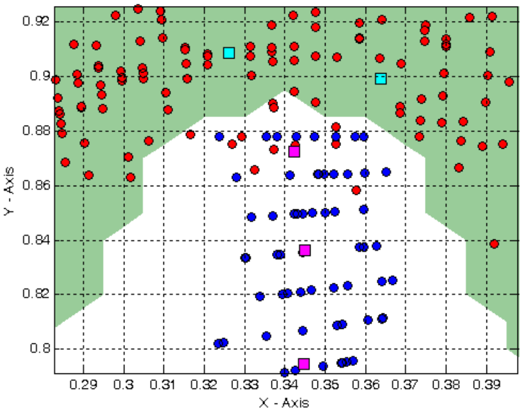

Figure 3: KM-RLS Results (Upper Zoomed)

5. DISCUSSION

Figures 4 through 7 were created to aid the comparison process of the classification techniques. Figure 4 demonstrates that the KM-RLS algorithm performed extremely well. The KM-RLS method obtained a higher average classification percentage over the four datasets, and also performed the calculations considerably faster (by about an order of 3). Figure 5 represents the ratio of average classification percentage and run time. In other words, it demonstrates the amount of classification per time. A higher ratio refers to a quicker classification process. Note how the SVM performed slowly. However, as shown in Figure 6, the SVM (Poly 6) yielded a higher quality classification, when compared with LapRLS (even though LapRLS was faster).

It is interesting to note that the Laplacian RLS performed differently depending on the number of labeled data used. For example, the best results were obtained when 25 labeled data were used. The algorithm therefore required a significant amount of parameter tuning in order to find the appropriate balance. Based on the above results and opinions of the authors, the ranking of the classification techniques (for the tested cases) is as follows: KM-RLS, SVM (Poly 6), LapRLS, and finally SVM (RBF). This ranking was based on computational complexity, run time, and classification results. Overall, the K-Means algorithm combined with Regularized Least Squares yielded more accurate solutions, as demonstrated in the results.

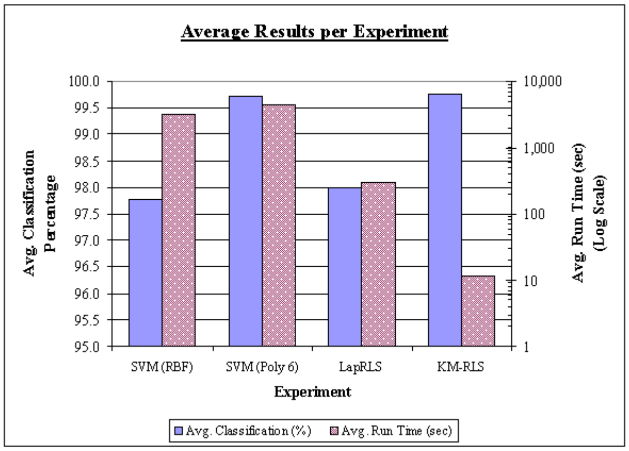

Figure 4: Average Results per Experiment

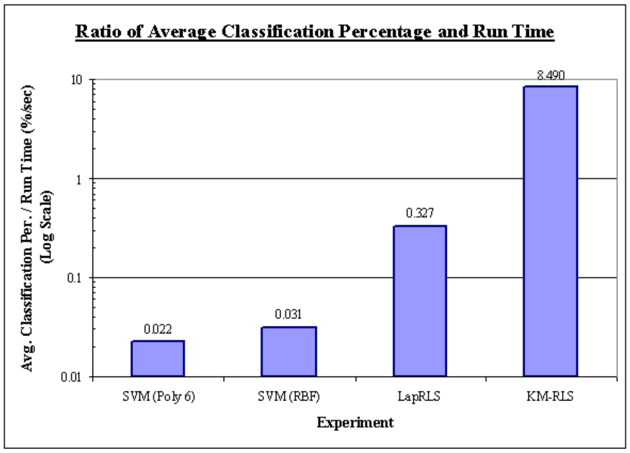

Figure 5: Ratio of Average Classification Percentage and Run Time

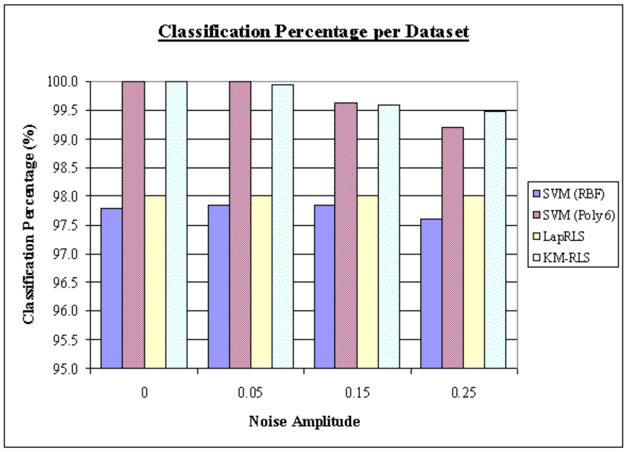

Figure 6: Classification Percentage per Dataset

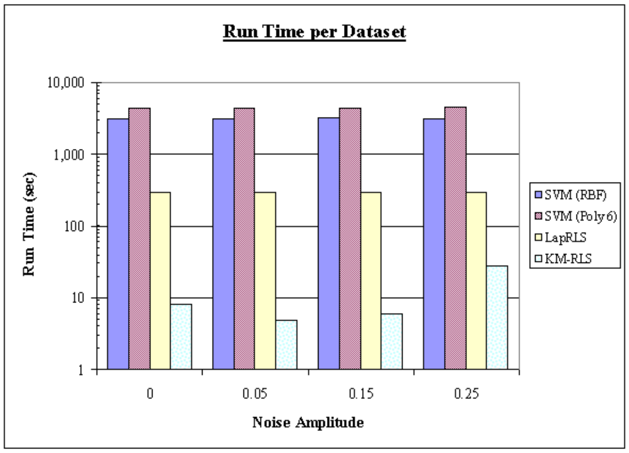

Figure 7: Run Time per Dataset

6. CONCLUSIONS

KM-RLS yielded excellent results. It was relatively easy to code and the algorithm compiled very fast. From a user point-of-view, the process of running the code was also very easy as the parameter tuning process (to obtain optimal results) was easy (especially when compared to LapRLS). For these reasons, using the two moon problem as an example, the KM-RLS classification technique was found to be superior when compared to the SVM and LapRLS.

7. ACKNOWLEDGEMENTS

The authors would like to thank the kind contributions made by Dr. Simon Haykin (McMaster University). His support and effort throughout the making of this paper proved to be essential to its successful completion. The authors would also like to express gratitude towards Mohammed El Sayed (McMaster University) for his contributions to the algorithms, computer code, and stimulating discussions.

8. REFERENCES

[1] S. Kotsiantis, *Supervised Machine Learning: A Review of Classification Techniques*, Informatica Journal, 2007, Volume 31, Pages 249-268.

[2] A. N. Tikhonov and V. Y. Arsenin, Solutions of Ill-Posed Problems, V. H. Winston and Sons, 1977.

[3] J. Hadamard, *Lectures on Cauchy's Problem in Linear Partial Differential Equations*, Yale University, 1923.

[4] R. Tibshirani, *Regression Shrinkage and Selection via the Lasso*, Journal of the Royal Statistical Society, 1996, Series B (Methodology), Volume 58, Number 1, Pages 267–288.

[5] R. M. Rifkin and R. A. Lippert, *Notes on Regularized Least Squares*, Computer Science and Artificial Intelligence Laboratory Technical Report, MIT, 2007.

[6] Nachman Aronszajn, *Theory of Reproducing Kernels*, Transactions of the American Mathematical Society, 1950, Volume 68, Number 3, Pages 337-404.

[7] J. A. Hartigan and M. A. Wong, *A K-Means Clustering Algorithm*, 1979, Applied Statistics, Volume 28, Number 1, Pages 100–108.