

# Incremental and Decremental Least Squares Support Vector Machine and Its Application to Drug Design

Hyunsoo Kim  
Dept. of Computer Science  
University of Minnesota, Twin Cities  
Minneapolis, MN 55455, USA  
hskim@cs.umn.edu

Haesun Park  
Dept. of Computer Science  
University of Minnesota, Twin Cities  
Minneapolis, MN 55455, USA  
hpark@cs.umn.edu

## Abstract

The least squares support vector machine (LS-SVM) has shown to exhibit excellent classification performance in many applications. In this paper, we propose an incremental and decremental LS-SVM based on updating and downdating the QR decomposition. It can efficiently compute the updated solution when data points are appended or removed. The experiment results illustrated that the proposed incremental algorithm efficiently produces the same solutions as those obtained by LS-SVM which recomputes the solution all over even for small changes in the data. For drug design, the results of each biochemistry laboratory test on a new compound can be iteratively included in the training set. This procedure can further improve precision in order to select the next best predicted organic compound. Instead of retraining entire data points, it is much efficient to update solution by incremental LS-SVM.

## 1. Introduction

There have been several approaches to build incremental learning machines, which can effectively compute an updated decision function when data points are appended. For many applications where expensive updating of transactions is frequently required, it is desirable to develop incremental and decremental machine learning algorithms, which can effectively compute updated decision function when data points are deleted or appended. Recently, incremental and decremental support vector machine has been introduced based on decay coefficients [3]. The least squares support vector machine (LS-SVM) [2] has shown to exhibit excellent classification performance in many applications. In this paper, an incremental and decremental least squares support vector machine (LS-SVM) is designed by updating and downdating the QR decomposition.

## 2. Incremental and Decremental LS-SVM

For the training data  $(\mathbf{a}_i, y_i)$  with  $y_i \in \{-1, +1\}$  for  $1 \leq i \leq m$ , the discriminant function in the feature space is  $f(\mathbf{x}) = \phi(\mathbf{x})^T \mathbf{w} + b$ , where  $\phi(\cdot)$  is a mapping function, i.e.  $\phi(\cdot) : \mathcal{S} \subset \mathbb{R}^n \rightarrow \mathcal{F} \subset \mathbb{R}^n$ , that maps the input data to a vector in the feature space. For this binary classification problem, the following linear system can be built as

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 \\ \vdots \\ \vdots & K + C^{-1}I \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} b \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} 0 \\ -\mathbf{u}_{m_1} \\ \mathbf{u}_{m_2} \end{bmatrix}, \quad (1)$$

where  $K \in \mathbb{R}^{m \times m}$  is a kernel matrix,  $C$  is a regularization parameter,  $\mathbf{u}_i \in \mathbb{R}^{i \times 1}$  is a column vector with 1's as its elements, and  $m_j$  is the number of data points that belong to class  $j$ . It can be easily shown that this formulation is equivalent to LS-SVM. The solution of the linear system can be found by computing the QR decomposition of the matrix  $\hat{K} \in \mathbb{R}^{(m+1) \times (m+1)}$ :

$$\hat{K} = \begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 \\ \vdots \\ \vdots & K + C^{-1}I \\ \vdots \\ 1 \end{bmatrix} = QR, \quad (2)$$

where  $Q \in \mathbb{R}^{(m+1) \times (m+1)}$  is an orthogonal matrix and  $R \in \mathbb{R}^{(m+1) \times (m+1)}$  is an upper triangular matrix. Then, the solution of the linear system can be directly computed

by

$$R\mathbf{x} = Q^T \begin{bmatrix} 0 \\ -\mathbf{u}_{m_1} \\ \mathbf{u}_{m_2} \end{bmatrix} = Q^T \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix}. \quad (3)$$

Now, LS-SVM of Eqn. (1) can be achieved by the QR decomposition. When data points are appended or removed, the updated solution  $\mathbf{x}^*$  can be efficiently obtained by updating and downdating the QR decomposition since the computational complexity of updating and downdating the QR decomposition is  $O(m^2)$  [1]. The updated solution  $\mathbf{x}^*$  can be computed by

$$R^*\mathbf{x}^* = (Q^*)^T \begin{bmatrix} 0 \\ \mathbf{y}^* \end{bmatrix}, \quad (4)$$

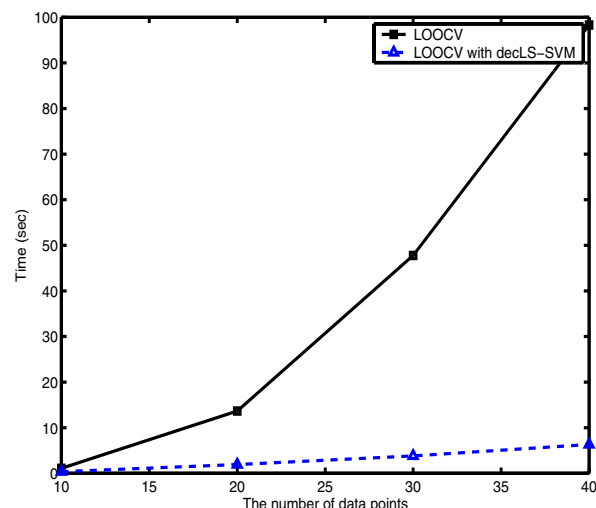
where  $\mathbf{y}^*$  is the updated  $\mathbf{y}$  vector and the matrices  $Q^*$  and  $R^*$  are obtained by updating and downdating the QR decomposition.

### 3. Results and Discussion

Four data sets, i.e. thyroid, diabetes, hearts, and titanic, were used. The data sets include 100 training and test splits, which are available from <http://ida.first.gmd.de/~raetsch/data/benchmarks.htm>.

Model selection is performed on the first five training splits and generalization is then measured by the mean error rate over the 100 test splits. The radial basis function (RBF) kernel,  $\mathbf{k}(\mathbf{a}_i^T, \mathbf{a}_j) = \exp(-\gamma\|\mathbf{a}_i - \mathbf{a}_j\|^2)$ , was used. For testing incremental classifiers, after training 75% of data points by LS-SVM, the remaining 25% data points were inserted one by one using incremental LS-SVM. The feasibility of incremental and decremental LS-SVM was confirmed by the observation that they generated the same solution vectors for 100 training/test splits for each data set. For drug design, the results of each biochemistry laboratory test on a new compound can be iteratively included in the training set. This procedure can further improve precision in order to select the next best predicted organic compound. Instead of retraining entire data points, it is much efficient to update solution by incremental LS-SVM.

The fifth data set was a data set used in the 2001 KDD cup data mining competition, which can be obtained from <http://www.cs.wisc.edu/~dpape/kddcup2001>. It consists of 1909 compounds tested for their ability to bind to a target site on thrombin, a key receptor in blood clotting. Of these compounds, 42 are active (bind well) and the others are inactive. Each compound is described by a single feature vector comprised of a class value (A for active, I for inactive) and 139,351 binary features, which describe three-dimensional properties of the molecule. In order to show the performance of the decremental LS-SVM, the computing times of leave-one-out cross validation (LOOCV)



**Figure 1. Computing times of leave-one-out cross validation (LOOCV) for different number of data points. The solid line presents computing time of ordinary LOOCV and the dashed line presents that of LOOCV using decremental LS-SVM.**

for various numbers of data points were observed by LS-SVM and decremental LS-SVM. Figure 1 shows that the LOOCV based on decremental LS-SVM is much more efficient. The Pentium III 600MHz computer was used to obtain the computing time for LOOCV algorithms which were implemented by MATLAB.

### 4. Conclusion

In this paper, an incremental and decremental LS-SVM is designed by updating and downdating the QR decomposition. It can efficiently compute the updated solution when data points are appended or removed.

### References

- [1] G. H. Golub and C. F. van Loan. *Matrix Computations, third edition*. Johns Hopkins University Press, Baltimore, 1996.
- [2] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.
- [3] A. Tveit, M. L. Hetland, and H. Engum. Incremental and decremental proximal support vector classification using decay coefficients. In *Proceedings of the 5th International Conference on Data Warehousing and Knowledge Discovery (DAWAK'2003)*, pages 422–429, Prague, Czech Republic, 2003. Springer-Verlag.