Data Reduction in Support Vector Machines by a Kernelized Ionic Interaction Model

Hyunsoo Kim and Haesun Park
Department of Computer Science and Engineering,
University of Minnesota, Twin Cities
{hskim, hpark}@cs.umn.edu

Abstract

A major drawback of support vector machines is their high computational complexity. In this paper, we introduce a novel kernelized ionic interaction (IoI) model for data reduction in support vector machines. We also present a data reduction method based on the kernelized instance based (KIB2) algorithm. We show that the computation time can be significantly reduced without any significant decrease in the prediction accuracy.

1 Introduction

When handling a large amount of data in machine learning, it is important to reduce the computation complexity and memory requirement without degrading the prediction accuracy. There have been several approaches proposed to overcome the computational difficulty and memory problems. Recently, for support vector machines (SVMs) [4, 8, 9], incremental learning has been proposed in order to handle huge data [6]. This repeats training a subset of the entire training data set, uses the support vectors found in the previous step, and merges these with what is found in iterative training.

Syes, *et al.* [7] selected a small subset of examples from the training set based on several selection strategies including the instance based IB2 selection method [1] and random sampling. They claim that the random sampling selection strategy is the most robust among the strategies they considered in terms of model independence i.e., with random sampling, training was successful with a wide range of different classifiers, such as multi-layer perceptron, nearest neighbor, C4.5, and SVMs. However, random selection method can sometimes cause a catastrophic decrease in accuracy.

In this paper, we propose three new methods for predicting an approximate set of support vectors in order to reduce computational complexity in SVMs. The first is based on an Ionic Interaction (IoI) model that utilizes concepts from Electrostatics for a binary decision. The second is a kernelized instance based (KIB2) method. The instance-based algorithm (IB2) [1] was already applied to select points in the input space [7]. Here, it is further developed it into a kernelized-IB2, where IB2 is applied in the high dimensional feature space based on the kernel. The third is a hybrid KIB2-IoI method where IoI is used as a complementary method to select data points in addition to the data points selected by KIB2. The test results show that with our methods, computational complexity in SVMs can be reduced substantially without significant decrease in prediction accuracy.

2 Ionic Interaction (IoI) Model and Data Reduction Methods

The support vectors in SVMs are the critical points near the boundary between the two classes, which determine an optimal separating hyperplane. Removing any training points that are not support vectors will have no effect on the hyperplane found [3]. If we can predict the support vectors before the training process, the computing time and the memory usage of SVMs will be reduced to an optimal separating hyperplane can be found by training a relatively smaller number of selected points instead of all of the given training points. For this purpose, we propose an Ionic Interaction (IoI) model for selecting the data points that are the candidates for the support vectors. The points in the (+) class and those in the (-) class are considered as plus ion ($\oplus$, *p-experon*) and minus ion ($\ominus$, *n-experon*), respectively, using a concept from Physics. These names, (*p-experon* and *n-experon*), are chosen since the data points can be considered as experience particles in a high dimensional parameter space of a human decision model based on experiences. The experience particles in the metaphysical space interact with each other like ions in the real physical space. The ions with the same sign repulse each other,

Algorithm 1 : IoI

Given m data points $\mathbf{x}_i$, $1 \le i \le m$, a kernel function K with kernel parameters, a desired selection percentage (ρ), and data selection threshold value (Φ_t), this algorithm finds an *approximate set of support vectors* with $|ASV| \le INT(m * \rho/100)$ using the ionic interaction (IoI) model.

1. Calculate potential energy for all data points $\mathbf{x}_i, 1 \le i \le m$, by
$$\Phi(i) = \sum_{j \in S_i} \frac{-1}{r_{ij}},$$
where S_i is the set of index j for all ions with the opposite sign to ith ion and $r_{ij}^2 = \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 = K(\mathbf{x}_i, \mathbf{x}_i) + K(\mathbf{x}_j, \mathbf{x}_j) - 2K(\mathbf{x}_i, \mathbf{x}_j)$.

2. For $c = p$ and n
   (a) Sort all data points in class c by potential energy in ascending order. Represent the sorted data points as $\mathbf{x}_k$ for $1 \le k \le l_c$, where l_c is the number of data points in class c.
   (b) $ASV_c \leftarrow \{\mathbf{x}_1\}$
   (c) For $k = 2$ to l_c
   if $|\Phi(k) - \Phi(k-1)| > \Phi_t$ then $ASV_c \leftarrow ASV_c \cup \{\mathbf{x}_k\}$
   else x_k is not a candidate for a support vector.
   (d) Choose data points which have lower potential energy to select given percentage (ρ) of l_c data points for a class. ($|ASV_c| \le INT(l_c * \rho/100)$).

3. $ASV = ASV_p \cup ASV_n$

and the repulsion makes them unstable. The ions with opposite sign attract each other, and the attraction makes them stable. The training data points which are adjacent to the boundary between the two classes tend to be stabilized by attraction between plus ions ($\oplus$) and minus ions ($\ominus$). The sample points which are far from the boundary are relatively unstable due to repulsion between adjacent ions with the same sign. The number of predicted sample points can be controlled by choosing a threshold of instability. Our data reduction method is derived based on the L_1 norm soft margin SVM which we first review briefly now. The dual formulation of SVMs with a kernel function K is

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C,\ i = 1, \ldots, n$$

where α_i, $1 \le i \le n$, denotes the optimal solution for the formulation. Then the decision rule is given by $sign(f(\mathbf{x}))$ with

$$f(\mathbf{x}) = \sum_{j=1}^{n} y_j \alpha_j K(\mathbf{x}, \mathbf{x}_j) + b, \tag{2.1}$$

where b is chosen so that $y_i f(\mathbf{x}_i) = 1$ for all i with $0 < \alpha_i < C$. According to the Karush-Kuhn-Tucker (KKT) conditions, optimal solutions α and $(\mathbf{w}, b)$ satisfy

$$\begin{aligned} \alpha_i[y_i(K(\mathbf{w}, \mathbf{x}_i) + b) - 1 + \xi_i] &= 0 \quad \text{and} \\ \xi_i(\alpha_i - C) &= 0, \quad i = 1, \ldots, n. \end{aligned}$$

These conditions can be rewritten as

$$\begin{aligned} &\text{Case 1:} \quad y_i f(\mathbf{x}_i) \ge 1,\ if\ \alpha_i = 0 \\ &\text{Case 2:} \quad y_i f(\mathbf{x}_i) = 1,\ if\ 0 < \alpha_i < C \\ &\text{Case 3:} \quad y_i f(\mathbf{x}_i) \le 1,\ if\ \alpha_i = C \end{aligned}$$

The second and third cases occur when $\xi_i = 0$ and $\xi_i > 0$ respectively. The slack variable can have a non-zero value only when $\alpha_i = C$. If $\alpha_i = 0$ (Case 1), then $\mathbf{x}_i$ is not a support vector. If $0 < \alpha_i < C$ (Case 2), then $\mathbf{x}_i$ is the support vector with $\xi_i = 0$. If $\alpha_i = C$ (Case 3), then $\mathbf{x}_i$ is the support vector with $\xi_i > 0$.

Now, we would like to eliminate data points that are not support vectors by dropping unstable ions which have relatively higher potential energy. In Case 2, the ions face the ions with the opposite sign over the separating hyperplane, so they are relatively stable. In Case 3, the ions are soaked into the opposite ionic pool, so they are very stable and will not be eliminated when unstable ions are dropped. In Case 1, the ions are surrounded by many with the same sign, so they are eliminated by the dropping process since they are relatively unstable.

After choosing training points which are considered as approximate support vectors by the ionic interaction algorithm (Algorithm 1), the separating hyperplane can be obtained by any variation of support vector machines. The approximate support vectors depend on the kernel type and kernel parameters. They are optimized to maximize prediction correctness of the test set for the iterative process.

3 Computation of Potential Energy and Data Reduction Algorithms

The Coulomb potential energy between the ith and jth ions is

$$V(i,j) = \frac{q_i q_j}{D \cdot r_{ij}} \tag{3.2}$$

where q_i and q_j are the charges of the ions, r_{ij} is the Euclidean distance between the ith ion and the jth ion, and D is a dielectric constant. Assuming the homogeneous dielectric medium with $D = 1$, the potential energy between the ith ion and other ions can be defined as

$$\Phi_o(i) = \sum_{j \neq i}^{n} \frac{q_i q_j}{r_{ij}} \tag{3.3}$$

where n is the total number of ions. The analogous potential energy between the ith data point and the rest which have been mapped to a higher dimensional feature space by a feature mapping $\phi(\mathbf{x})$ is

$$\Phi(i) = \sum_{j \in S_i} \frac{-1}{\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|} \tag{3.4}$$

where S_i is the set of index j of the ions with the opposite sign. Only the attraction is considered to select the points near the class boundary even though the ith ion may be surrounded by many same sign ions.

The distance r_{ij} between two data points in the feature space can be calculated by

$$\begin{aligned} r_{ij}^2 &= \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2 \\ &= K(\mathbf{x}_i, \mathbf{x}_i) + K(\mathbf{x}_j, \mathbf{x}_j) - 2K(\mathbf{x}_i, \mathbf{x}_j), \end{aligned} \tag{3.5}$$

assuming that the kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ which gives the inner products between the data points mapped by the feature mapping ϕ is given. This process requires about $n^2/2$ floating point operations (flops) [5] to obtain all pairwise distances r_{ij} between any two data points for $i > j$. Note that $r_{ij} = 0$ when $i = j$ and $r_{ij} = r_{ji}$ when $i < j$. When using the radial basis function (RBF) kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2)$, the distance between any two data points in the feature space becomes even more simple to compute as $r_{ij}^2 = 2 - 2K(\mathbf{x}_i, \mathbf{x}_j)$ since $K(\mathbf{x}_i, \mathbf{x}_i) = 1$ and $K(\mathbf{x}_j, \mathbf{x}_j) = 1$. If the potential energy $\Phi(i)$ is less than the *threshold potential value* Φ_s, then the ith ion is considered an approximate support vector since the ith ion is relatively stable compared to others. The reasonable range of Φ_s values can be found by numerical experiments considering various data sets. When selecting data according to potential energy, we did not choose a data point for which potential energy is too similar to that of the previous selected data point even though it has low enough potential energy to be chosen. We need only one representative point among many points that face a very similar environment for data reduction. In our experiments, we accept the data point only when the difference of potential energy between the data point and the previous selected data point is greater than the *data selection threshold value* (Φ_t).

Algorithm 2 : KIB2

Given m data points $\mathbf{x}_i$, $1 \leq i \leq m$, and a kernel function K with kernel parameters, this algorithm finds an *approximate set of support vectors* (ASV) in the feature space using the IB2 algorithm [1]. The distances between two data points $\mathbf{x}$ and $\mathbf{y}$ in the feature space are computed by $r(\mathbf{x},\mathbf{y})^2 = \|\phi(\mathbf{x}) - \phi(\mathbf{y})\|^2 = K(\mathbf{x},\mathbf{x}) + K(\mathbf{y},\mathbf{y})$ - $2K(\mathbf{x},\mathbf{y})$, where $\phi(\cdot)$ is the feature mapping such that $K(\mathbf{x},\mathbf{y})=\phi(\mathbf{x})^T\phi(\mathbf{y})$.

ASV $\leftarrow \{\mathbf{x}_1\}$
For each data point $\mathbf{x}_i$ for $i = 2, \ldots, m$

1. For each $\mathbf{y} \in ASV$ do
   Sim[$\mathbf{y}$] $\leftarrow 2K(\mathbf{x}_i, \mathbf{y}) - K(\mathbf{x}_i, \mathbf{x}_i) - K(\mathbf{y}, \mathbf{y})$
2. $\mathbf{y}_{max} \leftarrow \mathbf{y} \in ASV$ with maximal Sim[$\mathbf{y}$]
3. If class($\mathbf{x}_i$) $\neq$ class($\mathbf{y}_{max}$) then $ASV \leftarrow ASV \cup \{\mathbf{x}_i\}$

The IoI algorithm chooses data points that have a more different environment than a higher data selection threshold value. IoIB is a balanced selection around the percentage of data points required to balance the number of training points of binary classes. We added data points that have lower potential energy which had not already been included in the smaller class. Using the IoI model, we can expect that data points that have potential energy substantially lower than others, i.e. surrounded by many ions with opposite sign, are outliers. This is one of the advantages of the IoI model.

Our second approach is based on the boundary hunting algorithm, IB2 [1]. The approximate support vectors are found by IB2 instead of the ionic interaction (IoI) model in the feature space (See Algorithm 2). Although it has been observed that the selected data points in the input space by IB2 can hardly represent the support vectors that yield high prediction accuracy [7], we found that IB2 preforms well in the feature space obtained implicitly by a kernel, where the data points are assumed to be linearly separable. This approach is summarized in Algorithm 2 as kernelized IB2: KIB2.

Though the performance of KIB2 was good, it could not select additional data points that are able to improve the testing correctness since it only selected a predetermined number of data points. The ionic interaction model gave a rule to append data points to the selected data point by KIB2. The KIB2-IoI method is a combinational approach using the IoI model to append data points after

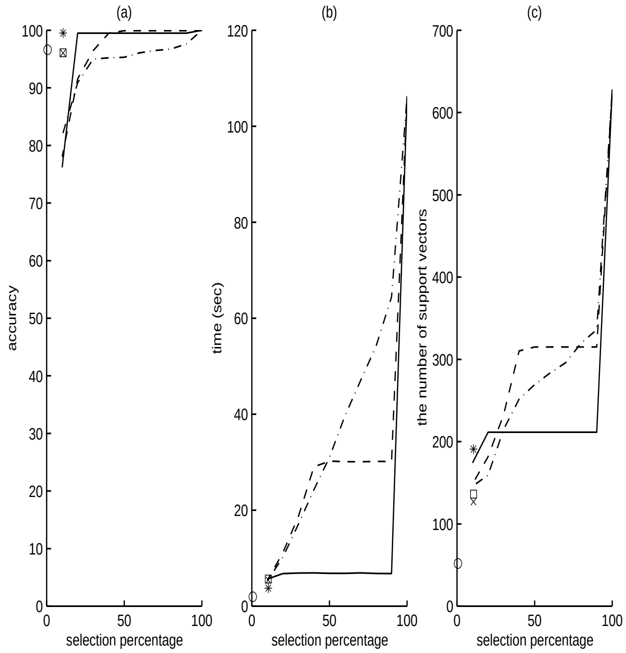

Figure 1: (a) Ten-fold cross validated accuracy (b) Computing time to obtain ten-fold cross validation accuracy. (c) The number of support vectors. All results were drawn against the percentage of selected data points on the UC Irvine 8124×22 Mushroom data set. The solid, dashed, and dashdot lines were the results of IoI for the selection threshold values Φ_t of 0.1, 0.01, and 0.0, respectively. The circle points were the results of KIB2. The star, square, and x-mark points were the results of KIB2-IoI with 10% additional data point for the selection threshold values Φ_t of 0.1, 0.01, and 0.0, respectively.

obtaining candidate support vectors by KIB2 in order to generate a more accurate optimal separating hyperplane with a minimal number of selected points among the input data. The performance of all introduced algorithms depends on the input kernel parameter set. We have to optimize the kernel parameters in order to achieve better data reduction performance producing higher prediction accuracy in the classification problem by trying various parameter sets.

4 Results and Discussion

The numerical results in Table 1 were achieved on UC Irvine test problems [2]: Ionosphere, BUPA Liver, Pima Liver, Cleveland Heart, and Mushroom to compare variations of the IoI data selection algorithm in terms of ten-fold cross-validation testing correctness. The IoIB algorithm generally showed good performance compared to IoI. This means that balancing the data points in the selected data points is also helpful to obtain a more generalized optimal separating hyperplane.

Table 1 shows the ten-fold cross validation accuracy and computing time to train and test for ten folds to compare the optimal ionic interaction (IoI) algorithm, the kernelized instance based (KIB2) algorithm, and the hybrid (KIB2-IoI) algorithm. The number of 50% data points are approximately similar to the number of support vectors that was found after training with the full number of data points, except the Ionosphere and Mushroom data sets. For the Mushroom data set, the percentage of the points predicted as support vectors is mush less than 50% though the desired selection percentage was 50% in the IoI algorithm. By selecting representatives from each group that have a similar ionic environment, we achieved data reduction using the IoI model. Figure 1 shows that an appropriate data selection threshold value enables the IoI algorithm to find a smaller number of data points without any significant decrease in the prediction accuracy.

The ionic interaction methods tend to need more data points than KIB2 to obtain a similar level of accuracy. This tendency can be seen in Figure 1 again. Therefore, the computing time of KIB2 is generally much smaller than that of IoI. KIB2 only needs about 0.6% of the data points to produce 96.61% testing correctness for the Mushroom data set. However, IoI is a more general tool if we want to obtain better prediction accuracy than KIB2 since it can be used for balancing selected data points and exclusion of outliers. For IoI, the accuracy slowly decreased until 50% selection, while the computing time dramatically decreased and the number of support vectors decreased only a small amount as shown in Figure 1. The number of support vectors of KIB2 is much smaller than that of IoI. KIB2 can select more compact data points that can be meaningful support vectors. However, we observed that the KIB2 data selector performed poorly due to the characteristics of the IB2 boundary hunting algorithm when the data points are too unbalanced. The IoI data selector can be used to exclude outliers. The hybrid KIB2-IoI data selector is a more reliable data reduction method compared to the KIB2-based approach, especially for unbalanced data. However, we would like to point out that KIB2 is still a comparable data selector when the data are well balanced.

In this paper, we applied the ionic interaction concept of Physics to the data reduction problem by considering data points of positive and negative classes as charged particles, i.e. *p-experons* and *n-experons* respectively. An experience can be represented as a charged particle in the

Data set points × features	Full data set time(sec), (SV†)	IoI time(sec), (ASV‡)	KIB2 time(sec), (ASV‡)	KIB2-IoI time(sec), (ASV‡)
Cleveland Heart 297×13	85.18% 1.69, (115.6)	83.18% 0.80, (128.5)	82.83% 0.97, (82.7)	80.48% 1.23, (108.7)
BUPA Liver 345×6	74.50% 4.39, (209.2)	73.62% 0.99, (159.5)	71.57% 1.42, (139.7)	73.01% 1.85, (170.7)
Ionosphere 351×34	95.20% 2.79, (172.4)	94.31% 1.52, (161.8)	92.26% 0.70, (55.7)	94.59% 0.97, (88.4)
Pima Indians 768×8	77.34% 11.54, (405.9)	72.79% 6.63, (300.1)	75.51% 2.57, (235.1)	76.16% 4.13, (325.1)
Mushroom 8124×22	100.0% 81.88, (628.1)	99.51% 6.85, (797.4)	96.61% 1.94, (52.9)	99.53% 6.34, (730.0)

Table 1: Ten-fold cross-validation testing correctness (%) on UC Irvine test problems: Ionosphere, BUPA Liver, Pima Liver, Cleveland Heart, and Mushroom. The execution times are the computed time to obtain ten-fold cross validation accuracy. SV† represents the average number of support vectors for ten-folds. ASV‡ represents the average number of data points predicted as support vectors. IoI: results with the desired selection percentage 50% of the data points which have lower potential energy by IoI model. KIB2: results with the selected data points by KIB2. KIB2-IoI: results with the data points of selected data points by KIB2 and 10% additional data points by IoI model.

metaphysical space of a human binary decision model. Through numerical experiments, the near boundary data points that have a close data point with the opposite sign are more important than the other data points for classification. We conjecture that the experons near boundary in the mapped high dimensional feature space are used for human binary decision. Even though support vector machine is a good mathematical model for a binary decision, we can hardly conjecture how it works in brain operated by delicate electric signals. If it is possible to abstract interactions of experons from interactions of neurons by electric signals when experons are expressed by local neural networks, depicting the representation of experons and their interactions in neural networks would be a challenging issue. Though the computational complexity of the proposed IoI algorithm is still high, we proposed more computationally efficient KIB2 algorithm for the same purpose to find out near boundary data points in the feature space. Further study contains developing new boundary hunting algorithms and a parallel algorithm of the IoI model, which can efficiently compute the potential energies.

Acknowledgements

We would like to thank Dr. D. W. Aha for his kind help.

References

[1] D. W. Aha, D. Kibler, and M. Albert. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.

[3] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[4] N. Cristianini and J. Shawe-Taylor. *Support Vecor Machines and other kernel-based learning methods*. University Press, Cambridge, 2000.

[5] G. H. Golub and C. F. Van Loan. *Matrix Computations, third edition*. Johns Hopkins University Press, Baltimore, 1996.

[6] N. A. Syed, H. Liu, and K. K. Sung. Incremental learning with support vector machines. In *Proceedings of the Workshop on Support Vector Machines at the International Joint Conference on Articial Intelligence (IJCAI-99)*, Stockholm, Sweden, 1999.

[7] N. A. Syed, H. Liu, and K. K. Sung. A study of support vectors on model independent example selection. In *Proceedings of the Fifth ACM Conference on Knowledge Discovery and Data Mining*, pages 272–276, 1999.

[8] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[9] V. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.