An Inductive Semi-supervised Algorithm for BCIs

Xilan TIAN*, Gilles GASSO*, Stéphane CANU*

*LITIS, INSA de Rouen, St Etienne du Rouvray, France

Correspondence: Xilan TIAN, LITIS EA 4108, INSA de Rouen, St Etienne du Rouvray, France.
E-mail: xilan.tian@gmail.com, phone +33 6 4 226 8653, fax +33 2 3 2959 903

Abstract. Semi-supervised learning has a successful application in Brain Computer Interfaces (BCIs) research. Current research always handle the learning problem in a transductive way which means that the classifier won't work well in the out-of-sample case. In this paper, we present a multi-kernel transductive SVM, named TSVM-MKL, to attain an inductive inference model for BCIs system. Next, we test the obtained model in the out-of-sample case on an EEG-based cursor control experiments. Experimental results show the effectiveness of proposed algorithm.

Keywords: Semi-supervised Learning, TSVM-MKL, BCI

1. Introduction

In Brain Computer Interfaces (BCIs), labeled data are expensive or time-consuming to obtain. With semi-supervised learning, the learning process can be taken as the labeling process for BCIs. [Qin et al., 2007] proposed a semi-supervised SVM algorithm to reduce the computational burden. [Zhong et al., 2009] used a Laplacian SVM (LapSVM) to reduce the subjects' training complexity. However, instead of the transductive behavior of the aforementioned algorithms, we prefer to obtain an inductive model that works well in the out-of-sample case. In this paper, we resort to transductive SVM (TSVM) whose decision function lies in the span of kernel functions to handle the out-of-sample case. We also adopted multi-kernel framework to modify the decision function from both cluster and manifold perspective. Empirical results demonstrate the potential of proposed algorithm is more effective than the single assumption algorithms in EEG based BCIs.

2. Multiple Kernel TSVM

In this section, we formulate the TSVM-MKL to be an inductive model for BCI research. For a data set \( \{x_1, \cdots, x_n\} \), the first \( l \) samples are labeled \( \{y_1, \cdots, y_l\} \), and followed by \( u \) unlabeled ones. We adopt the framework of [Rakotomamonjy et al. 2008] to extend TSVM to the multi-kernel case:

\[
\frac{1}{2} \sum_{k=1}^{m} d_k \|f_k\|_{H_k}^2 + C \sum_{i=1}^{l} H_1(y_i g(x_i)) + C^* \sum_{i=l+1}^{l+u} R_1(y_i g(x_i))
\]

(1)

\[\text{s.t. } \sum_{k=1}^{m} d_k = 1, \quad d_k \geq 0 \quad \forall k = 1, \ldots, m. \quad \frac{1}{u} \sum_{i=l+1}^{l+u} g(x_i) = \frac{1}{l} \sum_{i=1}^{l} y_i\]

Decision function is \( g(x) = \sum_{k=1}^{m} f_k(x) + b \), where \( f_k \) are functions determined by kernel matrix in different Hilbert spaces. \( H_1(z) = \max(0, 1 - z) \) is Hinge loss, \( a_k \) is a normalization term, \( d_k \) acts as the selector of appropriate kernels, \( R_1(z) = H_1(z) - H_1(z) \) is Ramp loss. This TSVM-MKL inherits the non-convexity of TSVM which is related to Ramp loss. To circumvent this drawback, we employ DC programming to solve this non-convex problem. Basically, the procedure amounts to linearizing \( R_1(z) \) around the current solution. Replacing \( R_1(z) \) by its affine minorization and plugging it in (1) gives rise to a convex MKL problem which can be solved by any off-the-shelf MKL algorithms.
algorithm. Finally, we attain the decision function for a new sample \( x \) (with new notation on \( \alpha_i, y_i, \) and \( \gamma_i \)) :

\[
g(x) = \text{sgn} \left( \sum_{k=1}^{m} d_{yk} \sum_{i=0}^{l+2u} \left( \alpha_i y_i + C^e \gamma_i \right) K_i(x,y) \right) + b
\]

(2)

3. Experimental Analysis

We apply TSVM-MKL on an EEG-based cursor control experiment. The data set was recorded from 3 subjects (AA, BB, CC). Each subject's data included 10 sessions. Each session consists of 192 trials. To compare with the method of [Qin et al., 2007], only the trials with the targets who are at the highest and lowest position of the right edge of the screen (96*10 trials for each subject) were used in our analysis. Data set from session 1-6 were used for learning, and those from session 7-10 acted as the out-of-sample set for test. We first extracted the dynamic common spatial patterns (DCSP) (proposed by [Qin et al., 2007]) for session 1-6. Next, we divided all the features into labeled set and unlabeled set. Labeled set consisted of 48 trials from session 1, the rest served as unlabeled set. Based on the transformation matrix obtained on learning set, the DCSP features for test set are attained.

For TSVM-MKL, we defined a pool of gaussian kernels: the number of neighbor points [40], deformed ratios [Sindhwani et al., 2005] [10 1000] and kernel options [1 10 100 1000]. Finally, we attained a TSVM-MKL with 9 kernels. Model selection is based on the 5-fold cross validation error on unlabeled set. Table 1 shows the best results of TSVM-MKL, Laplacian SVM and TSVM on the same data. Results of Semi-supervised SVM were from [Qin et al., 2007]. They first separated the unlabeled set into 9 subsets. And then trained a linear SVM using the dynamic power features from labeled set, estimated labels of the following sub-unlabeled set, selected the most confidently classified elements and added them together with their predicted labels to the training set for a 1-norm semi-supervised classifiers. Repeated this procedure until enough estimated labels are available to calculate DCSP features. The transformation matrix and classifier's model were updated in each loop. In this degree, although our results are superior to their's, we use some label information in DCSP feature extraction procedure.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>AA</th>
<th>BB</th>
<th>CC</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM ([Qin et al., 2007])</td>
<td>94.52</td>
<td>91.84</td>
<td>91.51</td>
<td>92.62</td>
</tr>
<tr>
<td>Laplacian SVM</td>
<td>96.36</td>
<td>93.75</td>
<td>94.17</td>
<td>94.76</td>
</tr>
<tr>
<td>TSVM</td>
<td>97.40</td>
<td>94.79</td>
<td>95.37</td>
<td>95.37</td>
</tr>
<tr>
<td>TSVM-MKL</td>
<td>97.68</td>
<td>96.35</td>
<td>95.77</td>
<td>96.60</td>
</tr>
</tbody>
</table>

4. Conclusion

By combing the manifold and cluster assumption in the framework of multi-kernel, we attained an inductive model to handle the out-of-sample case in BCI research. According to our algorithm, all the unseen samples can be labeled and classified in an inductive way. This characteristic make it possible to realize an effective on-line BCI system.

References

Qin J, Li Y, Sun W. A semisupervised support vector machines algorithm for BCI systems, in Computational Intelligence and Neuroscience. 2007, 1-12.


