

Pittsburgh Brain Activity Interpretation Competition 2006

Methods Description

Maximum Margin Regression using KCCA Feature Projection

Project Abstract

In the following competition we use a maximum margin framework realizing a regression type learning in an arbitrary Hilbert space whilst the corresponding dual problem preserves the structure and, therefore, the complexity of the binary Support Vector Machine (SVM). The novelty of this approach is that we are able to solve the multi-class classification/regression problem with the complexity of a binary SVM. We further enhance the maximum margin framework by first computing the Canonical Correlation Analysis (CCA) in kernel space. This aims to learn a semantic representation of fMRI brain scans and their associated activity (represented as multi-label representation). We find that using the feature direction from the Kernel Canonical Correlation Analysis (KCCA) can deliver better performance than using the maximum margin method alone. In our second submission we use the KCCA also as a voxel ROI selection approach, by excluding all the voxels with small weights across all the selected feature directions.

Introduction

In the following competition we have opted to address the problem as a multi-label prediction/regression problem. Commonly multi-label problem are considered to be extremely hard due to their taxing computation time and/or the need for reducing the multi-label problem into a multitude of smaller two-class problems. In our work we have opted on using the Maximum Margin Robot (MMR)[1], a novel vector label based learning method. The MMR offers the attractive ability of learning a multi-label problem at the computational complexity of a binary problem. Due to the large amount of features existing in the data we aim apply a feature selection approach by obtaining feature directions using Kernel Canonical Correlation Analysis [2] algorithm. We then use the MMR in conjunction with the KCCA features. We try a further to reduce the number of features used in training by removing voxels that were found to be of small significance (smaller correlation weightings).

Method

The Support Vector Machine (SVM) has been shown to be a very useful method of machine learning, but is restricted to directly solving binary classification problems only. There is a strong demand for extending the underlying idea towards multi-class classification and learning when the outputs have complex structure. The known approaches are tackling with the exploding computational complexity and the range of potential applications becomes very limited. There is a straightforward algebraic generalization of the SVM, which can handle arbitrary vector outputs and preserves the same computational complexity of its binary ancestor (Figure 2). The structural learning problems can then be solved via an embedding into a properly chosen vector space. The

learning strategy in the vector label learning can be stated as a three-phase process;

Embedding: where the structures of the input and output objects are represented in properly chosen Hilbert spaces, reflecting the similarity and the dissimilarity of the objects.

Optimization: has to find the similarity based matching between the input and the output representations,

Inversion : has to recover the best fitting output structure of its vector representation.

We refer the reader to [1] for more information on the MMR.

Table 1. SVM and MMR interpretation

Binary class learning Support Vector Machine (SVM)	Vector label learning Maximum Margin Robot (MMR)
$\min \frac{1}{2} \frac{\boxed{\mathbf{w}^T \mathbf{w}}}{\ \mathbf{w}\ _2^2} + C \mathbf{1}^T \boldsymbol{\xi}$	$\frac{1}{2} \frac{\boxed{\text{tr}(\mathbf{W}^T \mathbf{W})}}{\ \mathbf{W}\ _{Frobenius}^2} + C \mathbf{1}^T \boldsymbol{\xi}$
$\text{w.r.t. } \boxed{\mathbf{w} : \mathcal{H}_\phi \rightarrow \mathbb{R}}, \text{ normal vec.}$ $\boxed{b \in \mathbb{R}}, \text{ bias}$ $\boldsymbol{\xi} \in \mathbb{R}^m, \text{ error vector}$	$\boxed{\mathbf{W} : \mathcal{H}_\phi \rightarrow \mathcal{H}_\psi}, \text{ linear operator} \quad (2)$ $\boxed{\mathbf{b} \in \mathcal{H}_\psi}, \text{ translation (bias)}$ $\boldsymbol{\xi} \in \mathbb{R}^m, \text{ error vector}$
$\text{s.t. } \boxed{y_i(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_i) + b)} \geq 1 - \xi_i$ $\boldsymbol{\xi} \geq \mathbf{0}, i = 1, \dots, m$	$\boxed{\langle \boldsymbol{\psi}(y_i), \mathbf{W} \boldsymbol{\phi}(\mathbf{x}_i) + \mathbf{b} \rangle_{\mathcal{H}_\psi}} \geq 1 - \xi_i$ $\boldsymbol{\xi} \geq \mathbf{0}, i = 1, \dots, m$

We use the MMR on the KCCA obtained directions;

Proposed by Hotelling in 1936, Canonical Correlation Analysis (CCA) is a technique for finding pairs of basis vectors that maximize the correlation between the projections of paired variables onto their corresponding basis vectors. Correlation is dependent on the chosen coordinate system, therefore even if there is a very strong linear relationship between two sets of multidimensional variables this relationship may not be visible as a correlation. CCA seeks a pair of linear transformations one for each of the paired variables such that when the variables are transformed the corresponding coordinates are maximally correlated. The kernelising of CCA offers an alternate solution by first projecting the data into a higher dimensional feature space before performing CCA in the new feature space. For more detailed information on CCA and KCCA we refer the reader to [2].

The regression on the KCCA obtained features could be thought of a Partial Least Squares (PLS) regression on the labels. We have tried this and have found that the PLS would achieve slightly worse results than the MMR+KCCA.

First Submission: Linear kernels have been used for both the MMR and KCCA methods with a default plenty parameter of one. The KCCA regularisation parameter was set to 0.9. We have

projected the training data (and then later the testing) into the largest 200 feature KCCA direction and have applied the MMR regression to predict the 30 labels.

Second Submission: In this submission we have done as in the first submission but prior to the described stages we have run the KCCA (with the same configuration) and have computed the weights for the fMRI training data for the largest 200 KCCA directions. Using the weight values we would test each voxel (the weight can be thought of a $__ \times 200$ matrix where $___$ is the number of overall voxels in the entire brain) across the 200 directions. We then remove any voxel with a value, across the 200 directions, smaller than a fixed threshold.

Results and Discussion

Initially, we trained our method to minimize the L2-norm from the true labels. We found that while obtaining a low error on the L2-norm our method did not perform as well when using the Feature Rater application provided. Therefore we had modified the training criterion to maximize the Feature Rater score output. During our initial runs on Movie 1 and Movie 2 we had obtained an average score of 0.27.

In the first submission we have obtained using our method a score of 0.310 when using 200 feature KCCA feature directions and a default plenty parameters of one. In our second submission we had obtained a score of 0.309 (with a similar configuration as submission one). We mention the second submission, even though its lower score, as it was achieved using less than half of the overall voxels in the fMRI data using the voxel elimination preprocessing as described in the method section.

We believe that further working with ROI or voxel elimination and optimization over the distance from the true labels may achieve better performance. There is no neurological analysis on our results nor did we further investigate on the remaining voxel in the second attempt.

References (not part of word count)

[1] Sandor Szedmak, John Shawe-Taylor and Emilio Parado-Hernandez “Learning via Linear Operators: Maximum Margin Regression; Multiclass and Multiview Learning at One-class” Complexity <http://www.ecs.soton.ac.uk/~ss03v/mmr.html>

[2] David R. Hardoon, Sandor Szedmak and John Shawe-Taylor “Canonical Correlation Analysis: An Overview with Application to Learning Methods” Neural Computation, vol. 16 pp. 2639-2664, 2004.

Optional Comment on Competition (not part of review)

What suggestions might you have for next year’s competition?

Explaining or providing further information on the usage and application of the methods common in neuroscience but not in, for example, computer science. The usage of ROI on post filtered data, etc...