Multi-class extensions of the GLDB feature extraction algorithm for spectral data

Pavel Paclík, Serguei Verzakov, Robert P.W.Duin

Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
{P.Paclik, S.Verzakov, R.P.W.Duin}@ewi.tudelft.nl

Abstract

The Generalized Local Discriminant Bases (GLDB) algorithm proposed by Kumar, Ghosh and Crawford in [4], is a effective feature extraction method for spectral data. It identifies groups of adjacent spectral wavelengths and for each group finds a Fisher projection maximizing the separability between classes. The authors defined GLDB as a two-class feature extractor and proposed a Bayesian Pairwise Classifier (BPC) building all pairwise extractors and classifiers followed by a classifier combining scheme. With a growing number of classes the BPC classifier quickly becomes computationally prohibitive solution. In this paper, we propose two alternative multi-class extensions of GLDB algorithm, and study their respective performances and execution complexities on two real-world datasets. We show how to preserve high classification performance while mitigating the computational requirements of the GLDB-based spectral classifiers.

1. Introduction

This paper investigates multi-class extensions of a two-class feature extraction algorithm for hyperspectral data, proposed by Kumar, Ghosh and Crawford in [4]. Their contribution is twofold. Firstly, they proposed a feature extraction technique for two-class hyperspectral datasets, called Generalized Local Discriminant Bases (GLDB). It exploits apriori information on wavelength ordering and maximizes class separability using the Fisher ratio. Secondly, they developed a Bayesian Pairwise Classifier (BPC) based on the GLDB feature extractor. For each pair of classes, a classifier is built in a problem-specific feature space, extracted by a GLDB algorithm. Results of all the pairwise classifiers are then fused by a classifier combiner. Although this pairwise classification framework delivers high accuracy results, a serious obstacle to its practical usage lays in its computational demands. Each new spectrum to be labeled is subjected to all pairwise feature extractors and classifiers. The computational complexity of this setup is quadratic with respect to the number of classes. Taking into account the large number of samples processed in a typical hyperspectral imaging application, this execution methodology poses an obstacle for practical applicability of the GLDB feature extractor.

In this paper, we focus on mitigating the execution complexity of a GLDB-based multi-class classifiers. We propose two alternative techniques for refining data representation using a GLDB algorithm. We show, that a conventional multi-class classifier, built on such a representation reaches performance comparable with the BPC classifier but is significantly faster in the execution stage.

By performing a classification using a single multi-class classifier in a single, specially constructed, feature space we take a different direction as opposed to the GLDB authors in their recent work [5]. There, they construct a hierarchy of multiple two-class GLDB extractors and classifiers in a clustering-like fashion.

Our first proposal is to build all pairwise GLDB extractors similarly to BPC framework. However, instead of building also all the pairwise classifiers, we employ a second stage feature selection to find more compact representation for a conventional multi-class classifier. This significantly reduces the amount of computations when labeling new spectra.

Our second proposal is to use *a multi-class criterion* inside the GLDB feature extractor. Thereby, a feature representation is refined which directly maximizes the separability of all classes.

In the following sections, we explain the two-class GLDB feature extractor and the BPC classifier respectively. Then, we describe both methods, proposed by us: the feature selection over the pairwise GLDB representations and the GLDB feature extractor with a multi-class criterion. Experimental results follow in the section 6. Finally, we give conclusions and recommendations for practitioners.

2. Two-class GLDB feature extractor

In this section, we give short explanation of the GLDB feature extraction algorithm, as defined in [4]. A two-class dataset is given which contains spectral measurements with D wavelengths. Using this data, the GLDB algorithm identifies K groups of adjacent wavelengths (bases) $G_k = [l_k, u_k]$, where l_k and u_k represent lower and upper wavelength indices of the k-th group, respectively, and $1 \leq l_k \leq u_k \leq D$.

The groups are defined by a search algorithm maximizing a two-component criterion

$$\mathrm{crit}(l,u) = \mathcal{C}(l,u)\mathcal{D}(l,u) \tag{1}$$

The first part represents the minimum correlation between the wavelengths in the group k:

$$\mathcal{C}(l,u) = \min_{l \leq r < s \leq u} q_{r,s}. \tag{2}$$

Symbol $q_{r,s}$ denotes the correlation coefficient between the wavelengths r and s. Using this min-max approach, highly correlated wavelengths (often the neighboring ones) become favored candidates for merging. The second component of the criterion reflects the class separability measured by the Fisher ratio:

$$\mathcal{D}(l,u) = \frac{\mathbf{w}_{l,u}^T \mathbf{B}_{l,u} \mathbf{w}_{l,u}}{\mathbf{w}_{l,u}^T \mathbf{W}_{l,u} \mathbf{w}_{l,u}}. \tag{3}$$

Symbols $\mathbf{B}_{l,u}$ and $\mathbf{W}_{l,u}$ denote the between-class and the within-class scatter matrices respectively employing the wavelengths r, $l \leq r \leq u$.

Two variants of the GLDB algorithm are developed in [4]: the top-down and the bottom-up search. In this study, we use the bottom-up algorithm which appears to reach higher performance than its top-down counterpart. The bottom-up search starts from all singleton wavelengths and iteratively merges the adjacent bands maximizing the above mentioned criterion. The algorithm is stopped when no merge candidate yields the criterion increase.

A by-product of the GLDB optimization is a Fisher projection vector $\mathbf{w}_k$, maximizing the class separability for the k-th group. Each group of wavelengths is projected via the corresponding Fisher projection $\mathbf{w}_k$ into a 1D Fisher space (maximum possible dimensionality in the two-class case). In this way, the extractor ψ maps the spectra from a D dimensional space of wavelengths into a K-dimensional output space Y, $\psi : X \rightarrow Y$, $K \leq D$. Because only some of the new features y_k convey discriminative information, the authors propose to choose those by a feature selection step. If the feature selection yields a S-dimensional feature subset, $S \leq K$, the resulting GLDB feature extractor is a mapping $\psi : X \rightarrow Z$, $Z \in R^S, Z \subset Y$, $\psi = \{\mathbf{w}_s\}$, $s = 1, ..., S$.

3. Bayesian pairwise classifier

The Bayesian pairwise classifier (BPC), proposed in [3] is based on an idea that each pair of classes should be separated by a classifier, trained in a feature space, specifically constructed for the given problem. Therefore, for a C-class problem, $C > 2$, Kumar *et.al.* propose to construct $\binom{C}{2}$ feature extractors $\psi_{i,j}$ and corresponding classifiers $\phi_{i,j}$. The class indices i and j run over $1 \leq i < j \leq C$. When run on new spectra, all the feature extractors and classifiers are executed yielding a set of $\binom{C}{2}$ labels or estimates of posterior probabilities. A classifier combiner collates the estimates and makes the final decision. The execution complexity of the BPC classifier is $\mathcal{O}(C^2)$.

Apart from the majority voting combiner, Kumar *et.al* also discuss a sophisticated classifier combining technique, proposed by Hastie and Tibshirani [2]. They conclude that although the method produces slightly better results than the voting scheme, its execution is even slower due to its iterative nature. Because of our focus on fast execution, we considered only the majority voting combiner in BPC.

4. Feature selection over the pairwise GLDB representations

In this section we outline the first method, proposed by us. The reasoning behind the BPC framework is that "*Feature extractors for specific groups of classes should be determined separately.*" (see [4], page 1369). The BPC classifier therefore builds all pairwise feature extractors and corresponding classifiers. We think that, due to high redundancy, only a small subset of the extracted features may be neccessary to successfully separate the classes. Therefore, our proposal is to run the feature selection algorithm *on the concatenated set of all features generated by the pairwise extractors*. Because our eventual goal is to build a single multi-class classifier, we use the performance of such a classifier on the validation set as a selection criterion. We used a quadratic Bayesian classifier, assuming normal densities because classes of Gaussian-like shape are often generated by a GLDB feature extractor (see Figure 1). The feature selection procedure is stopped when the performance increase gets smaller then the user-specified threshold (e.g. 1%).

The execution complexity of a multi-class classifier built in a single feature space is $\mathcal{O}(C)$, because C class-models must be evaluated.

Please note that concatenating all the pairwise feature representations does not necessarily lead to excessively high-dimensional datasets. This is because each GLDB extractor is already bundled with a feature selector as described in section 2. We discuss this point further in the experimental section.

5. Multi-class criterion for GLDB feature extractor

In this section, we briefly describe our proposal to use a multi-class criterion directly in the GLDB extractor. For more detailed discussion on a general multi-class Fisher criterion, see [7]. To simplify the computations, the columns (wavelengths) of the data matrix are first transformed to be orthonormal. A pooled between-class scatter matrix for the k-th group (i.e. for wavelengths $l \leq r \leq u$) is estimated using:

$$\mathbf{B}_{(l,u)} = \sum_{1 \leq i < j \leq C} p_i p_j (\mathbf{m}_i - \mathbf{m}_j)^T (\mathbf{m}_i - \mathbf{m}_j), \qquad (4)$$

where $\mathbf{m}_i$ represents a mean vector of class C_i in the projected space and p_i the prior probability of class C_i. By solving a generalized eigenvalue problem for matrix $\mathbf{B}_{(l,u)}$, we obtain the Fisher projection vector $\mathbf{w}_k^{\text{multi}}$.

Note that the maximum output dimensionality of the mapping $\mathbf{w}^{\text{multi}}$ is $L = \min(|G_k|, C) - 1$, where $|G_k|$ is the number of wavelengths in the group G_k. We use the sum of the L largest eigenvalues as a measure of the Fisher separability $\mathcal{D}_{\text{multi}}(l, u)$ which may be plugged in the GLDB criterion (1).

Similarly to the two-class GLDB algorithm, a feature selection step determines a subset of features, forming a multi-class GLDB extractor.

6. Experiments

We have performed experiments on two real-world hyperspectral datasets. The first set contains 16 classes of NIR spectra of *plastics* acquired with InGaAs-camera using spectral range between $1\mu m$ and $1.6\mu m$ sampled into 120 bands. From a hyperspectral image we labeled 5989 spectra using the ground-truth information. We chose independent training and test sets with 3200 and 2789 data samples, respectively. The data was preprocessed by baseline subtraction, Savitsky-Golay smoothing and normalized to a unit area.

The second set is a Washington *DC-Mall* dataset acquired by an airborne sensor [6]. It contains 7 classes of IR spectra with 191 spectral wavelengths. We have used the training set with 1400 spectra and the independent test set with 8569 spectra. The spectra were normalized to a unit area.

For each problem, we train all the three discussed methods on the training set and estimate their performances using the test set. We used quadratic Bayes classifier assuming normal densities for building of all the pairwise classifiers in BPC and also for the multi-class classifiers in both proposed methods.

Table 1 summarizes the experimental results. The column *dim* shows the dimensionality used by different methods. For the original BPC classifier, it is the total number

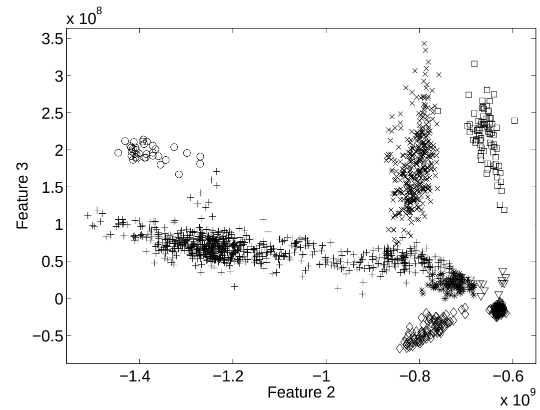

Figure 1. 2D scatter plot of DC-Mall dataset. Features were found from all pairwise GLDB representations by a feature selection. Different classes are distinguished by markers.

of wavelength groups, computed by all pairwise GLDB extractors. In adition to the original pair-wise feature count, the size of a selected subset is given in case of the feature selection method. Also for the multi-class GLDB criterion (MC-GLDB), two results are shown: the number of features extracted by the GLDB method, and the final subset of these features derived by feature selection. During the feature selection, half of the training data is used to train a quadratic Bayes classifier. Its accuracy on the rest of training data was used as a selection criterion. The classification error on the independent test set is given in the third column. The remaining columns show the number of operations needed in order to extract the features and apply the classifier to a new spectrum, respectively. The total number of operations is given in the last column. Our test implementation is based on is based on PRTools [1] and Hypertools [8] toolboxes.

Our experiments show, that for both datasets comparable or better accuracies were reached using a single multi-class classifier than employing the BPC combiner. As Table 1 shows, the number of operations needed to extract feature representation may be dramatically reduced by both proposed approaches. The computational complexity of the multi-class classifier is, however, surprisingly high compared with the BPC classifier.

For example, the 16-class plastic problem requires to build $\binom{16}{2} = 120$ feature spaces and consequently 120 classifiers. Table 1 shows that all the pairwise GLDB extractors derive in total 124 features. That means that almost all pairwise classifiers reduce to computationally cheap thresholding in $1D$ spaces. On the other hand, both proposed approaches derive multi-dimensional feature spaces. For the used quadratic Bayes classifier assuming normal densities, the computational complexity of classification grows quadratically with increasing number of features. There-

plastic dataset (16 classes, 87 wavelengths):

method	*dim.*	*error [%]*	*ops.feat. extract.*	*ops. classif.*	*ops total*
orig.BPC	124	0.4	2344	1640	3984
feature selection	124→4	1.7	196	672	868
	124→5	0.7	262	992	1254
	124→7	0.2	318	1824	2142
MC-GLDB	59→6	1.8	66	1376	1443
	59→9	0.7	118	2912	3030
	59→14	0.4	198	6752	6950

DC-Mall dataset (7 classes, 191 wavelengths):

method	*dim.*	*error [%]*	*ops.feat. extract.*	*ops. classif.*	*ops total*
orig.BPC	58	1.7	1880	4472	6352
feature selection	58→6	1.9	624	602	1226
	58→7	1.6	626	798	1424
	58→15	1.2	1038	3374	4412
MC-GLDB	77→ 6	1.8	212	602	814
	77→10	1.1	352	1554	1906
	77→15	1.0	560	3374	3934

Table 1. Experimental results.

fore, the multi-class classifier with 7 features becomes more computationally expensive than all 120 simple pairwise classifiers combined by the majority vote combiner. Because of that, the proposed methods become especially interesting for small feature sizes. Then, comparable or slightly worse accuracies may be reached for only a fraction of operations compared to the original BPC method.

It follows from our experiments, that the multi-class GLDB criterion generates larger number of features than the feature selection applied to the pairwise representations. While a two-class extractor derived on average 1.03 features on plastic dataset, the multi-class GLDB using all 16 classes extracted 59 features. In case of DC-Mall dataset with 7 classes, each of the 21 pairwise extractors used on average 2.8 features. The multi-class GLDB extracted 77 features. This is an intuitive outcome because the multi-class criterion optimizes several problems at once.

7. Conclusions

In this paper, we discussed multi-class classifier utilizing the GLDB feature extraction technique for spectral data. Our goal was to limit the computational requirements in the execution stage. We have investigated two strategies for extraction of a multi-class representation using the GLDB algorithm. The first one builds all the pairwise feature representations in the same way as the BPC classifier. However, only the relevant subset of features are selected by a second stage feature selection. The second method uses a multi-class criterion directly in the GLDB feature extractor.

Both proposed methods, followed by a conventional multi-class classifier, reached in our experiments comparable or better performances than the BPC classifier. Our results suggest that although the data representation may be derived very cheaply, the subsequent classification remains the computational bottleneck. If a small number of features is used in the multi-class classifier, the proposed methods offer up to and order of magnitude faster solutions in execution than the original BPC method. This is an especially important improvement for hyperspectral imaging applications where the raw execution speed is of importance (e.g. large-scale segmentation of hyperspectral images). It is possible that for some complex datasets with many highly overlapping classes, the BPC classifier might perform better than the proposed techniques. We believe, however, that our multi-class extensions suffice in most practical situations.

Acknowledgement

The authors would like to express their gratitude to CTR Carinthian Tech Research AG for the experimental dataset and to Thomas Landgrebe and David Tax for fruitful comments on the manuscript. This research is/was supported by the Technology Foundation STW, applied science division of NWO and the technology programme of the Ministry of Economic Affairs.

References

[1] R. P. W. Duin. *PRTools Version 3.0, A Matlab Toolbox for Pattern Recognition*. Pattern Recognition Group, TUDelft, January 2000.

[2] T. Hastie and R. Tibshirani. Classification by pairwise coupling. In M. J. Karens, M. I. Jordan, and S. A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10, pages 507–513. Cambridge,MA:MIT Press, 1998.

[3] S. Kumar, J. Gosh, and M. M. Crawford. A versatile framework for labeling imagery with large number of classes. In *Proc.Int.Joint Conf. Neural Networks*, Washington, DC, 1999.

[4] S. Kumar, J. Gosh, and M. M. Crawford. Best-bases feature extraction algorithm for classification of hyperspectral data. *IEEE Trans.on Geoscience and Remote Sensing*, 39(7):1368–1379, 2001.

[5] S. Kumar, J. Gosh, and M. M. Crawford. Hierarchical fusion of multiple classifiers for hyperspectral data analysis. *Pattern Analysis & Applications*, 5:210–220, 2002.

[6] D. Landgrebe. *Signal Theory methods in Multispectral Remote Sensing*. John Wiley & Sons, 2003.

[7] M. Loog and R. P. W. Duin. Multiclass linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(7):762–766, July 2001.

[8] P. Paclík, S. Verzakov, and R. P. W. Duin. Hypertools: the toolbox for spectral image analysis. Technical report, Pattern Recognition Group, TU Delft, The Netherlands, Dec. 2003.