Polynomial Approximation of Spectral Data in LVQ and Relevance Learning

Melchert, Friedrich; Seiffert, Udo; Biehl, Michael

Published in:
Workshop on New Challenges in Neural Computation 2015

Polynomial Approximation of Spectral Data in LVQ and Relevance Learning

Friedrich Melchert[1,2], Udo Seiffert[2], and Michael Biehl[1]

[1] University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science, P.O. Box 407, 9700 AK Groningen, The Netherlands
[2] Fraunhofer Institute for Factory Operation and Automation IFF, Sandtorstrasse 22, 39106 Magdeburg, Germany

Abstract. High dimensional data serves as input for a variety of classification tasks. In the case of spectral information, this data can be understood as discrete sampling of an (unknown) underlying function. In this paper we discuss an approach that improves classification performance for spectral data by expanding the data in terms of basis functions. Two real world spectral data classification problems demonstrate the advantages of the method.

Keywords: Classification; supervised learning; functional data; Learning Vector Quantization; relevance learning; dimensionality reduction

1 Introduction

A variety of real-world applications produce high-dimensional data that are usually difficult to handle with traditional methods. Apart from developing new methods suited for a high number of input dimensions, one possible solution is to use prior knowledge about the underlying structures of the input data for dimension reduction or data simplification. A very general and in most cases justifiable approach is to assume high dimensional data vectors are a discretized representation of a continuous function. This is true for different types of data, such as time series and spectral data, which are frequently high-dimensional.

Such data is usually recorded in order to serve as input data in a classification task. Different machine learning and classification algorithms can be applied, each having its own advantages and disadvantages. Prototype- and distance-based methods have the advantage of being intuitive to implement and interpret. In this paper an extension of the popular Learning Vector Quantization (LVQ) [6] is employed. In LVQ systems, prototypes serve as characteristic exemplars of their corresponding classes. In combination with an appropriate distance measure, they constitute an efficient method of classification [2].

The choice of the distance measure is the key to the design of an LVQ system [2]. In contrast to fixed Euclidean or other Minkowski measures, the Generalized Matrix LVQ (GMLVQ) makes use of the more flexible concept of relevance learning. In GMLVQ a parametrized distance measure is used and its parameters are determined in a data-driven training process [12]. Therefore, only the

basic structure of the distance measure has to be specified in advance. This approach offers greater flexibility and, moreover, the interpretation of the adaptive distance measure can give further insight into the structure of the data [14].

In this paper, we discuss the application of GMLVQ to functional input data. In [5] a functional representation of the relevances in GMLVQ was proposed and investigated, but samples and prototypes were still considered to be in original feature space. This paper presents an alternative approach that produces a functional representation of the data and performs GMLVQ adaptation in the space of coefficients for the functional representation. The idea was first presented in [13] where a wavelet expansion of mass spectrometry data was employed. Although this also led to a reduction of input dimensions, the wavelet expansion was motivated as an easy way of finding discriminative features of the sharply peaked mass spectra.

In [11] the SOM algorithm is modified for the unsupervised clustering of spectral information by expanding the input data in terms of B-Spline functions. This yields a reduction of input dimensions by a factor of two. In this paper we demonstrate that a similar approach using Chebyshev polynomials as functional basis can reduce the number of input dimensions even further without significant loss or even improvement of performance with respect to the full feature dataset. Since the potential benefit of the approach for the quality of the classification has been discussed in [8], here we focus on the aspect of dimensionality reduction.

2 Polynomial approximation of functional data

In order to reduce the number of input dimensions the approximation of the data using a weighted set of basis functions is considered. It is assumed, that the d-dimensional feature vectors $\boldsymbol{v}_i \in \mathbb{R}^d$ result from sampling a continuous (in general unknown) function $f_i(x)$.

$$\boldsymbol{v}_{i,j} = f_i(x_j) \quad \text{with} \quad j = 1, 2, ..., d. \tag{1}$$

For a given set of suitable basis functions $g_k(x)$ it is possible to find the coefficients $c_{i,j}$ so that

$$f_i(x) = \sum_{k=0}^{\infty} c_{i,j} \cdot g_k(x). \tag{2}$$

By limiting the maximum number of coefficients to n, and thus truncating the series, Eq. (2) becomes an approximation of the original function, which can be achieved by means of general approximation schemes like least mean squares or minimal maximum deviation.

From the obtained set of coefficients $c_{i,j}$ a new set of feature vectors $\boldsymbol{c}_i \in \mathbb{R}^n$ can be composed. Throughout the following we assume that $n < d$, obviously.

The quality of the functional approximation is highly dependent on the selection of an appropriate set of basis functions and, of course, on the number of approximation coefficients. Both choices should be guided by the specific properties of the input data. With respect to the number of functions and coefficients

the proposed approach is quite robust within a range of values n [8]. Possible basis functions include polynomials, trigonometric functions for periodic signals, wavelets or spline functions. In this paper, we focus on using a polynomial basis as an important example, more specifically the set of Chebyshev polynomials of the first kind. They are defined recursively by

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_n(x) &= 2xT_{n-1} - T_{n-2}(x) \end{aligned} \tag{3}$$

Using Chebyshev polynomials as basis functions Equation (2) becomes a Chebyshev series which is known to provide an efficient way to represent smooth non-periodic functions [3]. Within the context of this paper an open source MATLAB^{TM} library called *chebfun* [15] is employed to determine the coefficients of the series. Although the library provides a wide variety of functions in the context of Chebyshev polynomials, we only use the implementation of approximations for discrete data in this paper. For a more detailed description of the implementation see the documentation in [3].

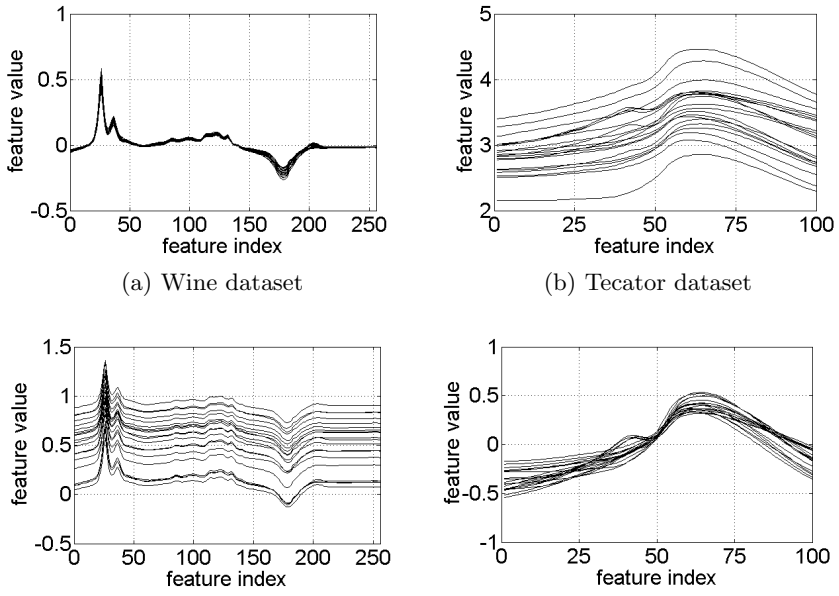
3 Application to example datasets

The approach is demonstrated by applying it to two publicly available datasets as well as manually distorted copies thereof. All of the datasets contain spectral and thus functional input data. An illustration of the datasets is shown in Figure 1.

The first dataset, the *Wine* dataset (available from [9]), contains 124 samples of wine infrared absorption spectra in the range between 4000 and 400 cm^{-1} with 256 sampled values each. One sample of the dataset, which could be clearly identified as an outlier, was removed for the following analysis. The samples are labeled according their alcohol content: A two class problem is created by thresholding the alcohol level as described in [7], the resulting classes correspond to low and high alcohol content.

As a second dataset we consider the *Tecator* dataset (available from [16]), which comprises 215 reflectance spectra in the range from 850 to 1050 nm wavelength. The spectra are sampled equidistantly using 2 nm step size resulting in 100 sampled values per spectrum. The spectral information was acquired from meat probes and labeled according to their fat content. Similar to the *Wine* dataset the fat content is thresholded at its median in order to obtain two classes.

For further illustration of the presented approach both datasets are artificially distorted. For the *Wine* dataset, which seems to be pre-processed in terms of offset elimination, cf. Fig. 1a, a random offset is added to each of the spectra. This yields a dataset which will be referred to as *WineRO* (Wine with Random Offset) in the following. The spectra in the *Tecator* dataset already have different offsets (cf. Fig. 1b), so for distortion the offsets in the dataset are removed by subtracting the mean value of each spectrum. The resulting dataset will be referred to as the *TecatorNO* (Tecator with No Offset) dataset.



(a) Wine dataset (b) Tecator dataset

(c) WineRO dataset (d) TecatorNO dataset

Fig. 1: Input spectra of the different datasets. For the sake of clarity only 20 examples are drawn for each dataset. The presence of offsets in the *WineRO* and *Tecator* datasets is clearly recognizable.

Two experiments are performed for each of the datasets: The first serves as a natural baseline for classification performance and disregards the functional characteristic of the input values entirely by training a GMLVQ system in the original feature space.

The second set of experiments involves a preprocessing of the data in terms of polynomial expansion as described in section 2. For each dataset an approximation is computed with $n = 5, 10, 15, ..., 50$ polynomial coefficients, resulting in new input feature vectors $\boldsymbol{c_i} \in \mathbb{R}^n$.

Demonstration code (MATLAB$^{\text{TM}}$) for GMLVQ training is available from [1]. The settings and parameters kept constant for all experiments. For most parameters the default values as specified in [1] were used. In detail this means, all trained GMLVQ systems comprised only one prototype per class which were initialized as the class-conditional means in the training set. The relevance matrix was initialized as proportional to the identity and batch gradient descent optimization was performed employing an automated step size control as described in [1, 10]. As an additional preprocessing step the input data underwent a z-score transformation that achieves unit variance and zero mean for all input features. This transformation was done in order to balance varying orders

Table 1: Comparison of processing workflows for experiments with and without incorporation of the functional characteristics of the input data.

	original data	functional approximation
Preprocessing:		polynomial approximation with n coefficients $\boldsymbol{v}_i \rightarrow \boldsymbol{c}_i^n$
	z-score transformation $\boldsymbol{v}_i \rightarrow \boldsymbol{v}_i^Z$	z-score transformation $\boldsymbol{c}_i \rightarrow \boldsymbol{c}_i^{n,Z}$
Training:	Train GMLVQ on 90% of $\boldsymbol{v}_i^Z$	Train GMLVQ on 90% of $\boldsymbol{c}_i^{n,Z}$
Validation:	Validate GMLVQ on remaining 10% of $\boldsymbol{v}_i^Z$	Validate GMLVQ on remaining 10% of $\boldsymbol{c}_i^{n,Z}$

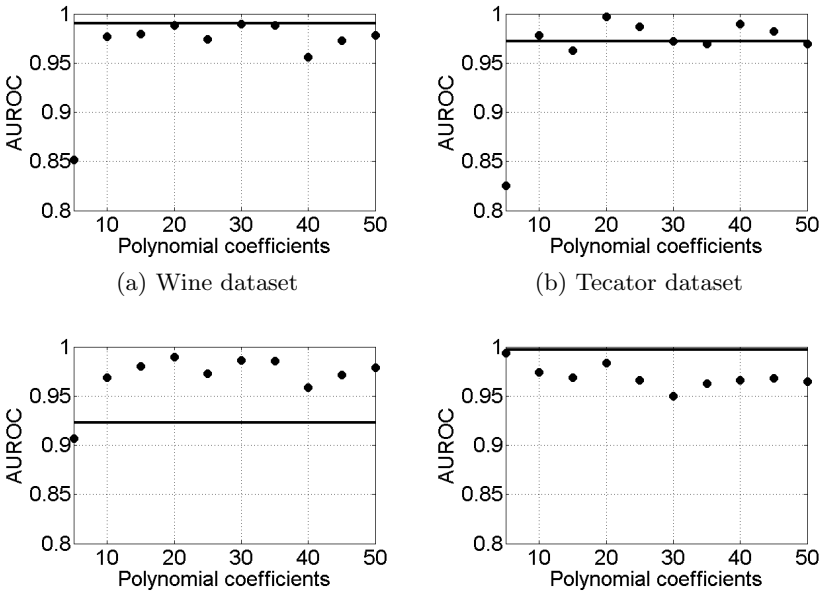
of magnitudes between the different features. Furthermore the transformation facilitates a better interpretation of the resulting relevance matrices [12].

A validation scheme dividing the datasets randomly into 90% training data and using the remaining 10% as a validation dataset is employed for each of the experiments. As a measure for classification performance the area under the ROC (AUROC) is evaluated with respect to the validation set [4]. The ROC is computed by varying a threshold when comparing the distances between the data points and the prototypes. The whole workflow is summarized in Table 1. All results were obtained by averaging over 10 random splits of the data. Figure 2 shows the obtained performance for all datasets in dependency of the number of polynomial approximation coefficients, as well as the performance of the classifier using the raw input data.

To illustrate the advantages of the approach we provide more detail on the *Wine* and *WineRO* datasets in Figure 3. The left-hand panels show the prototypes obtained using a 20 coefficient polynomial approximation. The prototypes are shown in the space of approximation coefficients, center panels display the corresponding relevance profiles. In the right-hand panels, the reconstructed prototypes are shown in the original feature space.

4 Discussion

The results depicted in Figure 2 reveal that the classification performance of the GMLVQ systems trained on the polynomial approximation coefficients are almost identical to or slightly better than the performance when using original data. However, in a polynomial approximation with, say, 20 coefficients, which performs well on all datasets (cf. Fig. 2), the number of input dimensions is drastically decreased by 80% for the *Tecator* datasets and by 92% for the *Wine* datasets.



(a) Wine dataset (b) Tecator dataset

(c) WineRO dataset (d) TecatorNO dataset

Fig. 2: Comparison of the achieved validation performance, i.e. the area under ROC for different datasets in dependence of the number of polynomial approximation coefficients. The solid line represents the value for the classification using the unprocessed spectral information as feature vectors. Filled circles represent results achieved using polynomial approximation coefficients.

Comparing the performance on datasets with and without artificial distortion, we conclude that the distortion has no significant effect on performance when the polynomial approximation is employed. However, the classification performance with original spectra as input data, is significantly better for the two datasets without offset (*Wine* and *TecatorNO*) than for their counterparts retaining the offsets.

The combination of polynomial approximation and relevance learning is able to suppress the influence of offsets. As shown in figure 3b and 3e the first polynomial coefficient, which represents $T_0(x) = 1$ and can therefore be understood as the constant part of the spectrum, is virtually disregarded by GMLVQ as indicated by a very low value of the corresponding diagonal element of the relevance matrix. Thus, the classification performance for both dataset versions, with and without offset, is nearly the same.

Another benefit of the polynomial approximation is an implicit denoising and smoothing of the data, as can be seen in Figure 3c and 3f. Apart from the (irrelevant) offset, the prototypes are almost identical. The significant smoothing

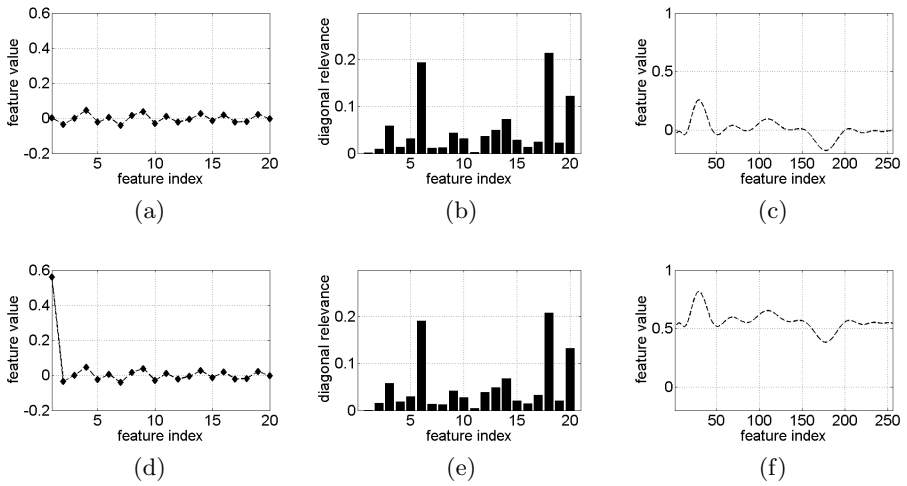


Fig. 3: Detailed comparison of one prototype for the *Wine* and *WineRO* dataset. The top panels (a,b,c) belong to the *Wine* dataset, the bottom (d,e,f) to the *WineRO* dataset. Left-hand panels (a,d) represent prototypes in space of polynomial coefficients, center panels (b,e) represent the relevance profiles obtained and the right-hand panels (c,f) represent the prototypes after retransformation to original feature space.

caused by the polynomial approximation becomes evident when comparing the prototypes to the input spectra in Figure 1.

5 Summary and Outlook

We presented a framework for reducing input dimensions for classification of functional data, by applying polynomial approximation and performing classification in the space of the approximation coefficients. We considered two real world spectral datasets, which were artificially distorted in order to illustrate the advantages of the presented approach. The results show that for a suitable number of polynomial coefficients the resulting classification performance is comparable or exceeds that for unprocessed data. In comparison to [11] using Chebyshev polynomials as basis functions, the number of input dimensions was more significantly decreased by up to 92%, thus drastically reducing the number of parameters, the risk of over-fitting, convergence problems and computational effort.

Furthermore, the robustness of the approach to offset distortion of the data was demonstrated for both example datasets. In forthcoming studies we will address the question whether this independence also holds for more complex distortions, such as different scaling of data or the superposition of trends or

other more complex offsets. Moreover, the intermediate functional representation of the data allows for a more convenient application of mathematical operations such as derivatives, integration or root finding to preprocess the data. These can be used to provide more complex descriptions of the underlying functions, e.g. the number of roots/maxima or maximum slope, to generate even lower-dimensional feature vectors, that can serve as classification input.

Acknowledgments. F. Melchert thanks for the support of an Ubbo-Emmius Sandwich Scholarship from the Faculty of Mathematics and Natural Sciences, University of Groningen.

References

1. Biehl, M.: A no-nonsense beginner's tool for GMLVQ. Available online, University of Groningen, http://www.cs.rug.nl/~biehl
2. Biehl, M., Hammer, B., Villmann, T.: Distance measures for prototype based classification. In: Grandinetti, L., Petkov, N., Lippert, T. (eds.) BrainComp 2013, Proc. International Workshop on Brain-Inspired Computing, Cetraro/Italy, 2013. Lecture Notes in Computer Science, vol. 8603, pp. 100–116. Springer (2014)
3. Driscoll, T.A., Hale, N., Trefethen, L.N.: Chebfun guide. Pafnuty Publ. (2014)
4. Duda, R., Hart, P., Stork, D.: Pattern Classification. Wiley (2000)
5. Kästner, M., Hammer, B., Biehl, M., Villmann, T.: Generalized functional relevance learning vector quantization. In: Verleysen, M. (ed.) Proc. Europ. Symp. on Artificial Neural Networks (ESANN). pp. 93–98. d-side (2011)
6. Kohonen, T.: Self-organizing maps. Springer, Berlin (1995)
7. Krier, C., François, D., Rossi, F., Verleysen, M., et al.: Supervised variable clustering for classification of nir spectra. In: Verleysen, M. (ed.) Proc. Europ. Symp. on Artificial Neural Networks (ESANN). pp. 263–268. d-side (2009)
8. Melchert, F., Seiffert, U., Biehl, M.: Functional representation of prototypes in LVQ and relevance learning. Preprint, submitted for publication (2016)
9. Meurens, P.M.: Wine mean infrared spectra dataset. University of Louvain, http://mlg.info.ucl.ac.be/index.php?page=DataBases
10. Papari, G., Bunte, K., Biehl, M.: Waypoint averaging and step size control in learning by gradient descent. Machine Learning Reports MLR-06/2011, 16 (2011)
11. Rossi, F., Conan-Guez, B., El Golli, A.: Clustering functional data with the SOM algorithm. In: ESANN. pp. 305–312 (2004)
12. Schneider, P., Biehl, M., Hammer, B.: Adaptive relevance matrices in Learning Vector Quantization. Neural Computation 21, 3532–3561 (2009)
13. Schneider, P., Biehl, M., Schleif, F.M., Hammer, B.: Advanced metric adaptation in Generalized LVQ for classification of mass spectrometry data. In: Proc. 6th Intl. Workshop on Self-Organizing-Maps (WSOM). Bielefeld University (2007), 5 pages
14. Strickert, M., Hammer, B., Villmann, T., Biehl, M.: Regularization and improved interpretation of linear data mappings and adaptive distance measures. In: Computational Intelligence and Data Mining (CIDM), 2013 IEEE Symposium on. pp. 10–17 (April 2013)
15. The Chebfun Developers: Chebfun - Numerical Computing with functions. Available online, University of Oxford, http://www.chebfun.org
16. Thodberg, H.H.: Tecator meat sample dataset. StatLib Datasets Archive, http://lib.stat.cmu.edu/datasets/tecator