

Local Subspace Classifier

Jorma Laaksonen

Helsinki University of Technology
 Laboratory of Computer and Information Science
 P.O. Box 2200, FIN-02015 HUT, Finland

Abstract. This paper describes a new classification technique named the *Local Subspace Classifier* (LSC). The algorithm is closely related to the subspace classification methods. On the other hand, it is an heir of prototype classification methods, such as the k -NN rule. Therefore, it is argued that the LSC technique fills the gap between the subspace and prototype principles of classification. From the domain of the prototype-based classifiers, LSC brings the benefits related to the local nature of the classification, while it simultaneously utilizes the capability of the subspace classifiers to produce generalizations from the training sample. A further enhancement of the LSC principle named the *Convex Local Subspace Classifier* (LSC+) is also presented. The good classification accuracy obtainable with the LSC and LSC+ classifiers is demonstrated with experiments, including the classification of the publicly available data sets of the StatLog project.

1 Introduction

A new statistical classification technique, named the Local Subspace Classifier (LSC) and its further enhancement, the Convex Local Subspace Classifier (LSC+) are described in this paper. The technique is closely related to the subspace classification methods [1]. On the other hand, it is an heir of prototype classification methods, such as the k -NN rule. The purpose the LSC method is to combine the beneficial characteristics of these two principles.

In the LSC processes, the nearest prototypes to the input vector in all the classes are sought. A local subspace – or more precisely, a *linear manifold* – is then spanned by these prototype vectors in each class. The classification is based on the minimum distance from the input vector to these subspaces.

2 Local Subspace Classifier Algorithm

A D -dimensional linear manifold $\mathcal{L}$ of the d -dimensional real space is defined by a matrix $\mathbf{U} \in \mathbb{R}^{d \times D}$ of rank D , and an offset vector $\boldsymbol{\mu} \in \mathbb{R}^d$, provided $D \leq d$,

$$\mathcal{L}_{\mathbf{U}, \boldsymbol{\mu}} = \{\mathbf{x} \mid \mathbf{x} = \mathbf{U}\mathbf{z} + \boldsymbol{\mu} ; \mathbf{z} \in \mathbb{R}^D\} . \quad (1)$$

The same manifold can alternatively be defined by $D + 1$ prototypes provided that the set of prototypes is not degenerate. The prototypes forming the classifier

are marked $\mathbf{m}_{ij}$ where $j = 1, \dots, c$ is the index of the class and $i = 1, \dots, N_j$ indexes the prototypes in that class. The manifold dimension D may be different for each class. Therefore, the class-dependent dimensions are denoted D_j . When classifying a vector $\mathbf{x}$, the following is done for each class $j = 1, \dots, c$:

1. Find the $D_j + 1$ prototypes closest to $\mathbf{x}$ and denote them $\mathbf{m}_{0j}, \dots, \mathbf{m}_{D_j j}$.
2. Form a $d \times D_j$ -dimensional basis of the vectors $\{\mathbf{m}_{1j} - \mathbf{m}_{0j}, \dots, \mathbf{m}_{D_j j} - \mathbf{m}_{0j}\}$.
3. Orthonormalize the basis to get the matrix $\mathbf{U}_j = (\mathbf{u}_{1j} \dots \mathbf{u}_{D_j j})$.
4. Find the projection of $\mathbf{x} - \mathbf{m}_{0j}$ on the manifold $\mathcal{L}_{\mathbf{U}_j, \mathbf{m}_{0j}}$:

$$\hat{\mathbf{x}}'_j = \mathbf{U}_j \mathbf{U}_j^T (\mathbf{x} - \mathbf{m}_{0j}) . \quad (2)$$

5. Calculate the residual of $\mathbf{x}$ relative to the manifold $\mathcal{L}_{\mathbf{U}_j, \mathbf{m}_{0j}}$:

$$\tilde{\mathbf{x}}_j = \mathbf{x} - \hat{\mathbf{x}}_j = \mathbf{x} - (\mathbf{m}_{0j} + \hat{\mathbf{x}}'_j) = (\mathbf{I} - \mathbf{U}_j \mathbf{U}_j^T) (\mathbf{x} - \mathbf{m}_{0j}) . \quad (3)$$

The vector $\mathbf{x}$ is then classified according to minimal $\|\tilde{\mathbf{x}}_j\|$ to the class j , i.e.,

$$g_{\text{LSC}}(\mathbf{x}) = \underset{j=1, \dots, c}{\operatorname{argmin}} \|\tilde{\mathbf{x}}_j\| . \quad (4)$$

In any case, the residual length from the input vector $\mathbf{x}$ to the linear manifold is equal to or smaller than the distance to the nearest prototype, i.e., $\|\tilde{\mathbf{x}}_j\| \leq \|\mathbf{x} - \mathbf{m}_{0j}\|$. These entities are sketched in Fig. 1.

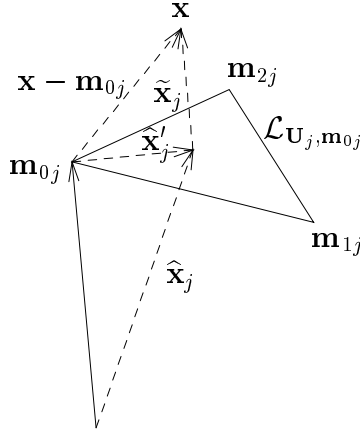


Fig. 1. The d -dimensional entities involved in the LSC classification when $D_j = 2$.

By introducing the multipliers $\{c_{0j}, \dots, c_{D_j j}\}$ forming the coefficient vector $\mathbf{c}_j = (c_{0j} \dots c_{D_j j})^T$, and using the matrix $\mathbf{M}_j = (\mathbf{m}_{0j} \dots \mathbf{m}_{D_j j})$, the projection vector $\hat{\mathbf{x}}_j$ can be expressed

$$\hat{\mathbf{x}}_j = \mathbf{m}_{0j} + \hat{\mathbf{x}}'_j = \sum_{i=0}^{D_j} c_{ij} \mathbf{m}_{ij} = \mathbf{M}_j \mathbf{c}_j , \quad \sum_{i=0}^{D_j} c_{ij} = 1 . \quad (5)$$

If the explicit values for the coefficients c_{ij} are needed, they can be solved with matrix pseudo-inversion

$$\mathbf{c}_j = \mathbf{M}_j^\dagger \hat{\mathbf{x}}_j = (\mathbf{M}_j^T \mathbf{M}_j)^{-1} \mathbf{M}_j^T \hat{\mathbf{x}}_j . \quad (6)$$

The principle of local subspaces can be illustrated by Fig. 2, which displays a two-dimensional linear manifold defined by the three handwritten ‘4’s in the corners of the triangle. Each of the three corners represents a pure instant of a normalized 32×32 -sized human printed glyph whereas the images in between are linear combinations thereof. It may be argued that such interpolated images are as good representatives of the class as the original prototypes. Therefore, if the input vector resembles any of such artificial images, it can be considered to originate from that class.

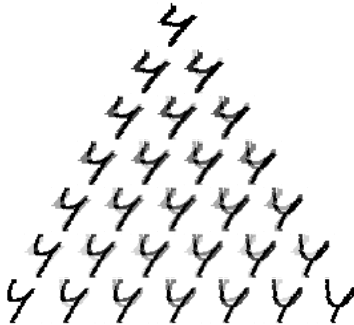


Fig. 2. A piece of a two-dimensional linear manifold spanned by three examples of handwritten digits ‘4’ in the corners. The gray areas in the intermediate virtual digit images reflect the gradual pixel-wise change from one real image to another.

3 LSC+ Classifier

The center image in Fig. 3 shows that the input vector $\mathbf{x}$ may occasionally be projected outside the convex region determined by the prototypes $\mathbf{m}_{0j}, \dots, \mathbf{m}_{D_jj}$. Whether this is beneficial or disadvantageous depends on the particular data sets involved. If no such extrapolation from the original prototypes is permitted, the projection may be enforced to lie within the area delimited by the prototype vectors. As a result, the orthogonality of the projection is lost. The modified method is here called the *Convex Local Subspace Classifier* (LSC+). The appended plus sign arises from the observation that the coefficients $\{c_{0j}, \dots, c_{D_jj}\}$ in (5) are all non-negative in the case the projection vector $\hat{\mathbf{x}}$ is not allowed to exceed the convex region determined by the prototype vectors.

In LSC+, the projection vector $\hat{\mathbf{x}}$ is thus not the one producing the shortest residual $\tilde{\mathbf{x}}$. Instead, it primarily lies within the convex region bounded by the

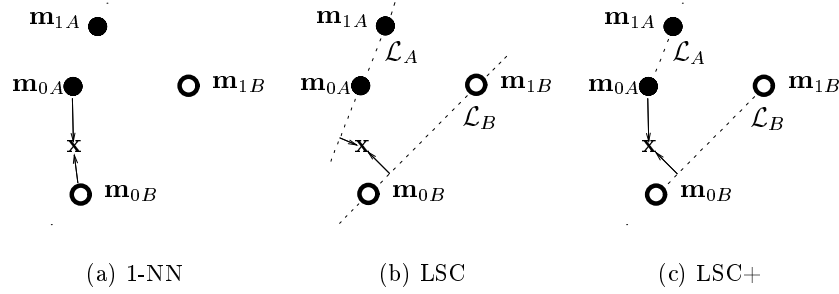


Fig. 3. Comparison of the 1-NN rule and the basic and convex versions of the Local Subspace Classifier. Using the 1-NN and LSC+ classifiers, $\mathbf{x}$ is classified as “B”, but as “A” when using the LSC method.

nearest prototypes and only secondarily minimizes the residual length. Obviously, if the orthogonal and convex projections differ, the convex projection lies on the boundary of the convex area and one or more of the coefficients are equal to zero. In the convex case, the residual length $\|\tilde{\mathbf{x}}\|$ is never shorter than in the orthogonal case. The two rightmost images in Fig. 3 show how a vector $\mathbf{x}$ is classified differently between the k -NN, LSC, and LSC+ classifiers.

The convexity condition can be achieved iteratively by removing from the basis $\mathbf{M}_j$ the vectors which have negative coefficients. This is effectively the same as forcing the corresponding coefficients to be equal to zero. Then, the coefficients for the remaining vectors are solved again and checked for negativeness. The dimension D_j of the linear manifold is decreased by one for each vector removed. In the experiments performed, the vector with the most negative coefficient has been removed in each iteration step.

4 Experiments

The first experiment data was the same as that used in [2]. 17880 binary handwritten digits were normalized to the size of 32×32 pixels. One half of the images was used in training set cross-validation and the other half in testing. Karhunen-Loève transformation was calculated from the training set and applied to the 1024-dimensional images. As the result, feature vectors with the maximum dimensionality of 64 were obtained. 10-fold training set cross-validation was used to solve the optimal values for the feature vector dimensionality, d , and the dimensionality of the linear manifolds, D , common to all the classes. The testing set error rates are shown in Table 1 together with three results from [2]. The 3-NN and ALSM classifiers represent the prototype and subspace principles of classification, respectively, upon which the Local Subspace Classifier technique is based. The *Local Linear Regression* (LLR) classifier attained the best classification accuracy in the comparison [2] in which 17 different types of classifiers were

considered. It can be seen that the LSC and LSC+ classifiers clearly outperform all the reference techniques in classification accuracy.

Table 1. Test set classification error percentages of the LSC and LSC+ classifiers together with the Nearest Neighbor rule (3-NN), Averaged Learning Subspace Method (ALSM), and the Local Linear Regression (LLR) classifier, for the data of [2].

<i>classifier</i>	<i>error %</i>	<i>parameters</i>
LSC	2.5	$d = 64, D = 12$
LSC+	2.1	$d = 64, D = 23$
3-NN	3.8	$d = 38$
ALSM	3.1	$d = 64, D = 29$
LLR	2.8	$d = 36$

The performance of the LSC and LSC+ classifiers was also tested by using the publicly available StatLog [3] data sets. The “australian” test was omitted because the data contained categorical variables. Likewise, the “german” and “heart” data sets were not used because those comparisons involved weighted cost matrices. All the other sets were tested with procedures similar to those described in the original report. It can be seen from the results shown in Table 2 that the performance of the LSC/LSC+ classifier is superior to all classifiers tested in the StatLog project in three cases out of seven, i.e., “letter”, “satimage”, and “segment”. In the “shuttle” and “vehicle” cases the performance is moderate, whereas the remaining two results, “diabetes” and “dna”, are clearly failures.

This series of results indicates that the LSC and LSC+ techniques are efficient in some situations but suffer seriously in some other cases. The performance of the local subspace methods is dependent on the nature and density of the data in the Bayesian class border areas. The smoother the hypothetical class boundaries are and the more there exist data to describe them, the better prospects the LSC/LSC+ techniques have.

Table 2. The performances of the LSC and LSC+ classifiers compared to the best StatLog results. Shown are, from left to right, the test set classification error percentage, the subspace dimension D common to all classes, and the rank of the result.

<i>test</i>	<i>StatLog result</i>			<i>LSC</i>			<i>LSC+</i>		
diabetes	LogDisc	22.3		31.25	1	22nd	31.25	1	22nd
dna	Radial	4.1		16.19	8	21st	16.44	8	21st
letter	Alloc80	6.4		3.22	3	1st	3.04	6	1st
satimage	k -NN	9.4		7.55	3	1st	7.45	3	1st
segment	Alloc80	3.0		3.68	2	5th	2.99	3	1st
shuttle	NewId	0.01		0.12	0	8th	0.11	1	8th
vehicle	QuaDisc	15.0		27.07	4	10th	24.35	13	9th

5 Discussion

In this paper, a new family of classification techniques has been introduced. It has been argued that the Local Subspace Classifier (LSC) and the Convex Local Subspace Classifier (LSC+) techniques combine the benefits of the subspace and prototype based classification methods. An experimental evaluations of the two proposed classification algorithms supported this argument.

In most cases, the optimal value of the manifold dimension parameter D is larger in the LSC+ than in the LSC, which does not demand the convexity of the subspace projection. At the same time, the classification accuracy is usually better in the former. These two facts can be interpreted together as follows: The increased value of D provides the classifier with more prototypes for each classification task. Due to the convexity requirement, only the most informative subset of these prototypes are actually used in the difficult regions of classification where the class distributions are sparse and overlapping. Compared to prototype based classifiers, such as 3-NN and LLR, both the LSC and LSC+ techniques seem to benefit from larger feature vector dimensionality d . In this sense, the LSC techniques thus resemble the subspace methods of classification, which, in general, are able to take advantage of large vector dimensionalities.

The application of the LSC principle allows for some additional options. The explicit use of the orthonormalization process and the matrix pseudo-inversion (6) can be avoided by using an iterative algorithm for the solution of $\hat{\mathbf{x}}_j$. Furthermore, the LSC technique can be given a neural interpretation. These two topics are addressed in [4]. The LSC technique may also be gracefully combined with adaptive and self-organizing prototype classifiers such as the *Learning k-Nearest Neighbor classifier* (L-k-NN) [5], the *Learning Vector Quantization* (LVQ) [6], and especially with the topology preserving vector quantization produced by the *Self-Organizing Map* (SOM) [6]. The combination of the LSC and SOM principles is named the *Local Subspace SOM* (LSSOM) [7]. Efficient methods for editing the design set in order to reduce the memory and computation requirements are also being developed.

References

1. Oja, E. *Subspace Methods of Pattern Recognition*. Research Studies Press, Letchworth, England (1983)
2. Holmström, L., Koistinen, P., Laaksonen, J., Oja, E. Neural and statistical classifiers – taxonomy and two case studies. *IEEE Tr. on Neural Networks* 8:5–17 (1997)
3. Michie, D., Spiegelhalter, D. J., Taylor, C. C., editors. *Machine learning, neural and statistical classification*. Ellis Horwood, New York (1994)
4. Laaksonen, J. *Subspace Methods in Recognition of Handwritten Digits*. Ph.D. thesis, Helsinki University of Technology (1997)
5. Laaksonen, J., Oja, E. Classification with learning k -nearest neighbors. In *Proc. of the Intl. Conf. on Neural Networks*, vol. 3, pp. 1480–1483, Washington D.C. (1996)
6. Kohonen, T. *Self-Organizing Maps*. Springer, Berlin, Germany (1995)
7. Laaksonen, J. Local subspace classifier and local subspace SOM. In *Proc. of the Workshop on Self-Organizing Maps*, Otaniemi, Finland (1997). To appear.