Sparse Representation for Classification of Faces

EE-596 Class Project

Naveen Kumar

Abstract

Sparsity of a signal in a basis or dictionary implies that it can be represented using only a few coefficients. Hence follows the intuition for use of sparsity in compression or denoising applications. However, a sparse representation is also intrinsically discrminative in nature. A sparse signal for a given dictionary has a unique representation in it, using few of the most similar basis vectors. This fact can be exploited for classification of facial images.

1 Introduction

Sparse Representation of a signal in an overcomplete dictionary, tries to express the signal in terms of the most similar atoms, which ensures that most of the signal can be reconstructed back using a very few coefficients. However this ability of sparse representations to select the most similar atoms from a dictionary is also the basis for its discriminative power [3].

This classification idea may in theory seem to be similar to classifiers like nearest neighbour, where one finds the nearest points to a given test sample and ascertains its class based on the identity of those points. In practice though sparsity of the solution ensures that there is maximal discrimination, making it easier to classify. This is in part because nearest neighbour usually minimizes an L_2 norm, whereas a sparse representation is typically found by minimizing an L_1 norm. Like KNN or other similar non parametric methods this method doesn't learn any models on the training data. The model then comprises the training data points themselves. However as we will see later, the role of feature extraction becomes quite insignifincant making it sufficient to store just compact representations of each data point.

Before moving ahead, we must consider how this method differs from traditional multiclass classifiers. The typical approach for traditional multiclass classifiers is to either construct several pairwise binary classifiers or sometimes use a joint multiclass formulation. In either case the classification accuracy declines rapidly as the number of classes increase. This problem is inherent to most conventional classifiers. For the *sparsity classifier* although, its a natural advantage. As the number of classes increases the proportion of samples from each class in the dictionary keeps decreasing. This makes the dictionary more and more incoherent or overcomplete which in turn makes it easier to obtain a sparse representation. A typical application is face recognition which has one class per face.

1.1 Sparse Representation

Given an overcomplete dictionary we now turn to the problem of finding the *sparsest* representation for a signal in that dictionary. More formally, given a signal y and a dictionary D where each column is an atom or basis function, we wish to find a *sparse* $\bar{\alpha}$ such that $y = D\bar{\alpha}$.

The sparsest solution involves minimizing the L_0 norm which essentially counts the number of non-zero entries in $\bar{\alpha}$. This problem is combinatorial in nature and infeasible. A naive

alternative is to minimize the L_2 norm, which has a closed form solution via the pseudoinverse. However a little observation of the norm locus diagrams clearly reveals that an L_2 norm minimum solution is not typically sparse. The L_1 norm on the other hand is a good tradeoff between complexity and sparsity. Hence we try to minimize the L_1 norm as follows

$$\bar{\alpha}^* = \arg\min_{\bar{\alpha}} \ ||\bar{\alpha}||_1 = \arg\min_{\bar{\alpha}} \sum_i |\alpha_i| \quad \textit{such that } y = D\bar{\alpha}. \tag{1}$$

Note that the solution to this minimization can easily be obtained via a dual formulation that allows it to be solved using a linear program [2].

1.2 Sparsity for Classification

In the context of face recognition there is infact a well grounded reason to believe that the above mentioned approach shall lead to a sparse representation on the dictionary of all training faces. Previous works suggest that it is safe to assume that **given *sufficient* training faces of a class, a test image of the same class can be represented as a linear combination of the training faces** [1]. In other words faces of a class lie on a subspace. If $v_i^{(k)}$ indicate the traning samples for the k^{th} class, then a test sample $y^{(k)}$ of the same class can be written as:

$$y^{(k)} = \sum_i \alpha_i^{(k)} v_i^{(k)} \tag{2}$$

This means that in theory if we represent a face using a dictionary comprising all training faces, $\bar{\alpha}$ will contain all zeros except for the positions which contain faces of that class in the dictionary. However in practice due to some additional noise associated with the test face we obtain an *almost* sparse representation for the test face, where the coefficients of the other classes are small, but not necessarily zero.

2 Method

To construct the dictionary D, all training faces x_n were resized to a nominal size. Evidence has been presented in [4] that for this method feature extraction does not play any important role. Hence simply reszing the image is as good as any other sophisticated feature. Now given a test face y, we find $\bar{\alpha}$, such that $y = D\bar{\alpha}$. The coefficients in $\bar{\alpha}$ are found via minimization of its L_1 norm using a linear program.

Once $\bar{\alpha}$ has been determined we can use some heuristics on these sparse coefficients to design a classifier that can infer the most likely class for the test sample. One simple heuristic would be to find the face with the largest coefficient and reporting its class label as that of the test sample. Alternatively we can find the average of the absolute coefficient values for each class in the dictionary, and predict the class with the maximum average to be the most likely class for this sample. However these methods are still ad-hoc at best. They do not exploit the linear subspace model on which this classifier is based.

Thus next we try to use this idea to compute reconstruction errors for the signal using this sparse reprensentation. If the facial image y was indeed from the class k it would be possible to reconstruct it using only the faces corresponding to class k in the dictionary. We thus compute the supposed reconstruction error for the face assuming that it belongs to class k as follows

$$r_k(y) = || \ y - \hat{y_k} \ ||_2 = || \ y - \sum_{x_i \in C_k} \alpha_i x_i \ ||_2 \qquad \forall k = 1 \cdots K \tag{3}$$

The class for which the residual error is minimum is selected.

3 Experiment

A fundamental assumption that this method makes is that faces of a class lie on a single subspace, given sufficient training samples. Here *sufficient* needs to be taken quite seriously, since it is related to the guarantee of sparsity. Additionally to make the system more overcomplete, it makes sense to also reduce the size of each atom in the dictionary. This was verified on the AT&T Faces Dataset, which has only 10 sample face images of size 92×112 for each of the 40 people. With so few training samples in each class the dictionary obtained does not give a sparse representation for test faces. Hence we experiment on the YaleB database instead which contains on average about 64 sample images of size 192×168 per class. It contains pictures of 38 different people in all, under different lighting conditions. Sparse representations are obtained for the test faces now. To further ensure that the dictionary is overcomplete, the images are resized to 12×10.

3.1 Cross Validation

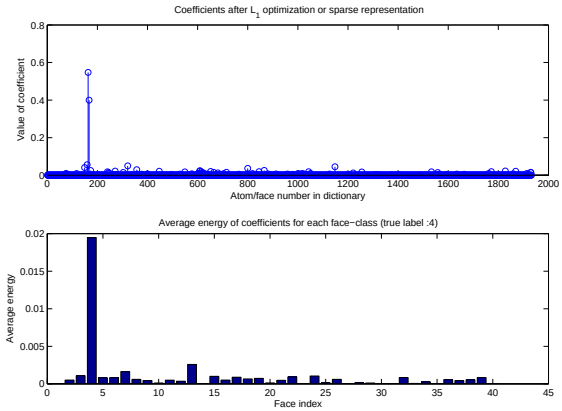
For 2414 images in the YaleB database, a five fold cross validation was performed for each of the three methods described above. Residual error on the test image outperforms the other methods as expected.

Method	Cross-validation accuracy
Max. Coeff.	0.9259
Average abs. coeff per class	0.9457
Residual Error per class	0.9462

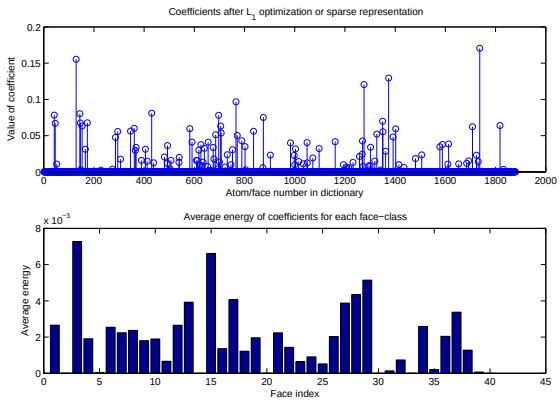
Table 1: Five fold cross validation accuracies on YaleB dataset

3.2 Out of database

Fig. 1 presents another interesting example. Consider the case that the training set doesn't contain any samples from the class that the test sample belongs to. For the case of facial images this could mean the test face is of a person who doesn't belong to the set of users it has been trained for. For a face recognition application, the classifier should reject this test sample as belonging to neither of the classes. In tradiitonal multiclass classifiers this is typically



(a) Inside training class test sample



(b) Out of training class test sample

Figure 1: Comparison of the sparse representation for inside and outside train class test samples. *Notice how the coefficient are almost equally spread out for the out of database case*

implemented using a background model or a model for "everything else". However this requires training data for "everything else".

The sparsity classifier has a smart answer to this problem. Since the test sample is not similar to any of the faces in the dictionary, a sparse representation is no longer obtained. Hence by checking for some sparsity metric of the representation, the classifier can first decide whether to accept or reject a sample.

4 Conclusion

In this project the benefits of using a sparse representation based classifier were verified, specifically in the context of facial image classification. The inherent discriminative property of sparsity is quite different from the age-old discriminative function based classifiers, which makes it suitable for multi class problems with a large number of classes. Additionally it has been suggested in [4] that feature extraction plays no role for this classifier. Hence compaction of the images using even simple image resizing gives a high performance. Sparsity breaks down when the test sample class does not have any samples in the dictionary. This fact can be exploited to reject out of database samples, without the need for an explicit background model. This fact can be easily extended to very real problems like noise rejection or impostor detection.There are some more advantages of this method like robustness to occlusion which have not been reported here.

It is important to remember however that it comes with a caveat of *sufficient* data. If there is a lack of training samples, the dictionary is not sufficiently overcomplete and sparsity is not guaranteed. The computational complexity is not trivial either since each classification needs to solve an L_1 optimization via a linear program which can be quite expensive.

References

[1] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):711–720, 2002.

[2] E. Candes and J. Romberg. l1-magic: Recovery of sparse signals via convex programming. *URL: www. acm. caltech. edu/l1magic/downloads/l1magic. pdf*, 4.

[3] K. Huang and S. Aviyente. Sparse representation for signal classification. *Advances in Neural Information Processing Systems*, 19:609, 2007.

[4] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2008.