Outlier Detection using Classifier Instability.

David M.J. Tax and Robert P.W. Duin

Pattern Recognition Group
Delft University of Technology
Lorentzweg 1, 2628 CJ Delft, The Netherlands
{davidt,bob}@ph.tn.tudelft.nl

Abstract. When a classifier is used to classify objects, it is important to know if these objects resemble the train objects the classifier is trained with. Several methods to detect novel objects exist. In this paper a new method is presented which is based on the instability of the output of simple classifiers on new objects. The performances of the outlier detection methods is shown in a handwritten digit recognition problem.

1 Introduction

A very important aspect of the use of neural networks is the ability to generalize. Good generalization means that the classifier gives reasonable responses on unseen data. This can only be achieved when the new data originates from the same distribution as from which the data is trained. When objects from a different data distribution are classified, the output responses of the classifier are completely unpredictable. To prevent these unpredictable responses the novelties have to be detected.

Detection of novel objects can be done by estimating the input data density and rejecting the objects in low probability areas [BL78]. The density estimation can be based on a model of the data, for instance a mixture of Gaussian distributions, or it can be estimated by Parzen windows (see for instance [Bis95]). Both methods have their drawbacks. In a Gaussian mixture the number of kernels has to be chosen beforehand. The assumption of Gaussian distributions can be a severe approximation and in these cases a large number of kernels is necessary to make a reasonable approximation.

Parzen density estimation requires large numbers of train objects to make a reliable probability density estimation. It also requires a width parameter σ which determines how smooth the resulting probability density distribution is. This parameter is often chosen to be optimal over the complete feature space using a cross validation method. When large differences in density exist, the Parzen kernel method will give poor results in low density areas.

Another approach of outlier detection is to find bounded regions which contain (almost) all data ([MKH93]). These methods use restricted shapes for their class boundaries like hyperspheres. This limits how tight the boundary can be put around the class objects, especially when classes are far from circular distributed, and this limits the ability to discern between novel data and valid data.

In this paper we propose a new method to detect novelties, based on the instability of the outputs of a simple classifier. In this way objects in regions which are difficult to learn for a classifier, are identified.

In section 2 we present the three simple methods to detect novelties (mixture of Gaussians, Parzen estimator and a nearest neighbor based estimator), and the instability based method. In section 3 the methods are used in practice in a handwritten digit recognition problem. We summarize the conclusions in section 4.

2 Theory

2.1 Probability density or instability estimates

We assume we have a training (or learning) set of objects $\mathbf{x}_i^{tr}$, $i = 1, ..., N$, each object containing d feature values. Each of these training objects is independently drawn from one fixed probability distribution (the objects are identical and independent distributed).

In statistical pattern recognition several methods to estimate probability densities exist. Two very simple methods are the Gaussian mixture model and the Parzen windows (see for instance [Bis95]). In a Gaussian mixture containing G kernels the probability density is estimated by:

$$p(\mathbf{x}) = \sum_{i=1}^{G} \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}_i^{tr}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i^{tr})^T(\mathbf{\Sigma}_i^{tr})^{-1}(\mathbf{x} - \mu_i^{tr})\right) \qquad (1)$$

where d is the dimensionality of the feature vectors, μ_i^{tr} and $\mathbf{\Sigma}_i^{tr}$ the mean and the covariance matrix of kernel i in the training set respectively. In this paper one kernel for each of the classes is estimated, with a common covariance matrix to increase the accuracy of the estimations. Because this matrix has to be inverted when actual probabilities have to be calculated, the mixture of Gaussians can not be used when the number of objects per class is smaller than the dimensionality of the feature space and the covariance matrix becomes singular.

In the Parzen window approach, there are as many kernels as there are training objects and only the width σ of the kernels has to be estimated.

$$p(\mathbf{x}) = \frac{1}{N} \sum_{\mathbf{x}^{tr}} \frac{1}{\sqrt{(2\pi)^d}\sigma^d} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{x} - \mathbf{x}^{tr})^2\right) \qquad (2)$$

This σ is estimated using the leave-one-out method[FH89].

Instead of estimating complete probability densities, an indication of the local densities can be obtained by comparing the distance between the test object $\mathbf{x}$ and it's nearest neighbour in the training set $\mathrm{NN}^{tr}(\mathbf{x})$, and the distance between this nearest neighbour $\mathrm{NN}^{tr}(\mathbf{x})$ and it's nearest neighbor in the training set $\mathrm{NN}^{tr}(\mathrm{NN}^{tr}(\mathbf{x}))$. When the first distance is much larger than the second distance,

the object can be regarded as an outlier. We use the quotient between the first and the second distance as indication of the validity of the object:

$$\rho(\mathbf{x}) = \frac{\|\mathbf{x} - \mathrm{NN}^{tr}(\mathbf{x})\|}{\|\mathrm{NN}^{tr}(\mathbf{x}) - \mathrm{NN}^{tr}(\mathrm{NN}^{tr}(\mathbf{x}))\|} \tag{3}$$

where $\mathrm{NN}^{tr}\mathbf{x}$ is the nearest neighbour of $\mathbf{x}$ in the training set.

A new method is to use the instability of a simple classifier. By taking bootstrap samples the same size as the original training set, and by training several classifiers on these sets, the outputs of the different classifiers on the test set will differ. The variation in the outputs indicates how large the influence is of taking another training set. A large variation indicates that the object is hard to classify.

$$\rho(\mathbf{x}) = \mathcal{E}\left[\mathrm{out}^{tr}(\mathbf{x})^2\right] - \mathcal{E}\left[\mathrm{out}^{tr}(\mathbf{x})\right]^2 \tag{4}$$

where $\mathrm{out}^{tr}(\mathbf{x})$ is the output value of the classifier trained with bootstrap samples of trainset $\{\mathbf{x}^{tr}\}$ for object $\mathbf{x}$.

2.2 Finding the outliers

When probability densities are estimated and used to reject uncertain objects, the interpretation of the rejection threshold is clear: when the threshold is put on 1%, objects with probability of 1% or less are rejected. When another continuous measure, like the quotient between two distances or the stability of classifier outputs, is used, the threshold level is harder to derive.

In this paper we assume that the measured quantity on the objects is roughly Gaussian distributed over the objects. New objects are measured using the different methods and compared with the measurements of the training objects. When the difference between the new object and the mean of the train objects is larger than three times the standard deviation in the training distribution, the new object is rejected.

For the density estimations, the Parzen estimator and the Gaussian kernel estimation, this results in:

$$\begin{aligned} &\text{reject } \mathbf{x} \text{ if:} \\ &\quad \log(p(\mathbf{x})) < \mathcal{E}\left[\log(p(\mathbf{x}^{tr}))\right] - 3.0 * \mathrm{var}(\log(p(\mathbf{x}^{tr})) \end{aligned} \tag{5}$$

and for the distance based methods, the nearest neighbour method and the stability method:

$$\begin{aligned} &\text{reject } \mathbf{x} \text{ if:} \\ &\quad \rho(\mathbf{x}) > \mathcal{E}\left[\rho(\mathbf{x}^{tr})\right] + 3.0 * \mathrm{var}(\rho(\mathbf{x}^{tr})) \end{aligned} \tag{6}$$

3 Experiments

The outlier detection was performed on hand written digits. The original digits were scanned from nine maps from a Dutch public utility. The digits were

deskewed, normalized to fit into a 30 by 48 pixel region and thresholded. Of 2000 hand written digits (200 per class) several different feature types are measured. In this experiment we used two data sets, the Zernike feature set and the Karhunen-Loève feature set. The Zernike feature set contains 49 Zernike moments and 6 morphological features. The Karhunen-Loève feature set contains the first 64 principal components of the digit images (see for more explanation [BDT97]). From these feature sets nine digit classes are used as normal train and test data, the last digit class is considered the outlier-class.

For the stability method, a linear classifier is used based on maximizing Fishers criterion (see for instance [Rip96]). The classifier outputs the normalized sigmoid of the distance to the decision boundary. The two-class classifier is adapted to handle multi-class problems by training separate classifiers between one class and all other classes combined. To obtain one output value for the instability of an object, 25 bootstrap samples and classifiers were generated and the variation in the nine class outputs is averaged.

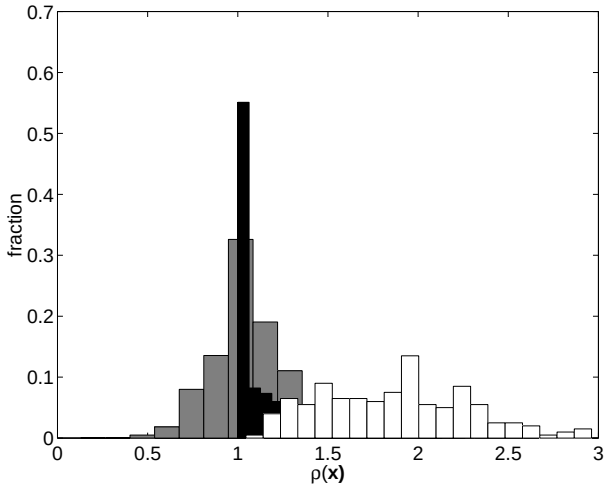

Fig. 1. Distribution of the nearest-neighbour-measure for the training set (black), the test set (gray) and the outlier set (white) for the Zernike data set with 50 train samples per class. In this case the threshold is at $\rho = 1.7$

The four different measures are compared using different training set sizes. For each training set size 5 training sets are draw and the results are averaged. The threshold is obtained by calculating the measures on the training sets. In almost all methods the assumption that the measure is approximately Gaussian distributed, is reasonably be met. Only in the nearest neighbour method the distribution of the measure in the training set is skewed to the left (see figure 1).

The measures are tested on the test set and the outlier set. The test set contains all available samples which are no outliers, including the training set. The final performance measure is the fraction of the outlier class that is rejected minus the fraction of the test set that is rejected. Good outlier detection means that the performance is close to one.

3.1 The results

In figures 2 and 3 the results of the four outlier detection methods for 10 different outlier classes is shown. From the class that is left out from the training set, the fraction rejected objects is plotted versus the number of training samples. The number of the class corresponds to the digit it represents.

The first thing that becomes clear, is the good performance of the Parzen estimator in the large dataset regime. This performance is biased, because for testing also the training data had to be used. Unfortunately for larger training set sizes this method becomes time and space consuming. All training objects have to be stored and for the processing of a new object distances to all training objects have to be calculated. Especially in high dimensional feature spaces this can be a burden.

Second important notice is that for classes 6 and 9 in the Zernike dataset all methods perform extremely bad. This is caused by the fact that the Zernike features are rotational invariant and the 6 and 9 become indistinguishable. The results of the Parzen estimator is largely due to the use of training samples in testing.

For small sample sizes the Gaussian model does not give an outcome as explained in subsection 2.1. Using more samples some classes can be clearly distinguished, for instance classes 0 and 8 in the Zernike dataset and classes 2 and 4 in the Karhunen-Loève dataset. For other classes this method does not work, classes $2, .., 6$ and class 9 in the Zernike dataset and classes $0, 7, 9$ in the Karhunen-Loève dataset. These classes are surrounded by the other classes in the training set.

The nearest neighbour method almost always performs poorly, only when classes are clearly distinguishable, like class 0 in the Zernike dataset, this method outperforms the other methods in relatively small sample sizes. This method seems to be sensitive to low density regions in the training classes. Due to the large internal sample distances, large portions of the feature space are then accepted as belonging to the class.

The performance of the instability method in general is not very good at large sample sizes, but at small sample sizes it outperforms all other methods. By using more samples, the variation in in output of the (linear) classifier on the bootstrap samples becomes smaller. Only data in the neighbourhood of the classifier experiences output variations, and large portions of the data just hides in the stable areas behind the train data clusters. Only a few classes can be distinguished using larger sample sizes, that are classes 7 and 8 in the Zernike set and class 4 in the Karhunen-Loève set. In general the performance on the Karhunen-Loève set is somewhat worse than on the Zernike set.

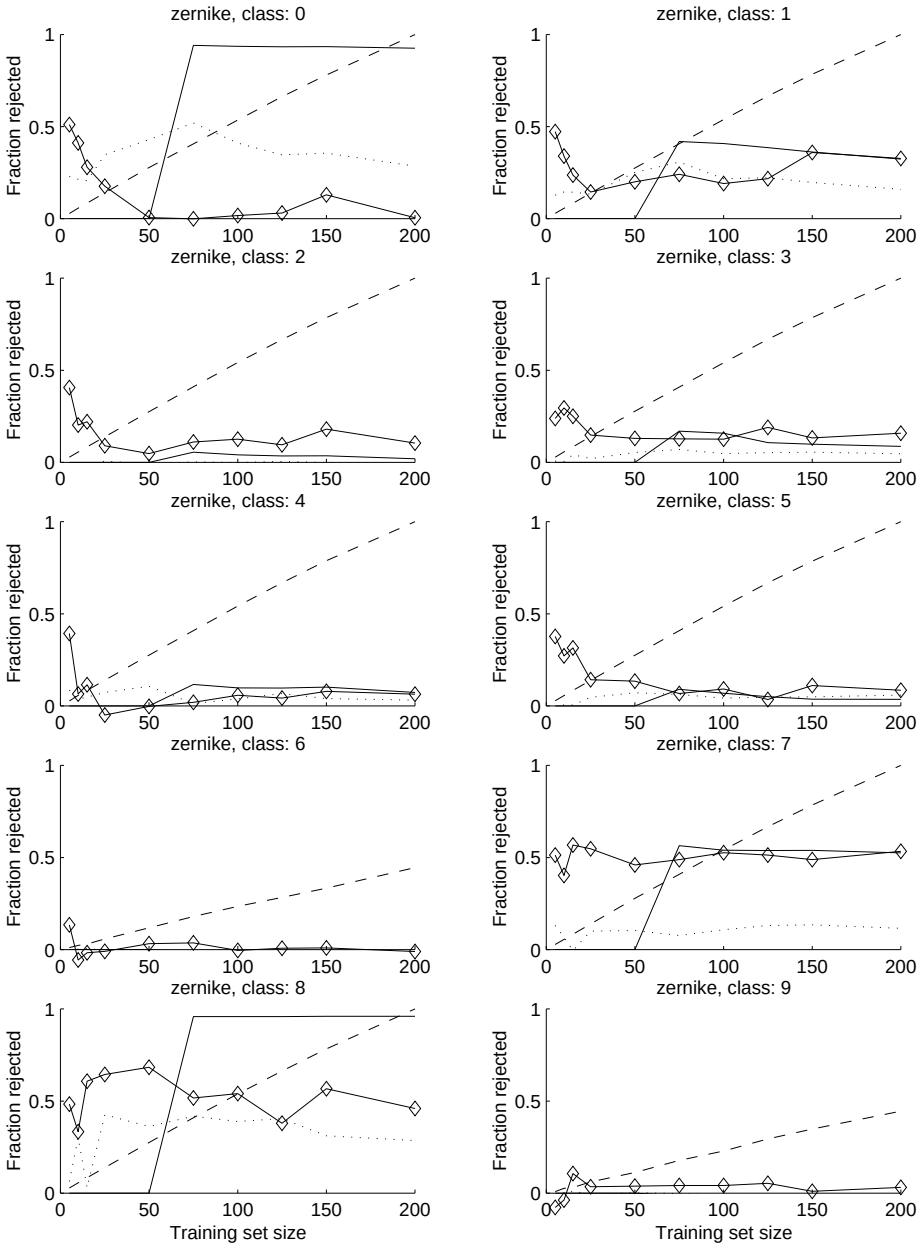

Fig. 2. Percentage of the outlier class rejected for different training set sizes using the Zernike feature set. solid line: mixture of Gaussians, dashed line: Parzen, dotted line: nearest neighbor, solid with diamonds: instability based.

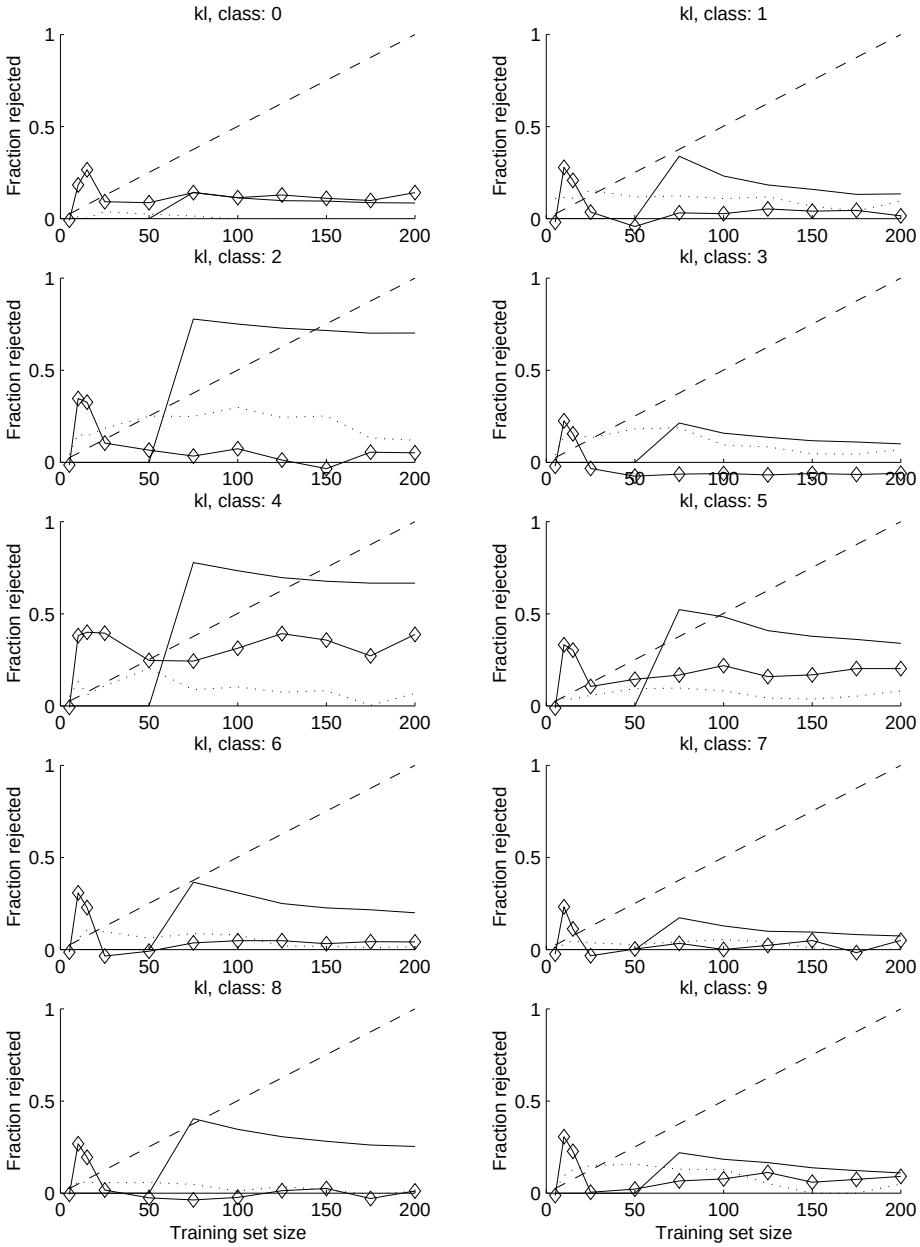

Fig. 3. Percentage of the outlier class rejected for different training set sizes using the Karhunen-Loève feature set. solid line: mixture of Gaussians, dashed line: Parzen, dotted line: nearest neighbor, solid with diamonds: instability based.

In the Karhunen-Loève set the classes are almost Gaussian distributed (a Gaussian classifier on all classes achieves a test error of 5.0%, on the Zernike feature set it is about 20.3%). Simple linear classifiers on the Karhunen-Loève set are more stable and especially in larger sample sizes the instability is not large enough to achieve as good results as in the Zernike feature set. To obtain also better results in larger sample sizes, the number of bootstrap samples have to be increased.

4 Conclusions

In this paper we presented a new method to detect outlier objects. This method uses the instability of the outputs of a simple classifier. Variations in outputs are obtained by training the classifiers on several bootstrapped versions of the training set. This method is compared with two models based on probability density estimation, a mixture of Gaussians and Parzen density estimation, and a method based on the estimation of local densities. In this last method the quotient between the distance to the nearest neighbor and the distance between the nearest and second nearest neighbor is used to detect outliers. The methods are compared on two feature set in a handwritten digits recognition problem, a Zernike feature set and a Karhunen-Loève feature set.

For large sample sizes the Parzen windows estimation is superior, at the expense of computing time and storage space. For some classes the Gaussian mixture model is superior for moderate sample sizes. The nearest neighbour method fails in almost all cases. Only well separated classes using smaller sample sizes can be distinguished.

For small sample sizes the instability method outperforms all other methods. Larger sample sizes deteriorate the performance of the instability method, because bootstrapped versions of a large training set do not result in much variation in the simple linear classifiers. Therefore the instability method does not detect all outlier objects, only those objects which are in areas of the feature space for which classification is hard. When an outlier class 'hides' in a stable region in the feature space, this method can not detect it. Also when a large number of training samples is used, a large number of bootstrap samples is needed to make a good estimation of the instability. But when simple classifiers are used, this will not be very expensive in calculation costs.

5 Acknowledgments

This work was partly supported by the Foundation for Applied Sciences (STW), the Foundation for Computer Science in the Netherlands (SION) and the Dutch Organization for Scientific Research (NWO).

References

[BDT97] Breukelen van M., Duin R.P.W, and Tax D.M.J. Combining classifiers for the recognition of handwritten digits. In Pudil P., Novovicova J, and Grim J, edi-

tors, *1st international workshop on statistical techniques in pattern recognition*, pages 13–18. Institute of Information Theory and Automation, June 1997.

[Bis95] Bishop C.M. *Neural Networks for Pattern Recognition.* Oxford University Press, Walton Street, Oxford OX2 6DP, 1995.

[BL78] Barnett V. and Lewis T. *Outliers in statistical data.* Wiley series in probability and mathematical statistics. John Wiley & Sons Ltd., 2nd edition, 1978.

[FH89] Fukunaga, K. and Hummels D.M. Leave-one-out procedures for nonparametric error estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):421–423, April 1989.

[MKH93] Moya, M.R., Koch, M.W., and Hostetler, L.D. One-class classifier networks for target recognition applications. In *Proceedings world congress on neural networks*, pages 797–801, Portland, OR, 1993. International Neural Network Society, INNS.

[Rip96] Ripley B.D. *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge, 1996.