IDIAP RESEARCH REPORT

Noise Robust Discriminative Models

Quan Le [a], Samy Bengio [a]

IDIAP–RR 03-40

September 2003

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 − 27 − 721 77 11
fax +41 − 27 − 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

[a] IDIAP, P.O. Box 592, CH-1920 Martigny, Switzerland

IDIAP Research Report 03-40

Noise Robust Discriminative Models

Quan Le , Samy Bengio

September 2003

Abstract. For classification problems, it is important that the classifier is trained with data which is likely to appear in the future. Discriminative models, because of their nature to focus on the boundary between classes rather than data itself, usually do not have the capability to deal with noisy training data. We propose the use of generative models as filters to make discriminative models more robust against noise. Firstly the distribution of the training data is estimated, then examples which do not satisfy some criterion, like having low likelihood, will be considered as outliers and discarded before training discriminative models. The idea was tested on a noisy data set from the UCI Machine Learning Repository.

1 Introduction

An important feature of a classifer is its ability to generalize. A good classifier is the one which gives good answer not only for training examples but also with unseen data. Discriminative approaches like Support Vector Machines (SVMs) are often favored to solve the classification problems because they directly optimize the discrimination criterion and find the boundary between classes, whereas the generative models solve the task by solving a more general task (estimating the distribution of data belonging to each class, then the decision will be induced using some criterion like the Bayesian Criterion). It is, however, the nature of focusing on the boundary between classes rather than data itself that makes discriminative models less robust when dealing with noisy data. Some anomalous pattern in the training data set (whose values are generated by flaws in data ascertainment, like faulty measurement instruments) can confuse discriminative models and cause the solution to be incorrect. Examples of problems with outliers can be found in [1].It is important to have some way to assess whether some training examples are abnormal and preprocess them before feeding training data to discriminative models.

We propose to use generative models to filter out bad examples (outliers) before training discriminative models. For our system, Gaussian Mixture Models (GMMs) were used as generative models and the discriminative models were the Support Vector Machines (SVMs).

The rest of the paper will be organized as follows. In the Section 2 we introduce the SVM and analyse the problem of dealing with noisy data. Section 3 describes the use of generative models for assessing noisy data. Our preliminary experiments will be presented in Section 4. Conclusions are drawn in section 5.

2 SVM and Noisy Data

The Support Vector Machine is a discriminative learning algorithm proposed by Vapnik [8],[3], and since then has been applied to many problems. The key idea in SVM is:

In case data is linearly separable, the SVM looks for the separating hyperplane with the largest margin, with respect to the labeled training set.

$$f^{max} = \arg\max_{f} \min_{i} \frac{y_i f(\mathbf{x}_i)}{\|\mathbf{w}\|}$$

$$\text{where } f(\mathbf{x}) = (\mathbf{x} \cdot \mathbf{w}) + b = \textstyle\sum_i \alpha_i y_i (\mathbf{x_i} \cdot \mathbf{x}) + b.$$
$$\mathbf{w} = \textstyle\sum_i \alpha_i y_i \mathbf{x}_i$$

for $\mathbf{x}$, $\mathbf{w} \in \Re^N$, $b \in \Re$, $\alpha_i \in \Re$ is the contribution of the sample i in the final solution , $y_i \in \{-1, 1\}$ are the labels corresponding to the training set $\{x_i\}$. α_i and b are determined in the training process. This is achieved by minimizing the square of l2-norm of $\mathbf{w}$

$$\frac{1}{2}\|\mathbf{w}\|^2 \qquad (1)$$

subject to the inequalities

$$(\mathbf{x_i} \cdot \mathbf{w} + b) y_i \geq 1$$

for all i.

Here comes the problem of the SVM, since it does not take into account "the importance" of training example and treats noisy data just as others, some bad examples can significantly affect the hyperplane solution and make it not generalize well with future data. This phenomenon is illustrated in figure 1

Another important feature of SVM is the soft margin, which is applied when training examples are linearly inseparable in feature space. To overcome this problem positive slack variables ξ_i are introduced into the inequalities such that:

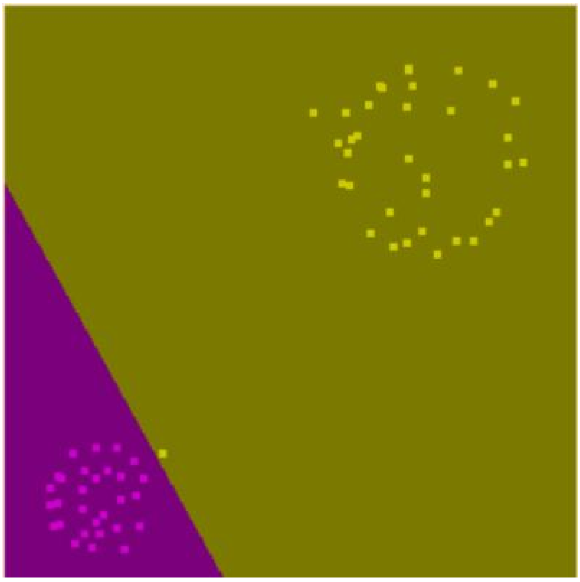

Figure 1: One bad example can change the solution, making it not generalize well to future data.

$$\begin{gathered}\mathbf{x}_i \cdot \mathbf{w} + b \geq 1 - \xi_i \text{ for } y_i = +1 \\ \mathbf{x}_i \cdot \mathbf{w} + b \geq -1 + \xi_i \text{ for } y_i = -1 \\ \xi_i \geq 0 \ \forall i.\end{gathered}$$

Then $\sum_i \xi_i$ is an upper bound on the number of training errors. A natural way for choosing the resulting problem to minimize is then

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i \tag{2}$$

subject to inequalities 2.

Parameter C is chosen by user and can be used as a trade off: a low value of C means we prefer the solution with a large margin (it will be more robust against noise), a high value of C means we prefer the solution with less classification errors on the training set (but small margin). Usually this is done by validation technique, for which various values of C in a predefined range of value will be tried, and the one which gives the best performance on a separate validation set will be chosen. However, the process of choosing C is not simple since it is not "invariant".

Proposition 1 *When we find the hyperplane in the input space, if the value of all data points is scaled up n times, in order to make the new C have the same effect as it is before scaling data one need to scale it down n^2 times.*

Sketched Proof 1 *After scaling up process we have $\mathbf{x}_i^{new} = n \cdot \mathbf{x}_i$, consider the hyperplane for the scaled up data set which correspond to the hyperplane in the orginal data set we have the margin will be scaled up n times. Hence the new normal vector $\mathbf{w}^{new} = \frac{1}{n}\mathbf{w}$.*

Since $\mathbf{w}^{new} = \sum_i \alpha_i^{new} y_i \mathbf{x}_i^{new}$. From equation 2 we can infer:

$$\alpha_i^{new} = \frac{1}{n^2}\alpha_i. \tag{3}$$

Minimizing the objective function (2) for the original data set is equal to maximizing the dual objective function:

$$L(\mathbf{w}, b, \alpha, \xi, \eta) = -\frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) + \sum_i \alpha_i \tag{4}$$

subject to $\sum_i \alpha_i y_i = 0$ *and* $0 \leq \alpha_i \leq C$.

The solution is the minimum of (2) and maximum of (4).

For the scaled up dataset, the new dual function would be:

$$L(\mathbf{w}^{new}, b, \alpha^{new}, \xi, \eta) = -\frac{1}{2}\sum_{i,j} \alpha_i^{new}\alpha_j^{new} y_i y_j (\mathbf{x}_i^{new} \cdot \mathbf{x}_j^{new}) + \sum_i \alpha_i^{new} \tag{5}$$

subject to $\sum_i \alpha_i^{new} y_i = 0$ *and* $0 \leq \alpha_i^{new} \leq C^{new}$.

From (3), for having the same hyperplane in scaled up data set which corresponds to the previous hyperplane of the original data set, the new value of C *would be* $C^{new} = \frac{1}{n^2}C$. *The corresponding value of the solution of the objective function will also be scaled down* $\frac{1}{n^2}$ *times.*

In case the hyperplane is to be found in the feature space, the choice of kernel and kernel parameters will decide the mapping feature space, so as the role of C in feature space. This means it can not be predefined in which range of value C should fall into. If there is a local minimum of the classification error rate in the range of value C for doing validation, the user would be misled and take the wrong value of C. So although the SVM has its own mechanism to deal with noisy data, it may not be good enough.

3 Outlier Detection based on Generative Models

To make discriminative models less affected by noisy data, our solution is to find out which data points in the training data set are outliers (or data points which are not likely to occur in the future) and reject them before training discriminative model.

Here the generative models' capability of modeling the distribution of data will be used. Once the distribution of data examples of each class has been estimated, the Bayes rule can be used to assess each example:

$$P(y|\mathbf{x}) = \frac{p(\mathbf{x}|y)P(y)}{p(\mathbf{x})}. \tag{6}$$

for which y is the label of the class, $p(\mathbf{x}|y)$ is the class based probability density function, $P(y)$ is the prior of each class, $p(\mathbf{x})$ is the likelihood of the example, and $P(y|\mathbf{x})$ is the posterior probability of class y.

From (6), various criterions can be used to decide whether an example is outlier. It could be the likelihood of the example given its true class (p(x/y)), the posterior probability of class given the example (P(y/x)), or p(x).. Other methods for outlier detections can be found in [5][7][2].

As for estimating the distribution of data, we choose the Gaussian Mixture Models (GMMs). They are a flexible semi parametric class of probability distributions. Given d-dimensions data examples the density of a GMM is defined as:

$$p(\mathbf{x}|\theta) = \sum_{j=1}^{k} \frac{w_j}{(2\pi)^{\frac{d}{2}}\sqrt{|\mathbf{\Sigma}_j|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \mathbf{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)\right). \tag{7}$$

where the parameter set of the GMM is $\theta = (w_j, \mu_j, \Sigma_j)_{j=1}^k$ with $\mu_j \in \Re^d, \Sigma_j \in \Re^{d^2}$ being respectively respectively the mean vector and the covariance matrix of the j^th Gaussian component . The prior probability of a data example belonging to the j^th Gaussian of the GMMs $w_j \geq 0 \in \Re$, therefore they need to fulfill the condition $\sum_j w_j = 1$. The number of Gaussian component k controls the capacity of the GMM, with k big enough the GMMs can approximate arbitrary distributions.

To learn the parameters of a distribution, usually the Expectation Maximization (EM) algorithm is employed [4]. The number of component in the GMMs can be chosen by validation method, in our case the criterion is the classification error.

4 Experiments

To check the idea, we chose a noisy data set from the UCI Machine Learning Repository [6], the Forest database. The problem is to classify data examples between seven major forest cover types. The data set has been preprocessed to turn from a multiple class problem to a binary classification task, the dimensions of data is 53. For our experiments, we chose 1000 examples as the training data set, 1000 examples as validation set and 1000 examples as test set.

Since the EM algorithm is known to be sensitive to initial parameter values, the K-means algorithm was used to initialize the parameters of GMMs. Diagonal covariance matrix GMMs were used to constraint the capacity of generative models. The number of Gaussian components for each class was varied and chosen by 5-fold cross-validation on the training data set (The validation data set was used for choosing other hyperparameters- the rejection rate and the variance of Gaussian kernels).The criterion for measuring the performance of GMMs systems was the classification error rate (using Bayesian Criterion for taking decision). This process ended up with 15 Gaussians GMMs for each class.

The noise rejecting process for the training data set would be as follows:

- Using the GMMs with number of Gaussian chosen previously to model data of each class.
- For each example in the training data set, compute its likelihood (or loglikelihood ratio) given its true class.
- For each class, sort the likelihood of all examples in descending order.
- The rejection rate r for each class was varied, for each r rejecting r% of examples in each class which has lowest values.

The training data set after rejecting outliers was used for training the SVM with Gaussian kernel. The rejection rate r was varied from 1% to 8% for each class, for each r the standard deviation σ of the Gaussian kernel was varied from 3 to 55. Firstly the rejection rate r and then the standard deviation σ was chosen based on the performance of the system on the validation data set.

Performance of the system on the validation data set with various rejection rates are shown in figure (2). Results of the validation process, in figure (3), show that the error rate of the discriminative models after rejecting 4% examples of each class were always better than without rejecting noisy data:

5 Conclusion

We analyze and point out some difficulties of the SVM when dealing with noisy data. A proposed solution is using generative models to filter out noisy data before training the SVM. The idea was tested on a noisy data set from the UCI Machine Learning Repository. For the future work the alternative approach might be to keep all data examples but use robust methods to avoid being led astray by outliers (such as weighting value of C on the optimization algorithm for each example).

Acknowledgment

The authors would like to thank the Swiss National Science Foundation for supporting this work through the National Center of Competence in Research (NCCR) on “Interactive Multimodal Information Management (IM2)”.

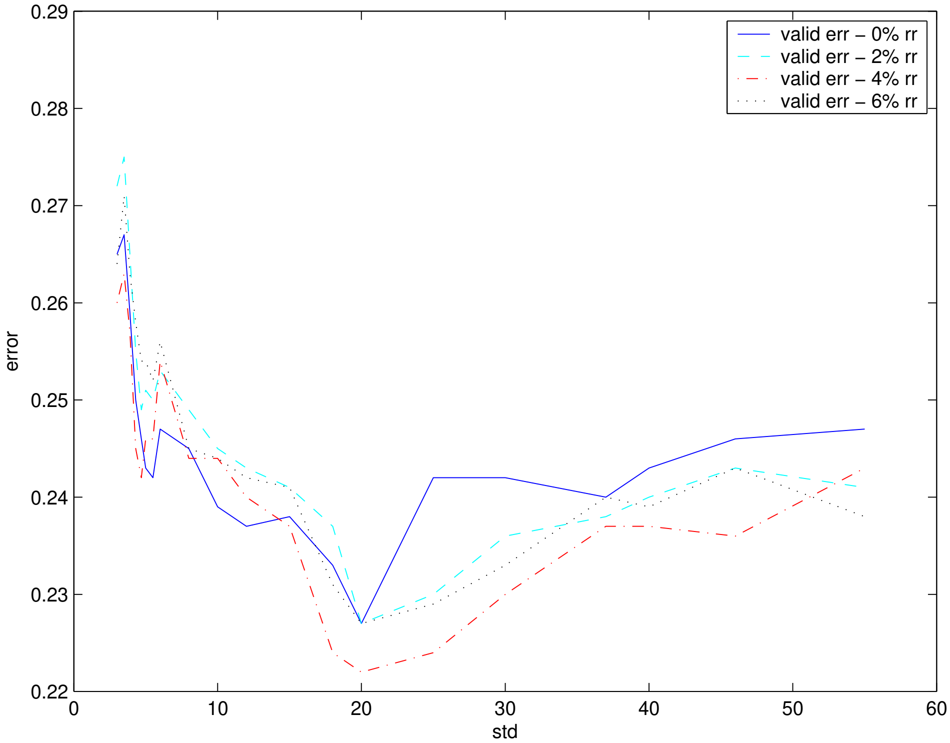

Figure 2: Performance of the system on the validation set with various rejection rate.

References

[1] V. Barnett and T. Lewis. *Outliers in Statistical Data.* John Wiley & Sons, 1994.

[2] Alex J. Smola Bernhard Schölkopf, Robert C.Williams and John Shawe Taylor.

[3] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data mining and Knowledge Discovery*, 2(2):1–47, 1998.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Jrnl. of Royal Statistical Society B*, 39:1–38, 1977.

[5] Martin Lauer. A mixture approach to novelty detection using training data with outliers. In *12th European Conference on Machine Learning*, 2001.

[6] P.M. Murphy and D. W. Aha. UCIrepository of machine learning databases. Technical report, University of California, Department of Information and Computer Science, 1992.

[7] D.M.J. Tax and R.P.W Duin. Outlier detection using classifier instability. In *Proceeding of Statistical Pattern Recognition*, 1998.

[8] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New-York, NY, USA, 1995.

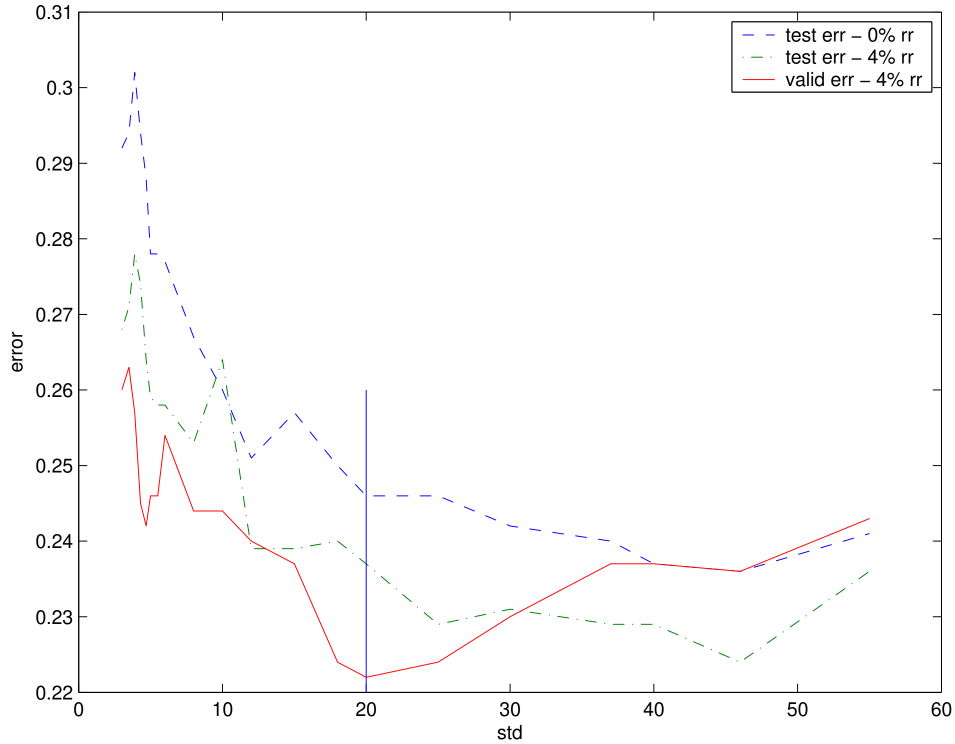

Figure 3: Performance of the system on the test data set, corresponding with 0 and 4% rejection rate, the std value was chosen from the validation set.