A Nonparametric Data Mapping Technique for Active Initialization of the Multilayer Perceptron

Aistis Raudys
Institute of Mathematics and Informatics
Akademijos 4, Vilnius 2600, Lithuania
E-mail: aistis.raudys@maf.vu.lt

Abstract. *A new nonparametric feature mapping technique for pattern classification is proposed and compared experimentally with a principal component and Sammon's mapping methods. We use the mapped training-set vectors for an active weights initialization of the multilayer perceptron classifier in a two-variate mapped space. Simulations have shown a usefulness of the proposed weights initialization method for designing the perceptrons when we need to obtain highly nonlinear decision boundaries.*

Key words. *Multilayer perceptron, training, initialization, data transformation, feature mapping, principal components, Sammon method.*

1. Introduction

One of principal difficulties that arises in the multilayer perceptron (MLP) training are a slow learning speed and a possibility to be trapped into a bad local minimum. In traditional training, the initial weights of MLP are generated by a random number generator in some narrow interval. Several authors, however, were trying to chose the weights from some definite heuristical considerations. It was noticed an active initialization affects a possibility to be trapped into a bad local minimum of the cost function. E.g. Raudys and Skurikhina (1992) used a piece-wise linear classifier to initialize hidden layer weights of the MLP classifier, and after training obtained 11.8% of errors in the generalization versus 21.9% of errors for a standard back-propagation with a random initialization (mean values of 10 independent experiments; an artificial computer generated data). Palubinskas (1996) suggested to initialize the weights of the hidden layer in a way that resulting hyper-planes of the hidden layer neurones would cut an input data feature space. He obtained better generalization results both on a synthetic XOR problem data as well as for a real remote sensing data. Karouia et al. (1995) used class-separability preserving feature vectors as the initial hidden layer weights and on a number of the real world and synthetic data sets showed that their new approach resulted lower generalization error and were less sensitive to network size and input dimension.

A present publication considers a possibility to initialize the weights using *a human ability* to analyse two variate data sets better than most sophisticated computer algorithms. Our main idea is to map the training-set data into a two-variate subspace, initialize the network in a man-computer interactive regime, and then - gradually, step-by-step to add remaining directions, and to return to the original feature space.

The paper is organised as follows. In Section 2, we briefly describe the main idea, and review the data mapping methods. In the third section, we present details of analysis of our original feature extraction method. Fourth and fifth sections contain simulation results with numerous artificial and real world data sets, and sixth one - a discussion.

2. Data mapping methods

To realise the active weights initialization idea we need to have a good data mapping algorithm that transform a vector **x** from the p-variate original feature space Ω_x into a *r-variate* vector **y** in a new feature space Ω_y ($r < p$) in such a way that the first two features are most informative ones, the third one - a little bit less, e.t.c. In this paper, we consider only linear transformations **y** = **Tx**, were **T** is a $r \times p$ orthonormal transformation matrix, such that **TT'**= **I** (an identity matrix).

Then we analyse the data in the 2-variate feature space of new components (directions) y_1 and y_2 of the transformed vector $\mathbf{y} = (y_1, y_2, y_3, y_4, \ldots, y_r)'$, initialize the MLP, save the weights of the perceptron. Afterwards we add the feature y_3, and keeping the weights of the perceptron constant, train additional, "third" weights of each neuron in the hidden layer. In the next step, we train all weights, and add the feature y_4, e.t.c..

In order to find new most informative directions we used two standard and developed two new *feature mapping methods.*

Principal component (PC) method often is known as discrete Karhunen-Loev expansion. It does not use information about the membership of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ to different pattern classes. Therefore, in spite of a large number of advantageous features that are characteristics to the principal component method (maximal accuracy of the representation of the pattern vectors and the covariance matrix, maximal entropy function if the pattern vectors are Gaussian; (see, e.g., Fukunaga, 1990), this method *destroys a separation* between the pattern classes sometimes.

In **Sammon mapping technique**, one seeks for a new direction $y_1 = \mathbf{t}_1\mathbf{x}$, that separates the training-sets of opposite classes at best. For this Sammon (1966) has used a standard Fisher linear discriminant (DF). In the second step, one constructs an orthonormal $2 \times p$ matrix $\mathbf{T}_2 = (\mathbf{t}_1' \ \mathbf{t}_2')'$. Note, two components of the p-variate vector $\mathbf{t}_2$ can be chosen arbitrarily. Further, one performs a transformation $\mathbf{y}_2 = \mathbf{t}_2\mathbf{x}$. In the third step, one seeks a new direction $\mathbf{y}_3 = \mathbf{t}_3\mathbf{x}$ where the training sets of the opposite classes are separated at best, forms a new (p-2)-variate space orthogonal to y_1 and y_2. This procedure is continued until a required dimensionality of the new transformed data is obtained.

Methods of the best and worst directions. The principal component method does not take into account the separability of the pattern classes, and therefore often destroys the separation of the pattern classes. The Sammon method performs well in case, when the pattern classes are unimodal, and the Fisher DF is a good classification rule for this kind of the data. The main objective of the MLP classifier is to construct nonlinear discriminat decision boundaries, where the pattern classes

are not linearly separable. In such situations, the Sammon feature extraction method can fail to obtain a small number highly informative new features. Therefore we suggest two new feature mapping techniques based on nonparametric density estimation.

Fukunaga (1990) describes a method to find the most informative directions $y_s = t_s x$. To evaluate the informativeness he uses the Parzen window (PW) estimate

$$\hat{f}(y_s \mid \pi_i) = \frac{1}{N_i} \sum_{j=1}^{N_i} \exp\{-\frac{1}{2} (y_s - y_{sj}^{(i)})'(y - y_{sj}^{(i)})/\lambda^2\}, \quad (1)$$

where λ is a smoothing constant, and a sum of squares error function

$$\sum_i \sum_j ((\hat{f}(y_s{}_j^{(i)} \mid \pi_1) - \hat{f}(y_s{}_j^{(i)} \mid \pi_2))^2. \quad (2)$$

The criterion (2) is different from a probability of misclassification - a main objective in the classifier design. It can lead to errors in determination of the best discriminative directions. Therefore, instead of seeking for the best discriminantive directions, we decided to seek for the worst ones. When in the new direction $y_s = t_s x$, the pattern classes differs negligibly, errors in inexact determinations of seperability by (2) are small. We hope, after sequential application of this procedure p-2 times remaining 2 directions will separate the pattern classes in a good way. We call this new original method - *a method of the worst directions*. The gradient descent optimization technique was used to find a minimum of the criterion (2).

In our research, we also modified the Fukunaga's nonparametric *method of the best directions* by introducing a new fast nonparametric feature informativeness measure. The nonparametric quality criterion is a multiextremal one. Therefore, in order to find the new directions quickly, we used a random search optimization technique. This algorithm appeared the most sucessful while applying to complex structured multimodal data, and is described in the third section with more details. An experimental comparison of the four feature mapping techniques is performed in the fourth section.

3. A method of the best directions

Main requirements to the feature extraction method for the active MLP initialization is an ability do discriminate *multimodal complex shaped pattern classes* in the bi-variate space, and *a high speed*. We used this approach to initialize the network, which will be trained afterwards. Later, we add the new informative directions. Thus, we can state weaker requirements to the seperability of the pattern classes just at the very beginning of the training process.

In our research, we have tested *following feature informativeness methods*:

a) the sum of squares error function (2),

b) the classification error estimated by the nearest neighbour rule in the leaving-one-out mode,

c) a function of distances between cluster centres c_{1i} and c_{2i} of the first and the second pattern classes $\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \frac{1}{0.00001 + d(c_{1i}, c_{2i})}$, where the cluster centres c_{1i} and c_{2i} have been found by the k-means clustering algorithm (see e.g. Fukunaga, 1990), and m_1, m_2 are the number of clusters in the learning-sets of the first and second class, respectively.

d) a number of the learning-set vectors which have changed their cluster membership after transforming the p-variate vectors $\mathbf{x}_j^{(i)}$ into the two-variate space of the "best seperability". In the last two criteria, we have rejected clusters that contained less than 3 learning-set vectors.

Our simulation studies performed by using a number of artificial and real-world data sets have shown that *all four criteria can be useful in detecting the data structures in the two-variate space*. The first two criteria are rather slow and impractical while applied to high-dimensional data sets. When the number of clusters in each pattern class reaches 15-20, the fourth criterion is pretty fast, and allows to reveal complicated data structures of complex mutimodal data sets. Therefore, this criterion was chosen as a main one in our research work.

All criteria are nonparametric in their nature and, at the same time, they are multiextremal ones. Their direct gradient type minimization does not lead to a global minimum. Therefore, in order to find new informative directions quickly, we used a *random search optimization technique* performed in an iterative way. In iteration α, we modified the transformation $2 \times p$ matrix $\mathbf{T}_{2(\alpha)} = [\mathbf{t}_{1\alpha}', \mathbf{t}_{2\alpha}']'$ by adding a small zero mean symmetrical noise, composed from a mixture of Gaussian components differing in their variances. The fastest convergence was achieved when we added a noise only to two components of each transformation vector ($\mathbf{t}_{1\alpha}$ or $\mathbf{t}_{2\alpha}$). In each iteration, we generated m_{iter}=1000 new vectors, estimated the criterion (d) for all of them, and then - choosed the best one. Then the variance of the noise was reduced, and a next, (α+1)-th, iteration was performed. As a starting vector $\mathbf{T}_{2(0)}$ we used: a) a random hyperplane that contains three randomly chosen points of the learning-set, b) a best hyperplane selected from m_0=1000 random ones, c) a best pair of original variables, d) the optimal Sammon's hyperplane, e) the best two eigendirections. The second, (b), strategy was most successful.

4. An experimental comparison of four mapping techniques

In order to reveal positive and negative peculiarities of the four feature mapping techniques investigated, we performed a number of simulation experiments by using both artificial and real-world data sets.

Types of **the *artificial* data sets**.

a) the pattern classes are a mixtures of Gaussian $N(\mu_{ij}, \Sigma)$ components in a bi-variate space; $\Sigma = \mathbf{I}(1-\rho)+\mathbf{E}\rho$ ($\mathbf{E}$ is $p \times p$ matrix composed from ones, and ρ is a correlation coefficient). Remaining p-2 features are Gaussian $N(0, \mathbf{I}\sigma^2)$. Then the data was rotated by a random orthogonal transformation $\mathbf{T}_{rot}$. Different combination

of the parameters p, μ_i, σ^2, ρ were tested. As an extreme model of this category, we mention a model where all the means μ_{11}, μ_{21}, μ_{12} are situated on a straight line, and any linear classification method is useless.

b) the first two features are generated in the same way as in a). Remaining p-2 features are functions of the first two first features, and, in addition, the Gaussian $N(0, I\sigma^2)$ noise is added. Different combinations of the parameters p, μ_i, σ^2, ρ, and the p-2 feature generation formulae were tested,

c) the first two features are spherical Gaussian density split into two pattern classes by a highly non-linear boundary (a contour of a palm, saw, e.t.c). A gap (a margin) between the pattern classes was formed, and the data was rotated, nonlinearly folded, and a small noise was added to remaining p-2 features.

The *real-world* data sets.

a) a vowels data; 28 spectral and cepstral features; in one pattern class we had 400 vowels pronounced by 20 speakers,

b) a lung noise data; 66 spectral and cepstral features; in one pattern class we had 180 vectors measured on 18 patients,

c) 47-variate data representing zernike moments for hand-written digits "3" and "8" recognition,

d) 51-variate data representing 51 discrete signal measurements used for a blind signal (a pulse signal, and a noise) separation,

e) 64 velocity and acceleration characteristics of machines vibration used for their classification into "good" and "bad" ones. The training-set contains $N_{1l} = 63$, $N_{2l} = 151$ vectors, and the test-set - $N_{1t} = 45$, $N_{2l} = 145$ vectors.

The four mapping techniques under investigation were applied to the artificial and the real-world data sets. Obviously, for each mapping technique one can construct such artificial data set for which this particular mapping technique outperforms the other methods. All four mapping methods can reveal the data structure only in cases where the data seperability is hidden in a subspace of dimensionality two.

The principal component method is fast and performs very well if all data structure is contained in a hyperplane, and the data variances in all other directions are small. The Sammon method is as fast as the principal component method, and reveals the seperability of unimodal pattern classes comparatively well. This method, however, fails if the optimal decision boundary is highly nonlinear one. The method of the worst directions allows to detect the nonlinearity of the discriminant hyperboundary, however a present, the gradient minimization based version of this method is very slow and impractical for high-dimensional applications. The method of the best directions is rather fast, and allows to detect the nonlinearity of the discriminant hyperboundary. Apparently, this method can be improved by using more sophisticated optimization algorithm.

As typical example of a difficult task in the feature mapping we present in Fig. 1 *a*, *b*, *c*, *d* four scatter diagrams of the learning-sets of the machines vibration data mapped into two-dimensional space by: the principal component method, the Sammon method, the method of the best directions just after start (the first $m_0 = 1000$ random generations), and at the very end of the iteration process.

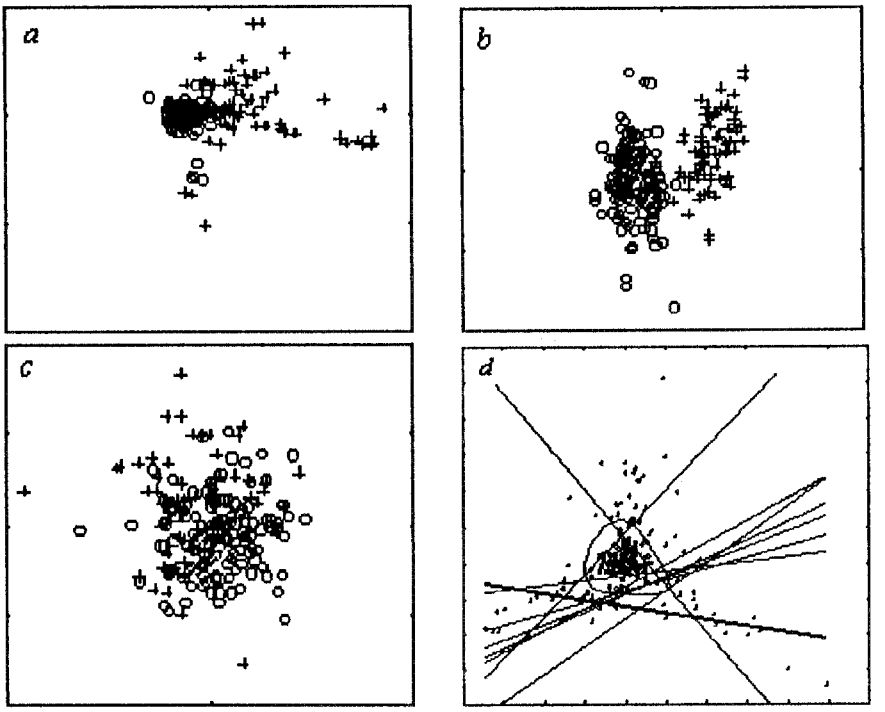

Fig.1. Scatter diagrams of the learning-sets of the machines vibration data mapped into two-dimensional space by: the principal component method (*a*), the Sammon method (*b*), the method of the best directions: *c* (at the start) , and *d* (at the end).

5. Experiments with MLP weights initialization

The active weights initialization procedure. We tested several principal possibilities of the active weights initialization in the two variate feature space. *At first*, while applying the MLP classifier, we had drawn a piecewise decision boundary on a computer screen in an interactive regime by using a mouse, and then we calculated the MLP hidden layer weights corresponding to each segment of the piecewise linear decision boundary. Afterwards, we were keeping the weights of the hidden layer constant, and trained only the weights of the output layer. We have found this approach to be ineffective.

Much more effective is a use of the piecewise linear, operator controlled, decision boundary to classify the training-set vectors, and then to select only correctly classified vectors. We use these vectors for the MLP initialization by conventional back propagation training. In many practical problems, the training-set is small. Therefore, we were adding a small zero mean and small variance Gaussian noise to each training vector several times, and in such a way we multiplied (regularized) the training-set. Then we classified the new bi-variate training-set by our "operator designed" piecewise linear decision boundary and presented for the

MLP classifier initialization only correctly classified vectors - the regularized and edited training-set.

As *a third technique* of the active weight initialization we used a simple multistart training of the MPL classifier with a subsequent selection of the best variant, and an operator controlled pruning of unnecessary hidden layer neurones in the bi-variate space. This technique appeared to be highly efficient.

As an illustration in Fig. 1*d*, we present the MLP classifier nonlinear decision boundary obtained in one of the experiments with the machines vibration data. Straight lines represent boundaries corresponding to each single neurone, and a nonlinear line - to a desision boundary of MLP. We see, several hidden neurones can be rejected without any damage to the performance of the non-linear MLP classifier.

Results. We used *two dozens of artificial and five real data sets* to test feature extraction algorithms and the active MLP initialization procedures. It is not difficult to construct artificial data sets where the active initialization is a useful tool. In four experiments with the real-world data sets out from five ones, however, our approach resulted a small or practically no gain in comparison with the conventional (random) MLP classifier initialization. A reason is very simple - for these four real-world problems the linear classifier was good enough to obtain a good separation of the pattern classes. There use of the MLP classifier instead of the single layer perceptron did not allow to reduce the generalization error substantially. The fifth real-world data set, however, required the non-linear decision boundary: the Fig. 1 *d* shows that good machines are situated in a centre, and bad ones are outside. The successful mapping of the data into the bi-variate space by means of the method of the best directions helps to choose a good architecture of the network and to reduce the generalization error. A part of our experimental results of application of different initialization strategies are presented in Table 1.

Table 1. Classification errors (in percents) and standard deviations of MLP classifier.

#	Feature space, initialization method	Training-set	Test-set
1.	Original 64 feature space, random initialization	0.5	12.5 (0.9)
2.	6 principal components space, random initialization	0.0	13.1 (0.4)
3.	6 best Sammon's directions, random initialization	0.0	13.6 (0.6)
4.	Individually best 6 features, random initialization	1.0	10.8 (0.9)
5.	Best 6 directions, random initialization	0.0	11.6 (1.1)
6.	Best 6 directions, active initialization	0.0	7.5 (0.7)

A selection of six individually best features was performed using criterion *d*) - the number of the learning-set vectors which have changed their cluster membership after the feature transformation into one-variate spaces. From the Table 1 we see that initialization and training the MLP in the two-variate space obtained by using the non-parametric criterion *d*) and subsequent increase in the number of dimensions up to 6 is much more effective than use of the conventional parametric the feature mapping and the perceptron initialization techniques.

6. Conclusions

A general conclusion which can be drawn from our investigation is rather trivial - the successful feature space reduction, and the network's initialization with subsequent its training in the low-dimensional space speeds up the training process and reduces the generalization error. *The active initialization of the feedforward network in the bi-variate mapped space is useful only when the data structure really is hidden in a subspace of dimensionality two, and when one needs to design a highly non-linear decision boundary.* We have met such situation only in one real world problem out of five problems investigated.

The principal component and the Sammon methods are fast. In practical situations, the principal component method with the small number of the first components usually fails to reveal separability of the pattern classes. The same can be said about the Sammon method, when pattern classes are described by complex multimodal distributions. Our new method of the worst directions performs well when other two classical methods misfire, but is very slow. The method of the best directions is comparatively fast and allows to reveal complex structure of the data distributions. However, it should not be applied in simple situations.

For practical applications, we recommend to use the Sammon method at first, and the nonparametric method of the best directions later, if the first of them fails.

Acknowledgements

The author thanks to Prof. B.Sankur from Istanbul Bogazici University, Dr. A.Rudžionis from the Speech Analysis Laboratory of Kaunas UT, Dr. R.P.W.Duin from Department of Applied Physics, Delft UT for providing the real world data sets for the experiments, his scientific supervisors V.Dičiūnas and Prof. Š.Raudys for useful discussions, an aid in formalising computational algorithms and a help in preparing a final version of the paper for publication.

References

Fukunaga, K. (1990). Introduction to Statistical Pattern Recognition. NY: Academic Press.

Karouia M., T.Denoeux and R.Lengelle. (1995). Influence of Weight Initialization on Multi-layer Perceptron Performance. *Proc. ICANN'95*, October 9-13, 1995, Paris.Vol **1**, 33-38,

Palubinskas G. (1996). On Weights Initialization of Back-propagation Networks. *Neural Network World,* **6**(1), 89-100.

Raudys S. and M. Skurichina (1992). The role of the Number of Training Samples on Weight Initialization of Artificial Neural Net Classifier. *Proc. of I-st Russian & IEEE Conf. on Neural Networks,* Rostov-na-Donu, Russia, 1992, IEEE Publication.

Sammon, J.W. (1970). An Optimal Discriminant Plane. - IEEE *Trans Comp.* **C-19**, 826-829.