EFFECTS OF DATA REDUCTION METHODS ON THE PERFORMANCE OF STATISTICAL NEURAL NETWORKS IN MEDICAL APPLICATIONS

Gökhan Bilgin Bülent Bolat Tülay Yıldırım
e-mail: gokhanb@ce.yildiz.edu.tr[1] e-mail: bbolat@yildiz.edu.tr[2] e-mail: tulay@yildiz.edu.tr[2]
Yıldız Technical University, Faculty of Electrical and Electronics, [(1)]Department of Computer Engineering,
[(2)]Department of Electronics & Communication Engineering, 34349, Besiktas, Istanbul, Turkey

Key words: Dimension reduction, statistical neural networks, linear projections, nonlinear kernel methods

ABSTRACT

Statistical neural networks are good pattern recognition structures. Basic problems of these networks which are known for their high accuracy results in most applications are excessive memory consumption and computational complexity. One way of solving computational complexity problem is to reduce the dimensions of feature vectors. Since developing medical data acquisition tools produce increasing amount of data, interest of scientists has been attached to this problem. In this work success of different types of data reduction methods have been examined and compared. As a result of experiments it is figured out that it is possible to reduce the computational load without lowering accuracy in these networks via data reduction.

I. INTRODUCTION

Statistical Neural Networks are known as good classifiers. In many applications, these networks can perform better than others. However in these structures, memory requirement and computational complexity are extremely high. By adding new instances to the training set, these problems grow faster. To maintain these weaknesses the training set may be reduced, but a reduced training set may lower the accuracy of the network.

Being a solution to that problem, size of the feature vector may be shrunken by some rule, which is called as dimension reduction. In the literature there are many dimension reduction techniques such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernel-PCA (KPCA) and Kernel-LDA (KLDA). In this work, the effects of these reduction techniques on statistical neural networks are analyzed. To compare the performances of these methods, three datasets which are taken from UCI MLREP Database [1] (which are New Thyroid, Pima Indian Diabetes and Wisconsin Breast Cancer -WBCD-) are used. Since these datasets are well-known, details of them are not included in this work; but for further information, see [1-7].

Section 2 describes the statistical networks briefly. Dimension reduction methods are introduced in section 3. In sections 4 and 5, details of the applications and results are proposed.

II. STATATISTICAL NEURAL NETWORKS

Simply, statistical neural networks combine statistical techniques with neural network structures. Radial basis function neural networks (RBF), generalized regression neural networks (GRNN) and probabilistic neural networks (PNN) are main well-known supervised structures which are explained briefly following in subsections

RADIAL BASIS FUNCTION N.N. (RBFN)

RBF Networks typically have three layers: an input layer, a hidden layer and an output layer. Input and output layers are related to the input vector space and the pattern classes respectively. Hence, the entire structure is degraded to determine the hidden layer's centers and weights between hidden and output layers. Activation function of jth hidden layer neuron is defined by a center (C_i) and a bandwidth (σ_i). The activation function is a Gaussian curve defined as

$$\varphi_j(X) = \exp\left(-\frac{\left\| X - C_j \right\|^2}{2\sigma_j{}^2} \right) \qquad (1)$$

Output of the jth output neuron is found by

$$s_j(X) = \sum_{i=1}^{K} w_{ij}\varphi_i(X) + b_j \qquad (2)$$

where ω_{ij} is the weight between the ith hidden neuron and jth output neuron, K is the number of the neurons in the hidden layer [8].

GENERALIZED REGRESSION N.N. (GRNN)

GRNN is a special case of RBF where the centers and bandwidths are determined by the training data. Alternatively, learning phase of a GRNN is not iterative. In the GRNN structure, ith training vector x_i is the center of the i^th gaussian kernel (RBF unit). For a given **x**, output of the i^th RBF unit is

$$\beta_i = \exp\left[-\frac{(x - x_i)^T (x - x_i)}{2\sigma^2} \right] \tag{3}$$

where (σ) is a smoothing parameter determined by the user. The output of the entire network is

$$y = \sum_{i=1}^{K} \alpha_i y_i \tag{4}$$

where $\alpha_i = \frac{\beta_i}{\sum_{i=i}^{K} \beta_i}$, and y_i is the desired output of the network for x_i.

If **x** is close enough to i^th training vector $\mathbf{x_i}$, $\boldsymbol{\alpha_i}$, which is related to $\mathbf{x_i}$, becomes maximum and the desired output y is close enough to y_i [9].

PROBABILISTIC N.N. (PNN)

PNN [10] is also known as Bayes-Parzen Estimator. Output of the PNN for a given feature vector x is determined by Bayes Decision Rule. X is a member of the i^th class if

$$\frac{f_i(x)P_i}{L_i} > \frac{f_j(x)P_j}{L_j} \quad \text{for all } j \neq i \tag{5}$$

where $f_i(x)$ is the probability density function, P_i is the occurrence probability and L_i is the misclassification probability of i^th class respectively. If the density functions of the classes are known, this equation can be solved. To realize the density functions, a nonparametric estimation technique known as Parzen Windows is used:

$$F(x) = \frac{1}{(2\pi)^{m/2} \sigma^m n} \sum_{i=1}^{n} \exp\left[-\frac{(x - x_i)^T (x - x_i)}{2\sigma^2} \right] \tag{6}$$

III. DIMENSION REDUCTION

In this work all classifiers are good for pattern recognition but their memory consumption is extremely high. One way to reduce memory consumption is adjusting data size in a proper way. Reducing the dimension of feature vectors is especially important where lowering training set is not allowed to lower. Dimension reduction can be made by linear and nonlinear transforms. Geometrical distribution of data on feature space is the key for choosing which transform can represent this distribution better.

LINEAR PROJECTION METHODS

PCA and LDA are two important statistical feature extraction tools for the structure of evaluating data. Basically, a set of correlated variables is transformed into a set of uncorrelated variables, which are ordered by decreasing variability. PCA is an unsupervised reduction method, so it doesn't need labeled training. Given a data set of *n* *d*-dimensional observations $\boldsymbol{X}=\mathbf{x_1,\ldots,x_n}$, $\mathbf{x_i} \in R^m$. The mean vector **μ** and *d*x*d* covariance matrix Σ is given by

$$\mathbf{\Sigma} = \frac{1}{n} \sum_{k=1}^{n} (\mathbf{x}_k - \boldsymbol{\mu})(\mathbf{x}_k - \boldsymbol{\mu})^t \tag{7}$$

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{k=1}^{n} \mathbf{x}_k \tag{8}$$

Eigenvectors and eigenvalues of this covariance matrix are computed. Eigenvectors represent basis of the new vector space, and their corresponding eigenvalues give the "significance" of each eigenvector. Usually there are just a couple of large eigenvalues, and the rest of eigen values can be ignorable. Eigenvalues are sorted in decreasing order; call the largest eigenvalue as λ_1 and corresponding eigenvector as e_1, the second eigenvalue λ_2 with eigenvector e_2, and so on ($\lambda_1<\lambda_2<\ldots<\lambda_m$). The top k eigenvectors are chosen and the rest of the dimensions is assumed as they generally contain noise. We form a dxk projection matrix A, whose columns consist of the selected eigenvectors. The original data is projected onto new principal components space by

$$\mathbf{x}' = \mathbf{A}^T (\mathbf{x} - \boldsymbol{\mu}) \tag{9}$$

where **x′** is the projection of observation vector **x** and A^T is the transpose of A.

Geometrically, data points form as a d-dimensional, hyperellipsoidally shaped cloud. Eigenvectors of the covariance matrix are the principal axes of this hyperellipsoid. Principal component analysis selects the directions along which the variance of the cloud is greatest [11].

Let the labeled training data, $\boldsymbol{T} = \{(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_n}, \mathbf{y_n})\}$, $\mathbf{x_i} \in R^m$ and $y \in Y = \{1, 2, \ldots, c\text{: class number}\}$. A subset of labeled dataset which belongs to only one class can be shown as $I_y=\{i\text{: } y_i=y\}$, $y \in Y$. Inner-class S_w and between-class S_b scatter matrixes can be defined as in (10) and (11).

$$S_W = \sum_{y \in Y} \sum_{i \in I_y} (x_i - \mu_i)(x_i - \mu_i)^T \tag{10}$$

$$S_B = \sum_{y \in Y} \left| I_y \right| (\mu_y - \mu)(\mu_y - \mu)^T \tag{11}$$

μ in the equation represents general mean vector for whole data, and $\boldsymbol{\mu_y}$, $y \in Y$ shows class mean vector.

The main aim of LDA is to find linear projection of data ($z=W^Tx$) and, connected with this projection, to maximize the class discrimination criterion (12).

$$F(W) = \frac{\det(\tilde{S}_B)}{\det(\tilde{S}_W)} = \frac{\det(S_B)}{\det(S_W)} \tag{12}$$

$\tilde{S}_B$ inner-class and $\tilde{S}_W$ between-class are new scatter matrixes obtained by data projection [12-13].

NONLINEAR KERNEL METHODS

Well known linear transforms (PCA and LDA) can be improved for dimension reduction by kernel functions. Traditional methods, PCA, LDA and their derivatives in literature, can result successfully where linear projections are adequate for reduction. If data distribution forms the maximum variance with a nonlinear projection, PCA and LDA will be insufficient for this kind of applications. To overcome this situation, data should be mapped nonlinearly to another feature space ($\phi : X \to \phi(X)$) [14].

With this mapping operation feature vectors pass over a higher dimensional space: $\phi : R^m \to R^n$, $n > m$. After transform it is assumed that data has zero mean for computational simplicity. Thus joint-variance matrix can be computed as in

$$\Sigma = \frac{1}{m}\sum_{k=1}^{m} \phi(\mathbf{x}_k) \cdot \phi^T(\mathbf{x}_k) \tag{13}$$

Mathematically, eigenvector and dimension that maximizes variance are in same direction. For this reason equation $\Sigma v = \lambda v$ must be solved. First of all, eigenvalue vector is assumed nonzero ($\lambda \neq 0$), so eigenvector v can be defined as a linear combination of $\phi(x_k)$,

$$v = \sum_k \alpha_k \phi(x_k) \tag{14}$$

So that,

$$\Sigma v = \lambda \sum_k \alpha_k \phi(x_k) \tag{15}$$

After substitution of statements, nonlinear principal component analysis can be expressed with a common kernel function definition ($K(\phi_i, \phi_j) = \phi(x_i) \cdot \phi(x_j)^T$) as,

$$y_i = v^T x_i = \sum_{j=1}^{m} \alpha_j K(x_i, x_j) \tag{16}$$

In this way nonlinear projections can be realized by kernel functions on higher dimensional spaces.

With a similar approach kernel function adaptation of LDA can be obtained. Because of transformed representation of original data space, inner-class (S_W^ϕ) and between-class (S_B^ϕ) scatter matrixes must be computed according to this transformed space representation. Consequently, on the high dimensional space, it is obtained maximum inner-class discrimination and minimum between-class variance as possible as it can [15, 16].

IV. APPLICATION

In this work, effects of different dimension reduction methods on the performance of the statistical neural networks were analyzed through biomedical data. The datasets were taken from UCI MLREP Database. At the first step, missing valued datasets were considered. If a dataset has missing values, these values were filled by inner class mean imputation method [17].

In this work, as a linear projection method, unsupervised PCA and supervised LDA were used. Furthermore, as a kernel approach (i.e. nonlinear projection) unsupervised K-PCA and supervised K-LDA were utilized on reduction evaluation. RBF and Polynomial kernel functions are considered in K-PCA and K-LDA.

Approximately 66% of the datasets were used for training; remaining portions were used for test sets. In the first stage of learning phase, optimum spread values for all neural classifiers were found by a trial-by-error strategy by randomly selected training sets. For comparison purpose between original and reduced datasets, original maximum test accuracy results which are found by optimum spread value for each classifier, are shown in Table-1.

Table 1: Test accuracies in percentages for original datasets

	Thyroid	**WBCD**	**Pima**
PNN	92,21	97,13	69.89
GRNN	89,61	87,70	69,89
RBF	68,83	65,98	65.06

Table 2: Maximum test accuracies in percentages for reduced Thyroid dataset

	PNN	**GRNN**	**RBF**
PCA	92,21 (4)	89,61 (4)	68,83 (3)
LDA	92,21 (3)	90.91 (3)	68,83 (3)
K-PCA(rbf)	68,83 (3)	68,83 (3)	29,87 (3)
K-PCA(poly)	92,21 (4)	89,61 (4)	68,83 (3)
K-LDA(rbf)	68.83 (3)	70,13 (4)	18,18 (3)
K-LDA(poly)	88.31 (3)	77,92 (3)	68,83 (4)

Table 3: Maximum test accuracies in percentages for reduced Pima dataset

	PNN	**GRNN**	**RBF**
PCA	72,49 (4)	72,49 (4)	65,06 (3)
LDA	73,98 (3)	73,98 (3)	65,06 (3)
K-PCA(rbf)	65,06 (3)	65,06 (3)	64,68 (3)
K-PCA(poly)	72,49 (4)	72,49 (4)	65,06 (3)
K-LDA(rbf)	65,06 (4)	65,06 (3)	88,10 (3)
K-LDA(poly)	65,56 (3)	65,56 (3)	65,06 (3)

Table 4: Maximum test accuracies in percentages for reduced WBCD dataset

	PNN	**GRNN**	**RBF**
PCA	98,36 (2)	97,95 (3)	88,93 (3)
LDA	98,77 (3)	97,95 (3)	85,25 (2)
K-PCA(rbf)	65,57 (2)	97,54 (2)	97,95 (2)
K-PCA(poly)	98,36 (2)	97,95 (3)	92,62 (2)
K-LDA(rbf)	95,49 (3)	96,72 (3)	97,54 (3)
K-LDA(poly)	98,77 (2)	97,95 (2)	90,98 (3)

At the final phase of the work, dimensions of the datasets were reduced by using different reduction methods which were explained in previous section and reduced datasets were obtained. Test accuracies of the reduced datasets were shown in Tables 2-4. Reduced sizes of the feature vectors which give the best test accuracies are shown in parentheses.

IV. RESULTS AND DISCUSSION

The main disadvantages of the statistical neural networks are high memory requirement and computational complexity. To solve this problem, size of the feature vectors can be reduced by using dimension reduction techniques. As seen in Table 1, PNN gives the best results and RBF gives the worst results for original datasets. By comparing the Tables 2-4, PNN is the best classifier again. For the reduced Thyroid dataset, the best accuracy is obtained by using LDA and PNN as 92,21%. In this case, size of the feature vector is just 3. For the Pima, the best score is obtained by K-PDA (rbf) and RBF as 88,10%; and this result is the only one which RBF outperforms the others. For WDBC, PNN is the best classifier again. In this case, LDA and K-LDA (poly) give the best results as 98,77%. Feature vector sizes are 3 and 2 for LDA and K-LDA (poly) respectively. These results are better than the accuracies of original datasets. On the other hand, by comparing the effect of the reduction methods respect to accuracies, it is seen that LDA is better than others in general. By combining these results, it is clear that the dimension of the feature vector is an important parameter for the statistical neural networks.

V. CONCLUSION

Especially traditional linear and kernel based dimension reduction methods are draw attention for many scientific application areas in last decade because of increasing amount of produced data by developed tools. In this work, effects of both linear and nonlinear techniques have been evaluated in biomedical datasets with statistical neural network structures. Results show that, with increased gain or considerable loss in accuracy, reduction techniques can be applied as a preprocessing technique for the statistical neural networks.

REFERENCES

1. D. J. Newman, S. Hettich, C. L. Blake and C. J. Merz, UCI-Repository of machine learning databases, http://www.ics.uci.edu/mlearn/MLRepository.html
2. J. M. Park, Convergence and Application of Online Active Sampling Using Orthogonal Pillar Vectors", IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 26(9), pp.1197-1207, 2004,.
3. L. Jiao, J. Liu and Z. Weicai, An Organizational Coevolutionary Algorithm for Classification, IEEE Trans. on Evolutionary Computation, Vol. 10 (1), pp. 67-79, 2006
4. E. Keogh, M. Pazzani, Learning Augmented Bayesian Classifiers: A Comparison of Distribution-based and Classification-based Approaches, 7th. Int. Workshop on AI and Statistics, pp. 225-230, Lauderdale, Florida,1999
5. C. Eick, N. Zeidat, and R. Vilalta, Using Representative-Based Clustering for Nearest Neighbor Dataset Editing, in Proc. Fourth IEEE International Conference on Data Mining (ICDM), Brighton, England, pp. 375-378, November 2004.
6. N. G. Pedrajas, C. H. Martinez and D. O. Boyer, Cooperative Coevolution of Artificial Neural Network Ensembles for Pattern Recognition, IEEE Transaction on Evolutionary Computation, 9(3):271–302, 2005.
7. Y. Sai; P. Nie, R. Xu, J. Huang, A Rough Set Approach to Mining Concise Rules from Inconsistent Data, IEEE International Conference on Granular Computing, pp.333–336, 2006
8. V. Paredes, E. Vidal, A Class-Dependent Weighted Dissimilarity Measure for Nearest Neighbor Classification Problems, Pattern Recognition Letters , Vol. 21, pp. 1027-1036, 2000.
9. H. S.Wong, M. Wu, R. A. Joyce, L. Guan, S. Y. Kung, A Neural Network Approach for Predicting Network Resource Requirements in Video Transmission Systems, Proc. Of IEEE Pacific RIM Conf On Multimedia, 2000.
10. D.Specht, Probabilistic Neural Networks for Classification, Mapping, or Associative Memory, IEEE Int. Conf. On Neural Networks, pp. 525-532, July 1988.
11. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, John Wiley & Sons, 2001.
12. A. M. Martinez, A. C. Kak, PCA Versus LDA, IEEE Trans. on Pattern Analysis and Machine Intelligence, , Vol. 23, No. 2, pp.228-233, 2001
13. C. M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1997
14. B. Scholkopf, A. Smola, and K. R. Muller, Nonlinear Component Analysis as a Kernel Eigenvalue Problem", Neural Computation,Vol.10, pp.1299-1319, 1998.
15. K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, B. Scholkof, An Introduction to Kernel-Based Learning Algorithms, IEEE Transactions on Neural Networks, Vol 12,No 2,March 2001
16. A. Kocsor, L. Tóth, Kernel-Based Feature Extraction with a Speech Technology Application, IEEE Transaction on Signal Processing, Vol. 52, No. 8, pp. 2250-2263, 2004.
17. L. Shen and L. Bai, Gabor Feature Based Face Recognition Using Kernel Methods, Proceedings of 6th IEEE International Conference on Automatic Face and Gesture Recognition(FGR '04), pp. 170–176, Seoul, South Korea, May 2004.
18. M. Kantardzic, Data Mining, Wiley Intersicence, 2003