ISSN 1392 – 1215 ELECTRONICS AND ELECTRICAL ENGINEERING 2009. No. 6(94)
ELEKTRONIKA IR ELEKTROTECHNIKA

T180 TELECOMMUNICATIONS ENGINEERING
TELEKOMUNIKACIJŲ INŽINERIJA

Matrix-based Neural Network with Linear Nodes

P. Daniušis[1,2], P.Vaitkus[1]
[1] *Vilnius University,*
Naugarduko 24, LT-03225 Vilnius, Lithuania, e-mail: povilas.daniusis@mif.vu.lt,
[2] *Vilnius Management Academy,*
J. Basanavičiaus g. 29a, LT-03109 Vilnius, Lithuania, e-mail: vaitkuspranas@gmail.com

Introduction

Many real-world machine learning tasks are either regression or classification problems. In standard regression or classification models the inputs usually are represented as vectors. However, in some practical problems matrix or tensor-based inputs arises naturally (for example in image, video stream, multidimensional time series or textual data analysis, bioinformatics and other fields). In such cases, the vector-based representation does not consider an inner structure of the inputs, which can provide useful information. For example, if the inputs are images (i.e. m×n matrices), representing an input as m·n dimensional vector will delete an information about the structure of the original image. Moreover, the dimension of such vectors can be very high, and consequently large training set is required to efficiently estimate the parameters of the model. Computer experiments [1,2,3] shows, that even when initial inputs are vectors, it can be useful to represent them as matrices (or higher order tensors) and apply matrix/tensor-based models. This approach can be especially useful in the case of small training sample [1,3], since matrix (or tensor) - based models usually have less parameters.

Recently, various machine learning techniques, involving matrix or tensor representation of the inputs received much attention in the literature (e.g. linear models [1,9], non-linear models [2,3], probabilistic techniques [5], tensor principal component analysis [8], tensor discriminant analysis [6], Tucker decomposition [7], and correlation tensor analysis [4]).

In this article we propose a new linear model for matrix-based regression or classification and apply it to some standard benchmark data sets. The computer experiments reveals that in the case of small training sample the proposed linear model can be more efficient than the standard linear regression and Cai's model [1].

The model

In this section we shortly describe a standard linear regression and introduce a new linear matrix-based model.

Let $y=[y_1,y_2,...,y_M]^T$ be a vector of the desired responses and $X=[x_1,x_2,...,x_M]^T$ be an observation matrix (where $x_i=[1,x_{i1},...,x_{im}]^T\in R^{m+1}$, $y_i\in R$ **).** In the standard linear regression we seek a m+1 dimensional vector α , which minimizes the regularized euclidean norm

$$J=\left\|y-X\alpha\right\|^2+\lambda\,\|\,\alpha\,\|^2, \qquad (1)$$

where $\lambda\geq 0$ a regularization constant.

Provided, that the inverse of $X^TX+\lambda I$ exist, the minimizer of (1) is defined by

$$\alpha=(X^TX+\lambda I)^{-1}X^Ty \qquad (2)$$

and the model is

$$y(x\,|\,\alpha)=x^T\alpha, \qquad (3)$$

where $x=[1,x_1,...,x_m]^T\in R^{m+1}$ – an input vector; I – an identity matrix. In [1] proposed a linear matrix-based model:

$$y(X\,|\,\theta)=u^TXv+b, \qquad (4)$$

where X – m×n-dimensional input matrix; u – m-dimensional parameter vector; v – n-dimensional parameter vector; b – bias (by θ={u,v,b} we denote all parameters of the model). When the inputs are matrices, (4) model often is more efficient than standard linear regression [1], because it has significantly less parameters (if X is m×n matrix (4) model has $m+n+1$ parameters, while standard linear regression (3) has even $mn+1$) and exploits an inner structure of the input matrix. Smaller number of the parameters also is useful when the training set is small.

In [2] we generalized (4) model to the non-linear version using the multilayer perceptron (MLP) neural network framework:

$$y(X\,|\,\theta)=\sum_{i=1}^{r}\alpha_i\sigma(u_i^TXv_i+b_i), \qquad (5)$$

where X is an m×n-dimensional input matrix; r denotes the number of neurones; $\theta=\{b,\{\alpha_i u_i, v_i\}_{i=1}^r\}$ denotes all parameters of the model; σ – a non-linear activation function (for example, hyperbolic tangent $\sigma_{TH}(t)=\frac{e^t-e^{-t}}{e^t+e^{-t}}$) or logistic sigmoid $\sigma_{LS}(t)=\frac{1}{1+e^{-t}}$ functions).

In this article we will investigate a linear variant of the (5) model:

$$y_{MNNL(r)}(X\mid\theta)=\sum_{i=1}^{r}u_i^T X v_i+b, \qquad (6)$$

where $\theta=\{b,\{u_i,v_i\}_{i=1}^r\}$ (all parameters of the model). We will call model (6) matrix-based neural network with linear nodes.

Denote the training set of M observatios by $T=\{X_i,y_i\}_{i=1}^M$ where X_i - m×n matrices (inputs), y_i – scalars (outputs). When $r<\frac{mn}{m+n}$ model (6) still have less parameters than standard linear regression and possibly can be used more efficiently than (3) when the training set is small. In our opinion (6) model can be more efficient than (4) and standard linear regression, when the size of the training set is too small to efficiently estimate the parameters of full linear regression, but sufficiently large to use more complex than (4) model, since (6) model provides more freedom than (4). (6) can be considered as an intermediate model between models (4) and (3) (full linear model).

In the following we will derive an algorithm for minimizing the regularized sum squared error (RSSE), defined by:

$$E=\sum_{(X,y)\in T}(y_{MNNL(r)}(X\mid\theta)-y)^2+\lambda(b^2+\sum_{k=1}^{r}u_k^T u_k+v_k^T v_k), \qquad (7)$$

where λ≥0 – a regularization constant, fixed by the user.

Note, that (6) model is not equivalent to (4). Because the (6) model is linear we do not need a gradient-based methods for parameter optimization, since we can easily compute the derivatives of (7) and solve them. However, each optimal parameter depend on other parameters of the model (6). Therefore an iterative procedure must be applied. Define

$$a_{k,X}=\sum_{j\neq k}u_j^T X v_j. \qquad (8)$$

By setting derivatives of (7) with respect to each parameter u_k, v_k and b to zero we derive the following iterative algorithm for training the model (6):

Algorithm 1

1. Fix the number of neurones of the model (6) $1\le r<\frac{mn}{m+n}$, arbitrarily chose initial parameters $u_k\in R^m, v_k\in R^n$ and $b\in R$, k=1,2,...,r, desired learning error $\varepsilon\ge 0$, regularization constant λ≥0, set an iteration number $t=0$ and fix maximal iteration number T.

2. For each k-th regressor calculate new parameters:

$$u_k=(\sum_{i=1}^{M}X_i v_k v_k^T X_i^T+\lambda I)^{-1}\sum_{j=1}^{M}(y_j-a_{k,X_j})X_j v_k, \qquad (9)$$

$$v_k=(\sum_{i=1}^{M}X_i^T u_k u_k^T X_i+\lambda I)^{-1}\sum_{j=1}^{M}(y_j-a_{k,X_j})X_j^T u_k, \qquad (10)$$

$$b=\frac{1}{M+\lambda}\sum_{j=1}^{M}(y_j-\sum_{i=1}^{r}u_i^T X_j v_i), \qquad (11)$$

where M – size of the training sample; I – an identity matrix; the coefficients a_{k,X_j} – defined by (8).

3. Set n := n + 1.

4. Repeat step 2.) until $RSSE\le\varepsilon$ or iteration number $t>T$.

Similarly as in [1] we can prove the convergence of the **Algorithm 1**.

PROPOSITION. For any λ≥0, the sequence of RSSE values, defined by the **Algorithm 1** converges.

Proof. At every step of **Algorithm 1** we solve a standard regularized least squares problem, thus the value of (7) function non-increases. Obviously, (7) is bounded by 0. Since any monotonous and bounded sequence converges, the result follows.

Define an inner product between two matrices A and B as

$$<A,B>=\sum_{ij}A_{ij}B_{ij}. \qquad (12)$$

The following proposition shows how acurately any linear model can be approximated by (6) model.

PROPOSITION. Let $X\in R^{m\times n}$. Then for any matrix $W\in R^{m\times n}$

$$\min_{\theta}|<X,W>-y_{MNNL(r)}(X\mid\theta)|\le\|X\|^2\sum_{i=r+1}^{n}\sigma_i^2, \qquad (13)$$

where σ_i – i-th singular value of W.

Proof. Without loss of generality we assume, that min(m,n)=n. Let $W=U\Sigma V^T$ – a singular value decomposition of the matrix W, where $U=[u_1,...,u_n]$, $V=[v_1,...,v_n]$, and Σ – diagonal matrix with singular values σ_i in the diagonal. By well known Eckart-Young theorem best rank-r approximation of the matrix W satisfies

$$\min_{\hat{W}:rank(\hat{W})=r}\|W-\hat{W}\|^2=\sum_{i=r+1}^{n}\sigma_i^2, \qquad (14)$$

and minimum is achieved at $\hat{W}=\sum_{i=1}^{r}\sigma_i u_i v_i^T$. On the other side, we can write model (6) as

$$y_{MNNL(r)}(X \mid \theta) = \sum_{i=1}^{r} \sigma_i u_i^T X v_i = < X, \hat{W} > . \qquad (15)$$

Thus, (6) is a linear model with special weights $\hat{W}$. Therefore by (14) left side of (13) is equal to

$$| < X, W > - < X, \hat{W} > | \le \|X\|^2 \sum_{i=r+1}^{n} \sigma_i^2 . \qquad (16)$$

Computer experiments

In this section we will empirically compare the proposed model (6) with (4) model and full linear regression model (3). In the computer experiments we will use the standard benchmark data sets from UCI machine learning repository. All data sets used in our experiments are binary classification problems. To demonstrate the effects of the matrix-based models we will consider small training sets. For convenience, the size of the training set is equal to the dimensionality of the input vector. The training data (inputs) was standardized by subtracting the mean and dividing by the standard deviation. In the cases of (4) or (6) matrix-based models, the input vectors were transformed into the matrices (Matrix column of Table 1). The measure of performance of the model is the correct classification probability over the testing set. In each experiment the training set of fixed size was selected randomly, all experiments were performed 100 times, and averaged results presented in the Table 1. To test the statistical significance of the results the signed rank test for zero median between the differences of the performances of the models was applied (see Table 2). The models (4) and (6) were trained according to the **Algorithm 1** for T=10 iterations. The regularization constant λ=0.01 was fixed for all data sets and all models.

Table 1. Correct classification probabilities over the testing set. r=1 – (4) (Cai's) model, r=2 – (6) model, Full – full linear model (3), Dim – dimensionality of the input vectors and Matrix – size of input matrices for (4) and (6) models

Dataset	r=1	r=2	Full	Dim	Matrix
Ionosphere	0.77	**0.79**	0.75	33	11x3
Breast cancer	0.90	0.88	0.89	10	5x2
Sonar	0.67	**0.71**	0.58	60	10x6
Specft	0.70	**0.72**	0.56	44	11x4
Australian	0.73	0.70	0.65	14	7x2
Musk	0.73	0.71	0.67	166	83x2
German	0.60	0.58	0.55	24	6x4

The results of the Table 1 are statistically significant with p-values in the Table 2 (small p-values indicates statistical significance).

Table 2. P-values of the signed rank test for zero median between the differences of the performances of the models

Dataset	r=1/r=2	r=1/Full	r=2/Full
Ionosphere	0.01	$\sim 10^{-4}$	$\sim 10^{-8}$
Breastcancer	0.04	0.11	0.71
Sonar	$\sim 10^{-10}$	$\sim 10^{-14}$	$\sim 10^{-18}$
Specft	0.01	$\sim 10^{-16}$	$\sim 10^{-17}$
Australian	0.008	$\sim 10^{-8}$	$\sim 10^{-5}$
Musk	$\sim 10^{-4}$	$\sim 10^{-16}$	$\sim 10^{-13}$
German	$\sim 10^{-4}$	$\sim 10^{-9}$	$\sim 10^{-5}$

Conclusions

According to the experimental results we can conclude, that in the case of small training sample, (6) model can be more efficient than (4) or full linear model. In most cases matrix-based models (r=1 and r=2 columns of Table 1) outperformed full linear model. In our opinion, this is because (4) or (6) models have less parameters than (3) and with the same amount of training data they are estimated more efficiently. Moreover, since the (6) is a linear model with structured parameters, it can be interpreted as a form of additional regularization. From the perspective of Vapnik-Chervonenkis (VC) theory [10], one can check, that the VC dimension of full linear model with m·n variables is equal to h_{Linear}=m·n+1, while that of model (6) with order r is equal to h_{LMNN}=r·max(m,n)+1. It is well known [10] that for any binary-valued decision function set S the following inequality holds with probability 1-η

$$E_{Test} \le E_{Training} + \sqrt{\frac{h(1+\log\frac{2N}{h}) - \log\frac{\eta}{4}}{N}} , \qquad (17)$$

where h is Vapnik-Chervonenkis (VC) dimension [10] of S, N is the size of training set, and E_{Test} and $E_{Training}$ are respectively testing and training errors. Thus, the confidence term in (17) of (6) model does not exceed that of the full linear model.

Table 2 shows that most of the results (except the breast cancer case) are statisticaly significant.

The proposed model can be easily extended to the higher order tensor case. An interesting questions, left to the future work, is an efficient estimation of the optimal model order r and the size of the input matrix or tensor.

References

1. **Deng Cai, He X., Han J.** Learning with Tensor Representation. – 2006.
2. **Daniušis P., Vaitkus P.** Neural network with matrix inputs // – Informatica, 2008. – Vol. 19. – Issue 4. – P. 477–468.
3. **Daniušis P., Vaitkus P.** Kernel regression on matrix patterns // Lith. Math. Journal. (spec. edition; ISSN 0132–2818). – Vol. 48-49, – P. 191-195, – 2008
4. **Fu Y., Huang T. S.** Image Classification Using Correlation Tensor Analysis // IEEE transactions on images processing. – Vol. 17. – No. 2. – 2008.
5. **Tao D., Sun J., Shen J., Wu X.** Bayesian Tensor Analysis // IEEE International Joint Conference on Neural Networks (IJCNN). – 2008.
6. **Tao D., Li X,. Wu X., Maybank J. S.** General Tensor Discriminant analysis and Gabor Features for Gait Recognition // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007. – Vol. 29. – No. 10. – P. 1700–1715.
7. **Tucker L. R.** Some Mathematical Notes on Three–mode Factor Analysis // Psychometrika, 1996. – Vol. 31. – No. 3.
8. **Vasilescu M. A. O., Terzopoulos D.** Multilinear Subspace Analysis on Image Ensembles. // IEEE proc. Int'l Conf. On Computer Vision and Pattern Recognition, 2003. – Vol. 2. P. 93–99.
9. **Zhe W., Songcan C.** New Least Squares Support Vector Machines Based on Matrix Patterns. // Neural processing letters, 2007. – P. 41–56.

10. **V.N. Vapnik**. The Nature of Statistical Learning Theory, Springer-Verlag New York, Inc., New York, 1995.

Received 2009 02 13

P. Daniušis, P. Vaitkus. Matrix-based Neural Network with Linear Nodes // Electronics and Electrical Engineering. – Kaunas: Technologija, 2009. – No. 6(94). – P. 39–42.

In this article we propose a new linear model for regression/classification of matrix input data. The algorithm for parameter estimation is constructed, some properties of the model are analyzed. The proposed model was applied for various binary classification problems, experimentally demonstrated, that in the case of small training sample, this model can be more efficient than standard techniques. In each experiment the training sample was selected randomly, the results (correct classification probabilities on the testing set) were averaged, statistical hypothesis about efficiency of the models were tested. By signed rank test most of the results are statistically significant. Bibl. 9 (in English; summaries in English, Russian and Lithuanian).

П. Данюшис, П. Вайткус. Линейная нейронная сеть для матриц // Электроника и электротехника. – Каунас: Технология, 2009. – № 6(94). – С. 39–42.

Предложена новая линейная модель для регрессии/классификации матричных данных. Построен алгоритм для оценки параметров, проанализированы некоторые свойства модели. Предложенная модель была применена для некоторых бинарных проблем классификации. Экспериментально продемонстрировано, что в случае малой выборки обучения предложенная модель может быть более эффективной, чем стандартные методы. В каждом опыте выборка обучения была выбрана случайно, результаты (вероятности правильной классификации на выборке тестирования) усреднены. Проверена статистическая гипотеза об эффективности моделей. По ранговому критерию знаков большинство результатов являются статистически значимыми. Библ. 9 (на английском языке; рефераты на английском, русском и литовском яз.).

P. Daniušis, P. Vaitkus. Tiesinis neuroninis matricų tinklas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2009. – Nr. 6(94). – P. 39–42.

Pasiūlytas naujas tiesinis regresijos bei klasifikavimo modelis, kai įėjimai yra matricos. Sukonstruotas iteracinis algoritmas šio modelio parametrams įvertinti, išnagrinėtos kai kurios modelio savybės. Pasiūlytas modelis pritaikytas įvairiems dviejų klasių klasifikavimo uždaviniams, eksperimentiškai parodyta, jog esant mažai mokymo imčiai jis gali būti efektyvesnis už žinomus analogiškus modelius. Kiekvieno eksperimento metu mokymo imtis buvo išrenkama atsitiktinai, gauti rezultatai (teisingo klasifikavimo tikimybės su testiniais duomenimis) suvidurkinti, patikrintos statistinės modelių efektyvumo hipotezės. Remiantis ženklų kriterijumi nustatyta gauta, jog daugeliui atvejų rezultatai yra statistiškai reikšmingi. Bibl. 9 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).