维普资讯 http://www.cqvip.com

一种全局收敛的PCA神经网络学习算法*)

黄克军　叶　茂　王雁东　李毅超
(电子科技大学计算机科学与工程学院　成都610054)

摘　要　主元分析(PCA)也称为K-L变换是进行特征提取的一种重要方法。近年来，为了处理海量数据，许多基于Hebbian学习算法的PCA神经网络被提出来。传统的算法，通常不能保证其收敛性或者收敛速度较慢。基于CRLS神经网络，本文提出了一种新的确保权向量收敛的学习算法，本算法无须在计算中规格化权向量。同时也证明了该学习算法使得权向量收敛到最大特征值所对应的特征向量。实验表明，与传统的CRLS神经网络比较，本文算法准确性得到极大提高。

关键词　主元分析，神经网络，特征向量，特征提取

A Convergent Algorithm for PCA Neural Network

HUANG Ke-Jun　YE Mao　WANG Yan-Dong　LI Yi-Chao
(School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054)

Abstract　Principal component analysis (PCA) is one of the most general-purpose feature extraction methods. For processing the huge data sets, a variety of learning algorithm for PCA has been proposed. However, traditional algorithms will either divergence or convergence very slowly. Based on the CRLS neural network, a novel convergence algorithm is proposed and the fact that the weight vector will converge to the largest eigenvector is also proved. Finally, simulation results are also included to illustrate the accuracy of this new algorithm.

Keywords　Principal component analysis, Neural network, Eigenvector, Feature extraction

在模式识别领域中，如何从复杂、高维的输入数据流中提取主要特征是一个非常重要的问题。通过特征提取，能从输入数据流中去掉多余的信息，并且可以将数据在低维空间进行处理。主元分析是其中一个特征提取方法，经典方法是QR分解或者SVD分解。为了处理海量数据，许多基于Hebbian学习算法的特征提取神经网络被提出来，如文[1～7,12]。通过神经网络可以帮助找到主要的特征方向而无需计算协方差矩阵。在比较现有算法的实验中，Costa，Fiori[8]说明串行CRLS神经网络[4]效果最好。CRLS算法是基于Hebbian学习算法的串行提取最大特征向量的神经网络，如图1所示。

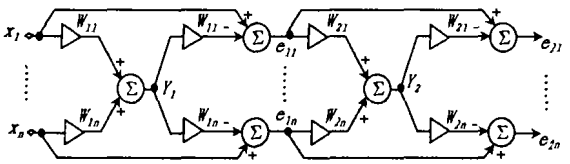

图1　CRLS神经网络模型

尽管CRLS神经网络可以较好地提取特征向量，但是该神经网络的权向量学习算法并非全局收敛。Anisse和Giansalvo在文[10]中证明了基于Hebbian学习算法的某些神经网络实际上并不收敛。以往的研究大都通过将学习算法变为常微分方程来证明其收敛性，Zufiria在文[11]中说明了这二者实际上通常不等价，并且举出了许多例子说明其不收敛性。

基于CRLS神经网络，本文提出新的权向量学习算法，该算法不需要规格化权向量，而且能够同步计算特征值、特征向量并保证其收敛性，所以准确性能得到很大提升。由此，文中提出的CRLS神经网络的缺点得到了较满意的解决。本文首先介绍一些数学背景及算法描述，其次对算法进行理论分析，证明其收敛性，最后通过实验揭示本算法的优越性。

1　算法描述

设$x(t)$是零均值的随机向量，$x(t)=[x_1(t),x_2(t),\cdots,x_n(t)]^T\in R^n$。PCA的目标是在$n$维空间中找到$m$个正交的向量$w=[w_1,w_2,\cdots,w_m]\in R^{n\times m}(m\leqslant n)$，而由这些向量所组成的线性空间能够最大化地表示该$n$维数据集。将$x$投影到$w$空间，令$y=w^Tx$，由$y$构造$n$维空间向量：

$$\hat{x}=\sum_{i=1}^{m}y_iw_i=\sum_{i=1}^{m}(w_i^Tx)w_i \tag{1}$$

为使$E\{\|x-\hat{x}\|^2\}$最小，$w_1,w_2,\cdots,w_m$必须为关联矩阵$E[xx^T]$前m个特征值所对应的特征向量。由矩阵理论，$E[xx^T]$所有特征值大于等于0。$y(t)=[y_1(t),y_2(t),\cdots,y_m(t)]^T\in R^m$称为主元。传统的PCA方法是计算关联矩阵的特征向量，但是对于海量数据，关联矩阵将会变得非常庞大，以至于计算机无法处理，所以必须采用全新手段在线地处理海量数据。

假设有n维向量$x(k),k=1,\cdots,N$，w为神经元的权向量。如图1所示，神经网络逐一提取特征向量。在第一层神经元，输入数据为$x(t)$，通过学习算法，权向量将会收敛于关联矩阵最大特征值所对应的特征向量。在第二层神经元，权向量也会收敛于相应输入数据关联矩阵的最大特征向量。此时为计算次大的特征向量，输入数据不再是$x(t)$，而是$x(t)$减掉最大特征向量方向分量。依此类推，可以计算出任意多个特征

*)电子科技大学青年基金资助。黄克军　学士，讲师，主要研究领域为神经网络，模式识别，计算机网络及信息安全。叶　茂　博士，讲师，主要研究领域为神经网络，模式识别，智能计算。王雁东　硕士，讲师，主要研究领域为神经网络，模式识别，计算机网络及信息安全。李毅超　硕士，副教授，主要研究领域为神经网络，模式识别，计算机网络及信息安全。

维普资讯 http://www.cqvip.com

向量。算法如图2所示。

现简要说明算法，在第(1)步，通过计算 Energy 的值，可以得到 $E[xx^T]$矩阵所有特征值之和。其理论根据是矩阵特征值之和等于矩阵的迹。利用 Energy 的值，可以帮助确定实际需要提取特征向量的数目。步(3)(5)是文[3]提出来的，使得学习速度是自适应的并且可以进行最优的训练。通过自适应选择学习速率，算法在收敛速度上有较大的提高。关于第(5)步的初始值由下面的算法确定：

$$\eta_j(0)=\sigma[e_{j-1}]/N=\eta_{j-1}(0)-E[y_{j-1}^2] \quad (2)$$

第(8)与第(9)步由文[4]提出来，称之为缩小(deflation)过程，它使得特征值与特征向量可以被依次序稳定地计算出来。每一层神经元计算相应关联矩阵的最大特征向量，避免了传统算法需要向量正交化而产生的误差。

本文的主要创新之处是第(6)与(7)步，它们迭代地计算特征值与特征向量。与其它算法不同，通过单位化权向量，该算法能有力地遏止权向量发散，同时我们还能够证明该算法的收敛性。算法第(6)步用于确保权向量收敛到正确的特征向量。

FOR x(k): FROM k=1 TO N

(1)set error $e_0(k)=x(k)$, $Energy=Energy+\sum_{i=1}^{n}x_i(k)^2$

Energy=Energy/N
FOR every neuron: FROM j=1 TO m
(2)Initialize $w_j(0)=1$
(3)set $\eta_j(0)=\sigma[e_{j-1}]/N$, $\sigma[e]=\sum_i\sum_k e_{ik}^2$, $E(0)=0$
FOR epoch index s: FROM s=1 TO MAX_S
FOR input data: FROM k=1 TO N
(4)$y_j(k)=w_j(k)^t e_{j-1}(k)$
(5)$\eta_j(k)=y_j(k)^2+\eta_j(k-1)$, $E(k)=E(k-1)+y^2/(w_j(k)^T w_j(k))$
(6)$\lambda_j(k)=\lambda_j(k-1)+[y_j(k)^2/\eta_j(k)][E(k)/k-\lambda_j(k-1)]$
(7)$w_j(k+1)=w_j(k)+[y_j(k)^2/\eta_j(k)][y_j(k)e_{j-1}(k)-\lambda_j(k)w_j(k)]$,
IF $|w_j(k+1)-w_j(k)|<\varepsilon$, $|\lambda_j(k+1)-\lambda_j(k)|<\varepsilon$;
$w_j=w_j(k)/\|w_j(k)\|$; $\lambda_j=\lambda_j(k)$, $\lambda=\lambda+\lambda_j$; GO TO STABLE
STABLE:
IF $\lambda/Energy>90\%$; STOP
FOR input data: FROM k=1 TO N
(8)set $y_j(k)=w_j^t e_{j-1}(k)$
(9)set $e_j(k)=e_{j-1}(k)-y_j(k)w_j$

图2 新 CRLS 算法

在算法中，如果能够较好地选择每一层神经元的初始权向量，本算法收敛速度能得到进一步提高，请参见文[9]。如果初始向量与已计算出来的特征向量正交，收敛速度将会得到提高。原因如下：从初始权向量出发，算法经过有限步迭代，权向量将最终进入与这些特征向量正交的空间。如果提前设置好初始向量，将节约这些迭代过程。以初始化 $w_j(0)$为例，设 λ_j, v_j 分别为 $E[xx^T]$按特征值大小排列第 j 的特征值与特征向量。首先随机产生初始权向量 $w_j(0)$，设 $\tilde{v}_j=[v_1, v_2, \cdots, v_{j-1}]$，最后令 $w_j(0)=w_j(0)-\tilde{v}_j\tilde{v}_j^t w_j(0)$。

2 理论分析

本节都假设 $E[xx^T]$有不同的特征值，有相同特征值情况将在稍后说明。由于每一层神经元权向量学习算法一致，因此如果能够证明第一层算法中的权向量确实收敛到最大的特征向量，即说明了任意的 j，w_j 收敛到所对应的特征向量$\pm v_j$。为简单起见，令 $\eta(k)$表示学习速率，第一层神经元学习算法化简为如下随机差分方程组：

$$\lambda_1(k)=\lambda_1(k-1)+\eta(k)[E(k)/k-\lambda_1(k-1)]$$

$$w_1(k+1)=w_1(k)+\eta(k)[e_0(k)y_1(k)-\lambda_1(k)w_1(k)] \quad (3)$$

$$w_j=w_j(k)/\|w_j(k)\|, \text{当 } w_j(k)\text{收敛} \quad (4)$$

由于 $w_1(k)$，$\lambda_1(k)$是随机变量，对方程组(3)取条件期望 $E\{w_1(k+1)|w_1(0), e_0(i), i<k\}$，并用 $w(k)$，$\lambda(k)$分别表示 $w_1(k)$与 $\lambda_1(k)$的均值。又因为 $w_1(0), e_0(i)=x(i), i<k$ 相互独立，令 $R=E[x(k)x(k)^t]$，(3)式改写为如下差分方程组：

$$\lambda(k)=\lambda(k-1)+\eta(k)[\frac{1}{k}\sum_{i=1}^{k}\frac{w^t(i)Rw(i)}{w^t(i)w(i)}-\lambda(k-1)] \quad (5)$$

$$w(k+1)=w(k)+\eta(k)[Rw(k)-\lambda(k)w(k)] \quad (6)$$

证明分为几个部分。首先证明 $w(k)$与 v_1夹角将会收敛到0，然后由(4)式可知 $w(k)$收敛到 v_1，其次证明 $\lambda(k)$收敛到 λ_1。

$\eta(k)$为学习速率。如果算法收敛，$\eta(k)$值越大，收敛的速度也越快，但其取值范围有一定的限制。定理1给出了在实际计算中对 $\eta(k)$取值的限制。

定理1 如果 $\eta(k)\geqslant\min(2, \min_{j=1,\cdots,n}\frac{2}{\lambda_1-\lambda_j})$，$w(k)$，$\lambda(k)$将不收敛。

证明：由(5)式有

$$\lambda(k)=(1-\eta(k))\lambda(k-1)+\eta(k)\frac{1}{k}\sum_{i=1}^{k}\frac{w^t(i)Rw(i)}{w^t(i)w(i)}$$

如果 $\lambda(k)$收敛，$\eta(k)$必须满足不等式$|1-\eta(k)|\leqslant 1$。即如果 $\eta(k)\geqslant 2$，$\lambda(k)$不收敛。

另一方面，令 $w(k)=\sum_{j=1}^{n}\varepsilon_j(k)v_j$，代入(6)式得

$$\varepsilon_j(k+1)=\varepsilon_j(k)+\eta(k)(\lambda_j-\lambda(k))\varepsilon_j(k), j=1,\cdots,n \quad (7)$$

如果 $\varepsilon_j(k)$收敛，η 必须满足不等式$|1+\eta(k)(\lambda_j-\lambda(k))|\leqslant 1$。如果 $\lambda(k)$收敛到 λ_1，那么有如下不等式成立 $\eta(k)(\lambda_j-\lambda_1)\geqslant -2, j=2,\cdots,n$，所以 $\eta(k)\leqslant\min_{j=1,\cdots,n}\frac{2}{\lambda_1-\lambda_j}$。证毕。

定理2 如果 $\eta(k)\lambda_1\leqslant\frac{1}{2}$，$w(0)^T v_1\neq 0$，$w(k)$与 v_1的夹角趋于零。

证明：因为对任意 k，都有 $w^t(k)Rw(k)\leqslant\lambda_1$，设 $\hat{\lambda}(k)$为下列方程的解

$$\hat{\lambda}(k)=\hat{\lambda}(k-1)+\eta(k)[\lambda_1-\hat{\lambda}(k-1)] \quad (8)$$

因为方程(8)的解 $\hat{\lambda}(k)$会收敛到 λ_1，所以任意 ε，存在 k_1，当 $k>k_1$时，使得 $\hat{\lambda}(k)<\lambda_1+\varepsilon$。由比较定理，有下列不等式成立：

$$\lambda(k)\leqslant\hat{\lambda}(k)<\lambda_1+\varepsilon \quad (9)$$

另一方面，令 $w(k)=L(k)v_1+\sum_{j=2}^{n}\varepsilon_j(k)v_j$，由定理条件知，$L(0)\neq 0$，所以 $L(k)\neq 0$。由(6)式得

$$L(k+1)=[1+\eta(k)(\lambda_1-\lambda(k))]L(k)$$

$$\varepsilon_j(k+1)=[1+\eta(k)(\lambda_j-\lambda(k))]\varepsilon_j(k), j>1 \quad (10)$$

令 $Ang(w(k))$为 $w(k)$与 v_1的夹角，$\tan^2(Ang(w(k)))=\sum_{j=2}^{n}\varepsilon_j^2/L^2$。要证 $k>k_2>k_1$时，$Ang(w(k+1))$小于 $Ang(w(k))$。只需说明 $\tan^2(Ang(w(k+1)))<\tan^2(Ang(w(k)))$。即是：

$$\frac{\varepsilon_j(k+1)^2}{L(k+1)^2}=\frac{[1+\eta(k)(\lambda_j-\lambda(k))]^2}{[1+\eta(k)(\lambda_1-\lambda(k))]^2}\cdot\frac{\varepsilon_j(k)^2}{L(k)^2}<\frac{\varepsilon_j(k)^2}{L(k)^2}, j\geqslant 2 \quad (11)$$

由于(9)式

$$1+\eta(k)(\lambda_j-\lambda(k))>1-\eta(k)[\lambda_1+\varepsilon-\lambda_j]>0, j\geqslant 1$$

由定理条件 $\eta(k)\lambda_1\leqslant\frac{1}{2}$，我们有

$$1+\eta(k)(\lambda_j-\lambda(k))>\frac{1}{2}+\eta(k)(\lambda_j-\varepsilon), j=1,\cdots,n$$

取 $\varepsilon\eta(k)<\frac{1}{2}$，当 $k>k_2$时，$1+\eta(k)(\lambda_j-\lambda(k))\geqslant 0$。所以当 $k>k_2$，

维普资讯 http://www.cqvip.com

$[1+\eta(k)(\lambda_j-\lambda(k))]^2/[1+\eta(k)(\lambda_1-\lambda(k))]^2<1, j\geqslant 2$ (11)式成立。证毕。

定理3 在定理1与定理2的条件下，$\lambda(k)\to\lambda_1$。

证明：因为 $w^t(k)Rw(k)\leqslant\lambda_1$，$\hat{\lambda}(k)$为下列方程的解：

$\hat{\lambda}(k)=\hat{\lambda}(k-1)+\eta(k)[\lambda_1-\hat{\lambda}(k-1)]$

该方程的解 $\hat{\lambda}(k)$收敛到 λ_1。

另一方面，由定理2知，$\frac{w^T(k)Rw(k)}{w^T(k)w(k)}\to\lambda_1$. $\forall\varepsilon$，$\exists k_1$，使得当 $k>k_1$，$\frac{w^T(k)Rw(k)}{w^T(k)w(k)}>\lambda_1-\varepsilon$。设 $\hat{\lambda}(k)$为下列方程的解：

$$\hat{\lambda}(k)=\hat{\lambda}(k-1)+\eta(k)[\frac{1}{k}\sum_{i=1}^{k_1}\frac{w^t(i)Rw(i)}{w^t(i)w(i)}+\frac{k-k_1}{k}(\lambda_1-\varepsilon)-\hat{\lambda}(k-1)] \quad (12)$$

因为方程(12)的解 $\hat{\lambda}(k)$收敛到 $\lambda_1-\varepsilon$，由比较定理，当 $k>k_1$时，$\hat{\lambda}(k)\leqslant\lambda(k)\leqslant\hat{\lambda}(k)$。令 $k\to\infty$，$\lambda_1-\varepsilon\leqslant\lim_{k\to\infty}\lambda(k)\leqslant\lambda_1$。对任意 ε 都成立。去掉 ε，证毕。

定理2，3需要假设关联矩阵 $E[xx^T]$有不同的特征值。实际上如果有相同的特征值，有类似定理2与定理3结论成立。假设最大特征值有 k 重，定理2将改为 $w(k)$与该 k 维特征向量空间夹角趋于零。证明方法与定理2类似。

3 实验结果

为了证明本文算法的有效性，进行了一系列的基于图像处理的实验。之所以选择 CRLS 算法，因为 CRLS 算法是目前效果最好的且本文算法建立在该算法基础之上，而其它效果较好的算法如0∞H[7]算法复杂度较高。本文实验平台为 Matlab5.0。

图3 实验图与压缩结果图

实验使用的512×512灰度图如图3中第一张图片所示。首先将图片分解为$\sqrt{n}\times\sqrt{n}$不相交的块集，并由这些块组成 n 维向量集。将像素点灰度值单位化使得均值为零。这些向量被随机地输入网络，通过提取特征向量，可以进行图像压缩。本文采用信号噪声比(SNR)进行图像压缩质量比较。令 I_{ij}，$\hat{I}_{ij}$分别为原图和压缩后重构图像像素值，有

$$SNR=10log_{10}\frac{\sum_{i=1}^{row}\sum_{j=1}^{col}I_{ij}^2}{\sum_{i=1}^{row}\sum_{j=1}^{col}(I_{ij}-\hat{I}_{ij})^2}.$$

在 n=16时，经过传统和新的 CRLS 网络压缩的图像如图3第2与3张图片所示。图4比较了传统和新的 CRLS 网络算法随着提取特征向量增加时 SNR 的变化情况。由图4可以看出传统的 CRLS 算法随着提取特征元的增加，错误也随之增加，新的算法则大幅度提高了准确率。

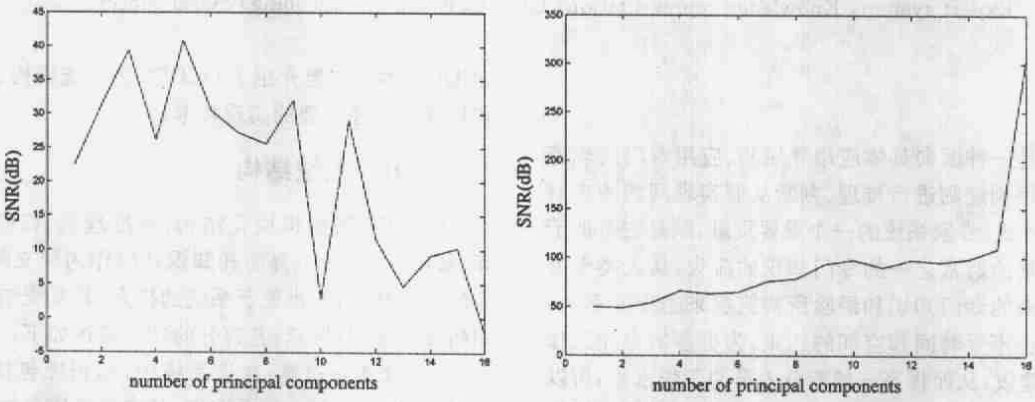

图4 传统 CRLS(左)与新 CRLS 算法(右)SNR 比较图

算法准确率之所以得到提高，在于本文算法确实保证权向量收敛到特征向量。从实验中可以看出，本文算法相比传统的 CRLS 算法准确度和效率有很大的提高。

结束语 本文基于 CRLS 神经网络提出了一种确保权向量收敛的学习算法。同时，也严格证明了本文算法的收敛性。实验结果表明，相比传统的 CRLS 算法，本文算法提取特征向量在质量和效率上都有很大的提高。

参考文献

1 Oja E. Neural networks, principal components, and subspaces. Int. J. Neural Syst. ,1989,1:61～68

2 Sanger T D. Optimal unsupervised learning in a single-layer linear feed forward neural network. Neural Networks,1989,2(6):459～473

3 Bannour S, Azimi-Sadjadi M R. Principal component extraction using recursive least squares learning. IEEE Trans. on Neural Networks ,1995,6:457～469

4 Cichocki A, Kasprzak W, Skarbek W. Adaptive learning algorithm for principal component analysis with partial data. Proc. Cybernetics Syst. ,1996,2:1014～1019

5 Chatterjee C, Kung Z , Roychowdhury V P. Algorithms for accelerated convergence of adaptive PCA. IEEE Trans. Neural Networks,2000 ,11(3):338～355

6 Firori S, Piazaa F. A general class of APEX PCA Neural algorithms. IEEE Trans. on Circuits and Systems Part I,2000,47(9):1394～1398

7 Marko V J. A new simple OH neuron model as biologically plausible principal component analyzer. IEEE Trans. on Neural Networks ,2003,14(4):853～859

8 Costa S, Fiori S. Image compression using principal component neural networks. Image and Vision Computing, 2001,19:649～668

9 Wong A S, Wong K W, Leung C S. A practical sequential method for principal component analysis. Neural Processing Letters , 2000,11:107～112

10 Anisse T, Giansalvo C. Against the convergence of the minor component analysis neurons. IEEE Trans. on Neural Networks, 1999,10(1):207～210

11 Pedro J Z. On the discrete time dynamics of the basic Hebbian neural network node. IEEE Trans. On Neural Networks,2002,13(6):1342～1352

12 章毅，王平安，周明天. 用神经网络计算矩阵特征值与特征向量. 计算机学报,2000,23(1):71～76