文章编号:1007-2985(2016)06-0026-03

# 基于不同误差函数的神经网络求解线性方程组*

汪思成[1],肖　林[1],严慧玲[2]

(1.吉首大学信息科学与工程学院,湖南 吉首 416000;2.吉首大学数学与统计学院,湖南 吉首 416000)

摘　要:对于同一线性方程,存在多种不同的求解方法,而且由于求解方法的不同,其收敛速度也存在差异.对一个一般的线性方程设计 2 个不同的误差函数,利用梯度下降法建立 2 个梯度神经网络模型.借助 Matlab 仿真软件进行计算机仿真,根据不同的梯度神经网络模型求出线性方程的解,从而证实 2 个梯度神经网络模型的可行性.最后借助 Matlab 软件模拟利用 2 个梯度神经网络模型求解线性方程时的收敛情况,比较 2 个梯度神经网络求解线性方程的收敛速度.

关键词:误差函数;梯度神经网络;求解线性方程组;Matlab 仿真

中图分类号:TP183;O151.1　　文献标志码:A　　DOI:10.3969/j.cnki.jdxb.2016.06.006

在自然科学与工程应用中,最小二乘法、样条函数插值、解非线性方程组、求解偏微分方程的差分法、有限元法、边界元法和目前工程实践中普遍存在的反演问题等,最终都转化为线性方程组的求解问题.[1]可见,线性方程组在理论研究和工程应用实践上有举足轻重的地位.近年来,人们也在这方面作了非常深入的研究,并取得许多有价值的成果.[2]但是,由于模型误差、测量误差、计算误差(舍入误差)等各种形式误差的存在,常常使获得的线性方程组中的系数矩阵和非齐次信息具有某种程度的近似性(即扰动性).这种近似性显然会使得线性方程组的求解不大容易得到真实的理论解[3-4].此时,不同的求解方法由于(迭代)运算机理不同,求解过程中误差积累程度也就不一样,因此必然会使得不同求解方法得到的解具有不同的逼近真解的误差程度,尤其是对那些具有病态性的方程组而言,许多现有的方法是失效的.因为它们不具有抗病态性,求得的解将完全失真.为了求解线性方程组,笔者设计了 2 个不同的误差函数,结合梯度神经的方法设计不同的梯度神经网络模型,并借助 Matlab 仿真软件进行建模、仿真和验证.

## 1　梯度神经网络设计

### 1.1 问题描述

对于一般线性方程 $AX=B$,$X\in \mathrm{R}$,设计如下 2 个表示形式:

$$AX=B\qquad X\in \mathrm{R},\tag{1}$$

$$X=A/B\qquad X\in \mathrm{R}.\tag{2}$$

针对这 2 个不同表现形式设计 2 个不同的误差函数,基于误差函数设计 2 个不同的梯度神经网络来实时求解 $X\in \mathrm{R}$,它能满足一般线性方程的要求.

* 收稿日期:2015-11-28

基金项目:国家自然科学基金青年项目(61503152);湖南省自然科学基金面上项目(2016JJ2101);湖南省教育厅优秀青年项目(15B192);吉首大学 2015 年实验教学改革研究项目(2015SYJG034);吉首大学大学生研究性学习和创新性实验计划项目(教通[2016]13 号);吉首大学研究生教改项目(JG201615);吉首大学研究生科研创新项目(JGY201643)

通信作者:肖　林(1986—),男,湖南邵阳人,吉首大学信息科学与工程学院讲师,博士,主要从事人工智能、机器人和神经网络研究.

### 1.2 梯度神经网络模型设计

1.2.1 基于误差函数(1) 设计梯度神经网络模型　误差函数是一个可正可负的函数，为了使函数图像都在第一象限里，取误差函数的绝对值.

步骤 1　基于梯度设计方法，定义一个基于平方的标准取值的误差函数 $\varepsilon(\boldsymbol{X})$，用于监控(1) 式的求解过程，

$$\varepsilon_1 = \|\boldsymbol{AX} - \boldsymbol{B}\|^2/2. \tag{3}$$

很明显，求解的问题是要方程(1) 为 0，故当误差函数 $\varepsilon_1 = 0$ 时，相对应的 $\boldsymbol{X}$ 的解就是目的解.那么，现在的问题发生了转移，只要求出误差函数 $\varepsilon_1 = 0$ 时的 $\boldsymbol{X}$ 的解就行了.

步骤 2　为使能量函数(3) 收敛到 0，运用梯度下降的思想，可以使它沿着负梯度方向下降，于是对其求导得

$$-\frac{\partial \varepsilon_1}{\partial \boldsymbol{X}} = -\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{AX} - \boldsymbol{B}). \tag{4}$$

步骤 3　基于梯度设计，取参数 $-\lambda$，使方程(4) 沿负梯度方向下降，即

$$\dot{\boldsymbol{X}} = -\lambda \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{AX} - \boldsymbol{B}). \tag{5}$$

其中，函数 $x(t)$ 从初始值 $x(0)$ 出发，对应于方程(1) 的解，参数 $-\lambda$ 为自定义设计，主要作用是调节模型(4) 的梯度收敛速度.

1.2.2 基于误差函数(2) 设计梯度神经网络模型　误差函数是一个可正可负的函数，为了使函数图像都在第一象限里，取误差函数的绝对值.

步骤 1　基于梯度设计方法，定义一个基于平方的标准取值的误差函数 $\varepsilon(\boldsymbol{X})$，用于监控(2) 式的求解过程，

$$\varepsilon_2 = \|\boldsymbol{X} - \boldsymbol{A}^{-1}\boldsymbol{B}\|^2/2. \tag{6}$$

很明显，求解的问题是要方程(2) 为 0，故当误差函数 $\varepsilon_2 = 0$ 时，相对应的 $\boldsymbol{X}$ 的解就是目的解.那么，现在的问题发生了转移，只要求出误差函数 $\varepsilon_2 = 0$ 时的 $\boldsymbol{X}$ 的解就行了.

步骤 2　为使误差函数(6) 收敛到 0，运用梯度下降的思想，可以使它沿着负梯度方向下降，于是对其求导得

$$-\frac{\partial \varepsilon_2}{\partial \boldsymbol{X}} = -(\boldsymbol{X} - \boldsymbol{A}^{-1}\boldsymbol{B}). \tag{7}$$

步骤 3　基于梯度设计，取参数 $-\lambda$，使方程(7) 沿负梯度方向下降，即

$$\dot{\boldsymbol{X}} = -\lambda(\boldsymbol{X} - \boldsymbol{A}^{-1}\boldsymbol{B}). \tag{8}$$

其中，函数 $x(t)$ 从初始值 $x(0)$ 出发，对应于方程(2) 的解，参数 $-\lambda$ 为自定义设计，主要作用是调节模型(7) 的梯度收敛速度.

## 2　计算机仿真验证

基于误差函数(1)，(2)，设计 2 个不同的梯度神经网络模型.为了验证基于不同误差函数设计的不同梯度神经网络模型的有效性，挑选几个具有代表性的线性方程进行求解.

**例 1**　考虑如下线性方程的求解：

$$\boldsymbol{AX} = \boldsymbol{B}, \tag{9}$$

其中 $\boldsymbol{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 2 & 6 \\ 7 & 4 & 9 \end{pmatrix}$，$\boldsymbol{B} = \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix}$.

很显然，在实数域里，方程(9) 的解为 $\boldsymbol{X}$.下面通过基于 2 个误差函数设计的 2 个梯度神经网络模型(5) 和(8) 来求解此线性方程.不失一般性，令参数 $\lambda = 1$，初始状态在[5，5] 区域内随机产生，得到的仿真结果如图 1—4 所示.对于 2 个不同的梯度神经网络模型求解梯度神经网络，对比图 1，3 可知，此线性方程的解为 $x_1 = 0.500\,0$，$x_2 = 1.000\,0$，$x_3 = -0.500\,0$，证实了 2 个不同梯度神经网络模型求解线性方程的正确性；对比图 2，4 可知，梯度神经网络模型(8) 收敛到 0 的速度比梯度神经网络模型(5) 的快，这可能是因为梯度神经网络模型的不同而导致在线性方程求解过程中收敛速度不同.

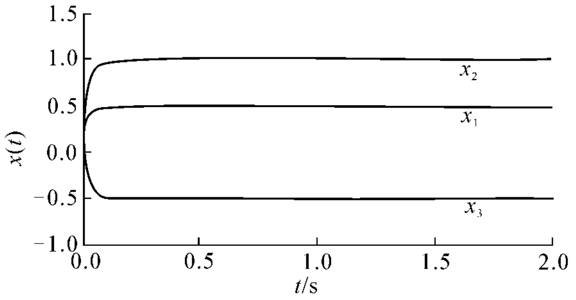


图 1 梯度神经网络模型(5) 仿真结果

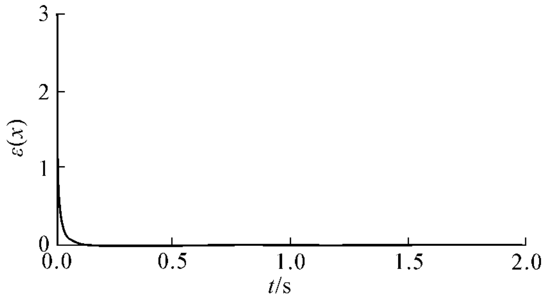


图 2 梯度神经网络模型(5) 收敛情况

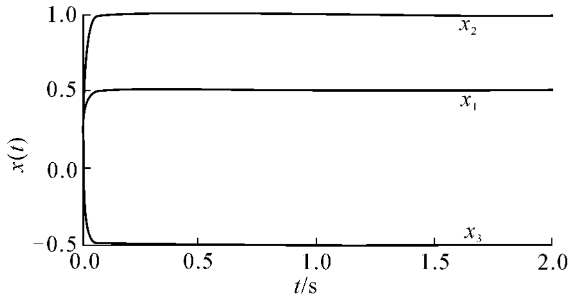


图 3 梯度神经网络模型(8) 仿真结果

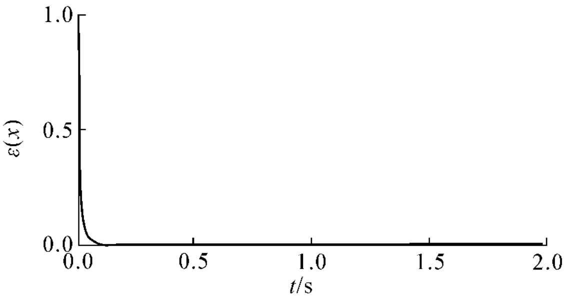


图 4 梯度神经网络模型(8) 收敛情况

## 3 结语

在线性方程的求解过程中,对于一般线性方程,可以设计出多个不同的误差函数.根据这多个不同的误差函数,按照梯度神经网络模型的设计方法可以建立多个不同的梯度神经网络模型.利用 Matlab 仿真软件进行计算机仿真,验证各梯度神经网络模型在求解线性方程中的正确性,并模拟求解过程中误差函数的收敛情况.对比各梯度神经网络模型的误差函数收敛情况,可以得到准确、高效求解某一类线性方程的梯度神经网络模型.

参考文献:

[1] 李海滨,尚凡华.基于神经网络的病态线性方程组求解[J].辽宁工程技术大学学报,2007,26(6):956-958.

[2] XIAO Lin.A Finite-Time Convergent Neural Dynamics for Online Solution of Time-Varying Linear Complex Matrix Equation[J].Neurocomputing,2015,167:254-259.

[3] 肖 林,皮赛男,孟凡斌.梯度神经网络在 $p$ 次方根求解中的应用[J].吉首大学学报(自然科学版),2015,36(3):15-18.

[4] 张雨浓,史艳燕,蔡炳煌,等.梯度神经网络解线性矩阵方程之收敛性分析[J].控制工程,2012,19(2):235-239.

# Neural Network Based on Different Error Functions for Solving Linear System of Equations

WANG Sicheng,XIAO Lin,YAN Huiling

(College of Information Science and Engineering,Jishou University,Jishou 416000,Hunan China)

**Abstract**:There are many ways to solve a linear equation,and various solving methods will result in different convergence speeds.Two different error functions are designed for one linear equation,and then two gradient neural network models are established according to the gradient descent method.Matlab is used for computer simulation,the solutions of linear equation are given according to the different gradient neural network models,and the feasibility of these two gradient neural network models is confirmed.Finally,the convergence speeds in solving the linear equation are compared.

**Key words**:error function;gradient neural network;system of linear equations;Matlab simulation

(责任编辑 向阳洁)