

# 基于 Hensel 构造的回归神经网络符号计算模型及算法\*

## Recurrent Neural Networks Symbol Computation Model and Algorithm Based on Hensel Construction

周永权 刘宣会\*\*

Zhou Yongquan Liu Xuanhui

(广西民族学院数学与计算机科学系 南宁市大学路 80号 530006)

(Dept. of Math. &amp; Comp. Sci., Guangxi Univ. for Nationalities, 80 Daxuelu, Nanning, Guangxi, 530006, China)

**摘要** 将传统意义下 Hensel 构造提升的方法与回归神经网络模型有机地结合起来,提出一种基于 Hensel 构造方法的回归神经网络近似代数符号计算新模型和 PFRNN 网络算法.该模型不但具有回归神经网络的特点,而且具有 Hensel 构造提升的思想,给人们研究代数符号计算与近似代数符号计算提供一种可视化手段.通过多元多项式近似因式分解算例分析可以看出,新模型刻划出在符号计算意义下精确计算与近似计算的本质与联系.

**关键词** 回归神经网络 Hensel 构造方法 近似分解 代数符号计算

中图法分类号 TP183

**Abstract** Under the traditional Hensel construction method and recurrent neural networks model, a new recurrent neural networks symbol algebra symbol computation model and PFRNN Algorithm based on Hensel construction is proposed. It has the characteristics of traditional RNN and the capability of function approximation, and may offer a kind of visual means for studying algebra symbol calculation and approximate algebra symbol calculation. Through multivariate polynomials approximate factorization, the essences and relationships between approximate calculation and accurate calculation are explained.

**Key words** recurrent neural networks, Hensel construction method, approximate factorization, algebra symbol computation

Hensel 构造方法最先用于求解一元多项式的  $P$ -adic 展开式,它是由 Hensel 本人在 1900 年提出,后来经 Wang 和 Rothschild 推广到多元多项式的情形,并应用于整数域上的多项式因式分解<sup>[1]</sup>.发展到今天, Hensel 构造方法和提升的思想,在代数符号计算、近似符号计算等领域,有着极其重要的理论价值和背景.作者认为,与 Hensel 构造方法有关的代数计算几乎都是一些精确计算,从函数逼近的观点来看,它只是一种在  $P$ -adic 意义下的逼近,完全不同于通常意义下一般函数的逼近,因而限制了 Hensel 构造方法的应用范围.因此,在代数符号意义下,如何刻划精确计算和近似计算之间的本质联系是值得探讨的问题.有关这方面的研究目前还未有相关文献的报道.

基于此想法,文中利用回归神经网络 RNN (Recurrent Neural Networks) 既有前馈通路,又有反馈通路的神经网络的特点,设置隐层中神经元的激活函数为多项式函数序列,将 Hensel 构造提升的思想与回归神经网络的特点有机地结合起来,提出了一种基于 Hensel 构造的回归神经网络近似符号计算新模型(图 1),它不但具有传统回归神经网络的特点,而且具有 Hensel 构造提升的思想和较强的函数逼近能力.使得  $P$ -adic 逼近意义下 Hensel 构造方法类似于一般意义下的函数逼近,推广了 Hensel 构造方法应用范围.其目的给人们研究代数符号计算与近似代数符号计算提供一种可视化手段.文中还以多元多项式近似因式分解为例,刻划出在代数符号计算意义下,精确计算与近似计算的本质与联系.

### 1 多项式函数型回归神经网络模型

多项式函数型回归神经网络 PFRNN (Polynomial function RNN) 是一种既有前馈通路,又有反馈通路的函数型神经网络,与传统 RNN 区别在于,其

2002-11-18 收稿.

\* 广西自然科学基金 (0141034) 及广西高校百名中青年学科带头人的资助.

\*\* 西安电子科技大学经济管理学院 陕西西安 710071 (College of Economic Manage, Xi'an University, Xi'an, Shannxi, 710071, China)

隐层  $L$  层经由一组激活函数为多项式函数的神经元映射的输出经过一步或几步递归后送到其它神经元或自身;反馈通路的作用,使得神经网络能够用来有效地处理递归序列的上下文相关的计算;而传统的 RNN 的  $L$  层神经元每一位输出代表着一个识别单位发生概率大小,而不是一些具体的激活函数,处理实际问题往往受到专家领域知识的限制,在很大程度上限制了 RNN 网络的应用范围.因此,有必要进行推广.推广后的网络,我们称为 PFRNN 网络,其 PFRNN 神经网络结构如图 1 所示.

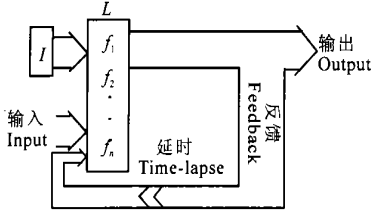


图 1 PFRNN 网络结构

Fig. 1 Network structure of PFRNN

其中  $I$  层是用来为  $L$  层产生初始取值,通常根据经验人为的设定,称为附加层, $L$  层的每一结点代表一个神经元  $f_i (i = 1, 2, \dots, n)$ ,所对应的激活函数序列为  $h_i (i = 1, 2, \dots, n)$ .对图 1 的多项式函数型 RNN 网络,对于一递归步长为  $N$ ,可把该网络按递归步长逐步展开,并且把展开后的网络看作是一个在各步之间共享权值的大网络,上、下步计算结果传递是通过每一步的隐层  $L$  层激活函数通过映射来完成.实际上图 1 神经网络展开后整个网络的整体结构如图 2 所示.

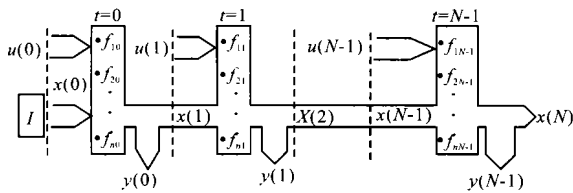


图 2 PFRNN 神经网络展开模型

Fig. 2 Outspread model of neural networks of PFRNN

其中:  $t$  代表递归步长, $u(t)$  为输入向量, $x(t)$  为反馈向量, $y(t)$  为输出向量.这样,我们可通过对初始权值  $I$  的训练,和每一步层  $L$  中激活函数序列的选取,便可完成对 PFRNN 网络的整体训练学习.

## 2 基于 Hensel 构造的 PFRNN 网络算法

在设计基于 Hensel 构造的 PFRNN 网络算法时,首先,以引理形式给出传统的 Hensel 构造方法,然后再给出基于 Hensel 构造的 PFRNN 网络算法:

引理 1<sup>[2]</sup> 一般 Hensel 引理 设  $F(x, y, \dots, z)$  是一多元多项式,对任何正整数  $k$ ,都存在着一多项

式集  $\{F_1^{(k)}, \dots, F_n^{(k)}\}$ , 满足:

$$F(x, y, \dots, z) \equiv F_1^{(k)}(x, y, \dots, z) \cdots F_n^{(k)}(x, y, \dots, z) \pmod{S^{k+1}},$$

$$F_i^{(k)} = F_i^{(0)} \pmod{S} \quad i = 1, 2, \dots, n,$$

其中  $S = (y, \dots, z)$  为由  $y, \dots, z$  构成理想.

下面以二元多项式为例,给出基于 Hensel 构造的 PFRNN 网络算法的描述.

在图 2 中,给定一递归长度为  $N$  的输入输出向量序列,其训练过程和步骤如下:

步骤 1 把  $x(0)$  设为从附加层  $I$  输出的初始权值, $u(0)$  设为第一步输入,向前传播计算得到  $y(0)$  和  $x(1)$ ;

步骤 2 对步长  $t > 0$ ,把  $x(t)$  设为前一步的输出, $u(t)$  设为当前步长的输入,向前传播计算得到  $y(t)$  和  $x(t+1)$ ,设  $z(t) = (u(t), x(t))^T$ , 则

$$y_i(t+1) = f_{i1}(w_j^{(t)} z(t)) = \sum_{j=0}^n w_j^{(t)} h_j^{(t)}(z(t)), \quad (1)$$

其中  $w_j^{(t)}$  为计算到第  $t$  步网络的权值,  $h_j^{(t)}(z(t))$  为第  $t$  步  $L$  层神经元  $f_{i1} (i = 1, 2, \dots, n; t = 0, 1, \dots, N-1)$  所对应的激活函数.

步骤 3 置步长  $t = 0$  变化到  $N-1$ ,神经元个数  $j = 0$  变化到  $n$ ,那么,整个 PFRNN 网络每一步输出:

$$y_i(t+1) = y_i(t) + \sum_{j=0}^n w_j^{(t)} h_j^{(t)}(z(t)), \quad i = 1, 2, \dots, n, \quad (2)$$

其中  $z(t)$  为参数,在实际应用中在确定.

步骤 4 重复步骤 1~ 步骤 3,计算出 PFRNN 网络在每一步输出,直到递归步长  $t = N-1$  为止.这样,我们可通过以上算法中计算公式就可完成对 PFRNN 的训练学习.

现考察该网络第  $t$  步  $L$  层结点的输出:

$$y_i(t+1) = f_{i1}(w_j^{(t)} z(t)) = \sum_{j=0}^n w_j^{(t)} h_j^{(t)}(z(t)), \quad (3)$$

显然,若将 (3) 中  $h_j^{(t)}$  看作逼近的基函数,则 (3) 定义了一个函数逼近结构.而且,式 (3) 中的参数  $w_j^{(t)}$ ,  $h_j^{(t)}$ ,  $z(t)$  都可调,使得用于逼近的基函数相应调整.所以,相比一般逼近方法,神经网络所定义的函数逼近结构的优越性在于它不但是一个对逼近系数寻优的过程,而且是一个对逼近基函数序列自适应寻优的过程.当然,由于  $h_j^{(t)}$  本身的限制,根据实际问题的需要,上述基函数的寻优是在一定范围内进行,随着 PFRNN 网络递归步长的增加,迭加的结果,使得基函数的自由度增加了,这是传统的逼近方法所不能比

拟的.其次,在式(3)中,  $h^0$  它又是 PFRNN 网络的激活函数,其形式不同于一般神经网络的激活函数的取法.一旦选定了网络的结构,那么,各逼近的参数可通过统一的训练算法求得,使得算法的适应性好,不依赖于数据.

设  $F(x, y) \in C[x, y], n = \deg_x F$  为  $F$  关于变元  $x$  的次数,其中  $C[x, y]$  为复数多项式域,若多项式  $F(x, y)$  能分解,则有:

$$F(x, y) = F_1(x, y)F_2(x, y) \cdots F_n(x, y),$$

其中  $F_j(x, y) = x + c_{j0} + c_{j1}y + c_{j2}y^2 + \cdots + c_{jk}y^k + \cdots, j = 1, 2, \dots, n,$  (4)

在(4)式中,  $c_{ji}$  为对应权值,其确定方法如下:

引理 2<sup>[3]</sup> 对任一多元多项式  $F(x, y)$ ,若有根  $x = h(y)$ ,其中  $h(y) = c_0 + c_1y + c_2y^2 + \cdots, j = 1, 2, \dots, \deg_x(F)$ ,则它的每一近似因式都是形如  $x - h(y)$  的形式,且存在一双输入、单输出的 4 层前向展开神经网络通过训练可导出每一  $F_j(x, y)$ .

引理 3<sup>[3]</sup> 对任一给定多项式  $F(x, y)$ ,其中  $n = \deg_x(F)$ ,总存在唯一的 PFRNN 网络,通过对(4)式的系数和基函数的训练,可递归地计算每一近似因式  $F_j(x, y)$ . ( $j = 1, 2, \dots, n$ ) 直到  $y^k$  项 ( $k \geq e = \deg_y(F)$ ).

引理 4<sup>[3]</sup> 对任意一正整数  $l, 0 \leq l \leq n - 1$ ,通过权值调整算法所得到权值集  $\{w_1^{(l)}, \dots, w_n^{(l)}\}$  满足:  $w_1^{(l)} [F_1 \cdots F_n / F_1] + \cdots + w_n^{(l)} [F_1 \cdots F_n / F_n] = x^l$ . (5)

现回过头来,考察基函数序列的确定方法.在图 2 的 PFRNN 网络中,为了与多项式符号表示一致,用  $F_i^{(t)}$  仍表示该网络第  $t$  步的第  $i$  个神经元的输出,那么,第  $t + 1$  步第  $i$  个神经元的输出,由式(2)有:

$$F_i^{(t+1)} = F_i^{(t)} + \Delta F_i^{(t+1)} = F_i^{(t)} + \sum_{l=0}^{n-1} w_l^{(t)} h^l(y),$$
 (6)

另一方面,从近似因式分解目的来讲,误差函数满足:

$$\Delta F^{(t+1)} \equiv F - F_1^{(t)} F_2^{(t)} \cdots F_n^{(t)} = \Delta F_1^{(t+1)} [F_1 F_2 \cdots F_n / F_1] + \cdots + \Delta F_n^{(t+1)} [F_1 F_2 \cdots F_n / F_n],$$
 (7)

结合引理 4 及式(7)进一步可表示成:

$$\Delta F^{(t+1)} \equiv h_{a-1}^{(t)} x^{n-1} + h_{a-2}^{(t)} x^{n-2} + \cdots + h_b^{(t)},$$
 (8)

通过比较(7), (8)两式,就可确定出 PFRNN 网络递归的每一步基函数或激活函数序列  $h^l$ .

这样,我们就完成了基于 Hensel 构造的回归神经网络模型的建立及学习算法的设计.

### 3 算例及分析

#### 3.1 算例

例 1<sup>[4]</sup> 在  $C[x, y]$  中,近似分解多项式  $F(x, y) = x^2 - 4y^2 - 4y - 1$ .

首先,设  $y = 0$ ,可得到一元多项式  $F(x, 0) = x^2 - 1 = (x - 1)(x + 1)$ ,附加层  $l$  初始权值选取: 1 和 -1,用权值调整算法<sup>[3]</sup>,通过训练最后的权值:

$$\begin{pmatrix} u(0) \\ u(1) \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

递归步长  $t = 0, 1$  那么,在第  $t = 0$  步,  $L$  层神经元  $f^{i0} (i = 1, 2)$  对应激活函数:

$$h^0(y) = 0 \quad i = 1, 2,$$

输出向量:  $x(0) = (x, y)^T$ ,通过计算得到第 0 步输出:  $F_1^{(0)} = x - 1, F_2^{(0)} = x + 1$  在  $t = 1$ ,选取基函数:  $h^1(y) = 0, h_b^1(y) = -4y^2 - 4y$ ,向前传播计算  $t = 1$  步,输出  $y(1)$  每一分量:

$$F_1^{(1)} = F_1^{(0)} + \sum_j w_j^{(1)} h^1(y) = x - 1 - 2y^2 - 2y,$$

$$F_2^{(1)} = F_2^{(0)} + \sum_j w_j^{(1)} h^1(y) = x + 1 + 2y^2 + 2y,$$

同理,在  $t = 2$  步,输出  $y(2)$  每一分量,此时,基函数选取:

$h^2(y) = 0, h_b^2(y) = 4y^2 + 8y^3 + 4y^4$ ,此时 PFRNN 网络的输出:

$$F_1^{(2)} = x - 1 - 2y + 4y^3 + 2y^4,$$

$$F_2^{(2)} = x + 1 + 2y - 4y^3 - 2y^4,$$

为所求给定多项式近似因式.

#### 3.2 算例分析

下面以例 1 来分析文中提出的网络算法的优越性在于:

(I) 与文献 [4] 中提出的 NNBFM (neural network based factorization model) 算法相比,该算例采用  $2 \times 2 \times 2 \times 1$  网络结构实现,经 42034 次迭代才完成近似分解,得到 2 个近似因式:  $x - 1.998y - 0.999$  及  $x + 1.999y + 0.999$ ;若用文中提出的基于 Hensel 构造的回归神经网络算法,只须迭代 3 次,就可到 2 个近似因式:  $F_1^{(2)} = x - 1 - 2y + 4y^3 + 2y^4, F_2^{(2)} = x + 1 + 2y - 4y^3 - 2y^4$ ,由于所给多项式关于变元的次数为 2,若去掉 2 个因式中的次数大于 2 的项,不难得到 2 个精确因式.由此可见,文中提出的算法收敛速度快,计算精度高.

(II) 与传统的 Hensel 构造方法相比,当给定多  
(下转第 182 页 Continue on page 182)

$$\int_{t_1}^{+\infty} Q_1(t) |f(t, x(t), x(g(t)), x'(t), x'(h(t)))| dt + \int_{t_1}^{+\infty} Q_1(t) |e(t)| dt < \frac{m}{3} + \frac{m}{3} = \frac{2}{3}m.$$

此式矛盾,故  $\limsup_{t \rightarrow +\infty} |x(t)| = 0$ , 又  $x(t)$  是方程 (E) 的振动解, 则  $\liminf_{t \rightarrow +\infty} |x(t)| = 0$ . 从而  $\lim_{t \rightarrow +\infty} x(t) = 0$ .

### 例 3 考虑方程

$$\left[ \frac{1+x^2(t)}{2+x^2(t)} \frac{1}{t} x'(t) \right]' + \frac{1}{t^4} \frac{x(t) + x(\ln(t))}{[x(t) + x(\ln(t))]^2 + 1} [1 + (x'(t))^2 + (x'(\frac{1}{t}))^2]^{-1} = \frac{1}{t^4}, \quad (10)$$

在此方程中, 可取  $r_1(t) = \frac{1}{2t}$ , 从而  $Q_1(t) = t^2 - t_0^2$ ,

$$\int_{t_0}^{+\infty} Q_1(t) |e(t)| dt = \int_{t_0}^{+\infty} (t^2 - t_0^2) \frac{1}{t^4} dt < \int_{t_0}^{+\infty} \frac{1}{t^2} dt < +\infty, \quad (11)$$

$$\int_{t_0}^{\infty} Q_1(t) |f(t, x(t), x(g(t)), x'(t), x'(h(t)))| dt = \int_{t_0}^{\infty} (t^2 - t_0^2) \left| \frac{1}{t} \left[ \frac{x(t) + x(\ln(t))}{[x(t) + x(\ln(t))]^2 + 1} [1 + (x'(t))^2 + (x'(\frac{1}{t}))^2]^{-1} \right] \right| dt$$

$$(x'(\frac{1}{t}))^2]^{-1} dt \leq \int_{t_0}^{\infty} \frac{1}{2t^2} dt < +\infty.$$

由 (11) 式, 知 (C<sub>1</sub>) 成立; 由 (12) 式, 知 (C<sub>2</sub>) 成立. 利用定理 3 可知方程 (10) 的一切有界振动解当  $t \rightarrow +\infty$  时趋于零.

### 参考文献

- 1 张炳根. 泛函微分方程振动理论的发展. 科学通报, 1998, 43: 345~354.
- 2 燕居让.  $n$  阶非线性时滞微分方程的振动性与渐近性. 数学学报, 1990, 33: 537~545.
- 3 王其如. 二阶非线性微分方程的振动准则. 数学学报, 2001, (44): 2.
- 4 仇志余. 非线性二阶微分方程的振动准则. 系统科学与数学, 2000, 20(1): 1~10.
- 5 Ladde G S, Lakshmikantham V, Zhang B G. Oscillation Theory of Differential Equation with Deviating Arguments. New York: Marcel Dekker, 1987.
- 6 Hamedani G G, Krenz G S. Oscillation criteria for certain second differential equations. Math Anal Appl, 1990, 149: 271~276.
- 7 郑祖麻. 泛函微分方程理论. 合肥: 安徽教育出版社, 1992. 346.

(责任编辑: 黎贞崇)

(上接第 178 页 Continue from page 178)

项式次数较高时, 存在着组合爆炸问题; 由于神经网络具有大规模并行处理, 良好的自组织性、自适应性等优点, 因此, 本文提出的算法, 特别适合于多维计算情形.

(III) 从整体来看, Hensel 构造方法的神经网络模型相比传统的 Hensel 构造方法, 由于每步逼近基函数选取自由度大, 无须进行模运算. 因此, 算法稳定性、容错性好.

(IV) 从该算例看出, 基于 Hensel 构造的回归神经网络近似符号计算新模型, 完全刻划出在代数符号计算意义下, 精确计算与近似计算的本质与联系.

### 4 结束语

文中将 Hensel 构造提升的方法与回归神经网络的特点有机地结合起来, 提出了一种基于 Hensel 构造方法的回归神经网络近似符号计算新模型, 它不但具有传统回归神经网络的特点, 而且具有 Hensel 构造提升的方法和较强的并行计算功能, 其目的是给人

们研究代数符号计算与近似代数符号计算提供一种可视化手段, 完全刻划精确计算与近似计算的本质与联系.

### 参考文献

- 1 Wang P S, Rothschild L P. Factoring polynomials over the integers. Math Comp, 1975, (29): 936~950.
- 2 Tateaki Sasaki, Massayuki Suzuki, Miroslav Kolar et al. Approximate factorization of multivariate polynomials and absolute irreducibility testing. Japan J Indust Appl Math, 1991, 5(3): 357~375.
- 3 周永权. 基于代数神经网络的多元多项式近似因式分解模型及学习算法. 计算机研究与发展, 1999, 36(6): 668~674.
- 4 Huang D S. A neural network based factorization model for polynomials in several elements. In: Beijing 2000 5th International Conference on Signal Processing Proceedings (WCC2000-ICSP2000), 2000, 1617~1622.

(责任编辑: 黎贞崇)