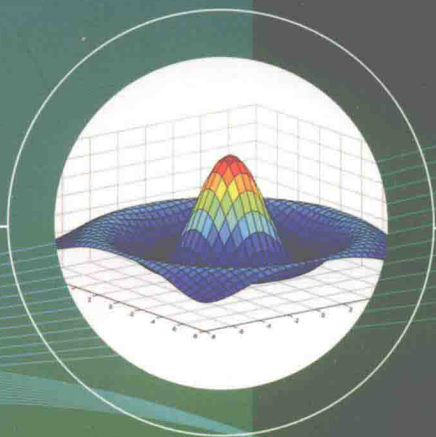


— 创新型人才培养“十三五”规划教材

神经·模糊·预测控制 及其 MATLAB 实现 (第4版)

— ◆ 李国勇 杨丽娟 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

创新型人才培养 “十三五” 规划教材

神经·模糊·预测控制 及其 MATLAB 实现 (第4版)

李国勇 杨丽娟 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书系统地论述了神经网络控制、模糊逻辑控制和模型预测控制的基本概念、工作原理、控制算法,以及利用 MATLAB 语言、MATLAB 工具箱函数和 Simulink 对其实现的方法。本书取材先进实用,讲解深入浅出,各章均有相应的例题,并提供了大量用 MATLAB/Simulink 实现的仿真实例,便于读者掌握和巩固所学知识。

本书可作为高等院校自动化、电气工程及其自动化、电子科学与技术、计算机科学与技术、测控技术与仪器、交通设备与控制工程、轨道交通信号与控制、机械设计制造及其自动化和机械电子工程等专业研究生和高年级本科生的教材,也可作为从事智能控制与智能系统研究、设计和应用的科学技术人员的参考用书。

本书还有配套的教学课件,可登录电子工业出版社的华信资源教育网 www.hxedu.com.cn 免费下载。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

神经·模糊·预测控制及其 MATLAB 实现/李国勇,杨丽娟编著. —4 版. —北京:电子工业出版社, 2018.11

创新型人才培养“十三五”规划教材

ISBN 978-7-121-35168-6

I. ①神… II. ①李… ②杨… III. ①Matlab 软件—应用—人工神经网络—高等学校—教材 IV. ①TP183

中国版本图书馆 CIP 数据核字(2018)第 227780 号

责任编辑:牛平月(niupy@phei.com.cn)

印 刷:北京天宇星印刷厂

装 订:北京天宇星印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:24 字数:614.4 千字

版 次:2005 年 5 月第 1 版

2018 年 11 月第 4 版

印 次:2018 年 11 月第 1 次印刷

定 价:89.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: niupy@phei.com.cn。

第 4 版前言

本书自 2005 年 5 月第一版、2010 年 1 月和 2013 年 5 月的两次再版以来,得到广大读者的关心和支持,先后多次重印,被国内多所大学选为教材。

这次修订在保持前三版内容系统,实用、易读的特点,以及框架结构基本不变的基础上,主要修改了第 1 章和第 4 章的内容,并充分考虑了新形势下智能控制类课程教学及不同层次院校的选学需要,体现宽口径专业教育思想,反映先进的技术水平,强调教学实践的重要性,有利于学生自主学习和动手实践能力的培养,适应卓越工程师人才培养的要求,同时也满足多学科交叉背景学生的教学需求。

本书是本着把当前国际控制界最为流行的面向工程与科学计算的高级语言——MATLAB 与神经网络、模糊逻辑和预测控制相结合的宗旨编写的。书中主要从三个方面阐述利用 MATLAB 对神经网络、模糊逻辑和模型预测控制系统的计算机仿真方法。其中,第一种方法为采用 MATLAB 语言根据具体的控制算法编程进行仿真;第二种方法为利用 MATLAB 提供的神经网络、模糊逻辑和预测控制工具箱函数或图形用户界面直接进行仿真;第三种方法为根据 Simulink 动态仿真环境进行仿真。比较以上三种方法,第二种方法最为简单,它不需要了解算法的本质,就可以直接应用功能丰富的函数来实现自己的目的;第三种方法最为直观,它可以在运行仿真时观察仿真结果;第一种方法最为复杂,它需要根据不同的控制算法进行具体编程,但这种方法也最为灵活,使用者可以根据自己所提出的新算法任意编程,该方法主要用于对某种新控制算法的仿真和应用。当然利用其他计算机语言也可根据控制算法进行具体编程,但比较而言,以利用 MATLAB 编程最为简单,因为 MATLAB 具有强大的矩阵运算和图形处理功能。而第二种和第三种方法较适合初学者,主要用于对某种成熟控制算法的仿真和应用。由于其编程简单,给使用者节省了大量的编程时间,使其能够把更多的精力投入到网络设计而不是具体程序的实现上。

本书可作为高等院校自动化类、电气类、电子信息类、计算机类、仪器类和机械类等学科类研究生和高年级本科生的教材,也可作为从事智能控制与智能系统研究、设计和应用的科学技术人员的参考用书。鉴于本书的通用性和实用性较强,也可作为相关专业的教学、研究、设计人员和工程技术人员的参考用书。

本书由李国勇和杨丽娟编著,共包含 9 章和 3 个附录。其中,第 1 章~第 7 章和第 9 章由李国勇编写;第 8 章和附录 A 由杨丽娟编写;附录 B 和附录 C 由阎高伟、王芳、任密蜂、杨云云和杨麦仓编写。此外,感谢责任编辑牛平月女士为本书的出版所付出的辛勤工作。

本书配套的教学课件,可登录电子工业出版社的华信资源教育网 www.hxedu.com.cn 免费下载。

为便于读者查阅,部分章节中矩阵、向量等符号与 MATLAB 代码及软件界面中一致,未做标准化处理,在此特加以说明。

由于作者水平有限,书中仍难免有遗漏与不当之处,恳请有关专家、同行和广大读者批评指正。

作者

2018 年 6 月

目 录

第一篇 神经网络控制及其 MATLAB 实现

第 1 章 神经网络理论	(1)
1.1 神经网络的基本概念	(2)
1.1.1 生物神经元的结构与功能特点	(2)
1.1.2 人工神经元模型	(3)
1.1.3 神经网络的结构	(5)
1.1.4 神经网络的工作方式	(6)
1.1.5 神经网络的学习	(6)
1.1.6 神经网络的分类	(9)
1.2 典型神经网络的模型	(9)
1.2.1 MP 模型	(9)
1.2.2 感知机	(11)
1.2.3 自适应线性神经网络	(15)
1.2.4 BP 神经网络	(17)
1.2.5 径向基神经网络	(27)
1.2.6 竞争学习神经网络	(31)
1.2.7 学习向量量化神经网络	(40)
1.2.8 Elman 神经网络	(41)
1.2.9 Hopfield 神经网络	(42)
1.2.10 Boltzmann 神经网络	(58)
1.3 神经网络的训练	(61)
小结	(65)
思考练习题	(65)
第 2 章 MATLAB 神经网络工具箱函数	(66)
2.1 MATLAB 神经网络工具箱简介	(66)
2.1.1 神经网络工具箱中的通用函数	(66)
2.1.2 感知机 MATLAB 函数	(69)
2.1.3 线性神经网络 MATLAB 函数	(79)
2.1.4 BP 神经网络 MATLAB 函数	(85)
2.1.5 径向基神经网络 MATLAB 函数	(96)
2.1.6 自组织神经网络 MATLAB 函数	(102)
2.1.7 学习向量量化神经网络 MATLAB 函数	(115)
2.1.8 Elman 神经网络 MATLAB 函数	(118)
2.1.9 Hopfield 神经网络 MATLAB 函数	(121)
2.1.10 利用 Demos 演示神经网络的建立	(126)
2.2 MATLAB 神经网络工具箱的图形用户界面	(127)

2.2.1	神经网络编辑器	(128)
2.2.2	神经网络拟合工具	(136)
2.3	基于 Simulink 的神经网络模块	(139)
2.3.1	模块的设置	(139)
2.3.2	模块的生成	(140)
2.4	神经网络在系统预测和故障诊断中的应用	(143)
2.4.1	系统输入/输出数据的处理	(143)
2.4.2	基于神经网络的系统预测	(144)
2.4.3	基于神经网络的故障诊断	(155)
	小结	(162)
	思考练习题	(162)
第 3 章	神经网络控制系统	(163)
3.1	神经网络控制理论	(163)
3.1.1	神经网络控制的基本原理	(163)
3.1.2	神经网络在控制中的主要作用	(164)
3.1.3	神经网络控制系统的分类	(165)
3.2	基于 Simulink 的三种典型神经网络控制系统	(174)
3.2.1	神经网络模型预测控制	(175)
3.2.2	反馈线性化控制	(180)
3.2.3	模型参考控制	(182)
	小结	(185)
	思考练习题	(186)

第二篇 模糊逻辑控制及其 MATLAB 实现

第 4 章	模糊逻辑控制理论	(187)
4.1	模糊逻辑理论的基本概念	(187)
4.1.1	模糊集合及其运算	(187)
4.1.2	模糊关系及其合成	(194)
4.1.3	模糊向量及其运算	(196)
4.1.4	模糊逻辑规则	(197)
4.1.5	模糊逻辑推理	(199)
4.2	模糊逻辑控制系统的基本结构	(205)
4.2.1	模糊控制系统的组成	(205)
4.2.2	模糊控制器的基本结构	(206)
4.2.3	模糊控制器的维数	(206)
4.3	模糊逻辑控制系统的基本原理	(207)
4.3.1	模糊化运算	(207)
4.3.2	数据库	(208)
4.3.3	规则库	(211)
4.3.4	模糊推理	(213)
4.3.5	去模糊化	(215)
4.4	离散论域的模糊控制系统的设计	(217)

4.5 具有 PID 功能的模糊控制器	(222)
小结	(223)
思考练习题	(223)
第 5 章 MATLAB 模糊逻辑工具箱函数	(224)
5.1 MATLAB 模糊逻辑工具箱简介	(224)
5.1.1 模糊逻辑工具箱的功能特点	(224)
5.1.2 模糊推理系统的基本类型	(225)
5.1.3 模糊逻辑系统的构成	(225)
5.2 利用模糊逻辑工具箱建立模糊推理系统	(226)
5.2.1 模糊推理系统的建立、修改与存储管理	(226)
5.2.2 模糊语言变量及其语言值	(229)
5.2.3 模糊语言变量的隶属函数	(230)
5.2.4 模糊规则的建立与修改	(236)
5.2.5 模糊推理计算与去模糊化	(240)
5.3 MATLAB 模糊逻辑工具箱的图形用户界面	(243)
5.3.1 模糊推理系统编辑器	(243)
5.3.2 隶属函数编辑器	(245)
5.3.3 模糊规则编辑器	(245)
5.3.4 模糊规则浏览器	(246)
5.3.5 模糊推理输入/输出曲面浏览器	(246)
5.4 基于 Simulink 的模糊逻辑的系统模块	(248)
5.5 模糊推理系统在控制系统中的应用	(251)
小结	(256)
思考练习题	(256)
第 6 章 模糊神经和模糊聚类及其 MATLAB 实现	(258)
6.1 基于 Mamdani 模型的模糊神经网络	(258)
6.1.1 模糊系统的 Mamdani 模型	(258)
6.1.2 系统结构	(260)
6.1.3 学习算法	(261)
6.2 基于 Takagi-Sugeno 模型的模糊神经网络	(263)
6.2.1 模糊系统的 Takagi-Sugeno 模型	(264)
6.2.2 系统结构	(264)
6.2.3 学习算法	(266)
6.3 自适应神经模糊系统及其 MATLAB 实现	(268)
6.3.1 采用网格分割方式生成模糊推理系统函数	(269)
6.3.2 自适应神经模糊系统的建模函数	(270)
6.3.3 自适应神经模糊推理系统的图形用户界面编辑器	(272)
6.3.4 自适应神经模糊推理系统在建模中的应用	(275)
6.4 模糊聚类及其 MATLAB 实现	(281)
6.4.1 模糊 C-均值聚类函数	(281)
6.4.2 模糊减法聚类函数	(282)
6.4.3 基于减法聚类的模糊推理系统建模函数	(284)

6.4.4 模糊 C-均值和减法聚类的图形用户界面	(285)
小结	(288)
思考练习题	(288)

第三篇 模型预测控制及其 MATLAB 实现

第 7 章 模型预测控制理论	(289)
7.1 动态矩阵控制理论	(289)
7.1.1 预测模型	(289)
7.1.2 滚动优化	(291)
7.1.3 误差校正	(292)
7.2 广义预测控制理论	(292)
7.2.1 预测模型	(293)
7.2.2 滚动优化	(293)
7.2.3 反馈校正	(295)
7.3 预测控制理论分析	(296)
7.3.1 广义预测控制的性能分析	(296)
7.3.2 广义预测控制与动态矩阵控制规律的等价性证明	(300)
7.3.3 广义预测控制与动态矩阵控制的比较	(302)
小结	(302)
思考练习题	(302)
第 8 章 MATLAB 预测控制工具箱函数	(303)
8.1 系统模型辨识函数	(303)
8.1.1 数据向量或矩阵的归一化	(303)
8.1.2 基于线性回归方法的脉冲响应模型辨识	(305)
8.1.3 脉冲响应模型转换为阶跃响应模型	(308)
8.1.4 模型的校验	(309)
8.2 系统模型建立与转换函数	(309)
8.2.1 模型转换	(310)
8.2.2 模型建立	(314)
8.3 基于阶跃响应模型的控制器设计与仿真函数	(316)
8.3.1 输入/输出有约束的模型预测控制器设计与仿真	(316)
8.3.2 输入/输出无约束的模型预测控制器设计	(317)
8.3.3 计算由阶跃响应模型构成的闭环系统模型	(319)
8.4 基于状态空间模型的预测控制器设计函数	(320)
8.4.1 输入/输出有约束的状态空间模型预测控制器设计	(320)
8.4.2 输入/输出无约束的状态空间模型预测控制器设计	(321)
8.4.3 状态估计器设计	(324)
8.5 系统分析与绘图函数	(325)
8.5.1 计算和绘制系统的频率响应曲线	(326)
8.5.2 计算频率响应的奇异值	(327)
8.5.3 计算系统的极点和稳态增益矩阵	(327)
8.5.4 系统分析和绘图	(327)

8.6 通用功能函数	(328)
8.6.1 通用模型转换	(329)
8.6.2 方程求解	(330)
8.6.3 离散系统的分析	(330)
8.7 MATLAB 模型预测控制工具箱的图形用户界面	(331)
小结	(336)
思考练习题	(336)
第 9 章 隐式广义预测自校正控制及其 MATLAB 实现	(337)
9.1 单输入单输出系统的隐式广义预测自校正控制算法	(337)
9.2 多输入多输出系统的隐式广义预测自校正控制算法	(340)
9.3 仿真研究	(344)
9.3.1 单输入单输出系统的仿真研究	(344)
9.3.2 多输入多输出系统的仿真研究	(347)
小结	(348)
思考练习题	(348)
附录 A MATLAB 程序清单	(349)
附录 B MATLAB 函数一览表	(361)
附录 C MATLAB 函数分类索引	(367)
参考文献	(369)

第一篇 神经网络控制及其 MATLAB 实现

第 1 章 神经网络理论

人脑是一部不寻常的智能机，它能以惊人的高速度解释感觉器官传来的含糊不清的信息。它能觉察到喧闹房间内的窃窃私语，能识别出光线暗淡的胡同中的一张面孔，更能通过不断的学习而产生伟大的创造力。古今中外，许许多多科学家为了揭开大脑机能的奥秘，从不同的角度进行着长期不懈的努力和探索，逐渐形成了一个多学科交叉的前沿技术领域——神经网络（Neural Network）。

人工神经网络的研究可以追溯到 1800 年 Frued 的精神分析学时期，他已经做了一些初步工作。1913 年人工神经系统的第一个实践是由 Russell 描述的水力装置。1943 年美国心理学家 Warren S McCulloch 与数学家 Walter H Pitts 合作，用逻辑的数学工具研究客观事件在形式神经网络中的描述，从此开创了对神经网络的理论研究。他们在分析、总结神经元基本特性的基础上，首先提出神经元的数学模型，简称 MP 模型。从脑科学研究来看，MP 模型不愧为第一个用数理语言描述脑的信息处理过程的模型。后来 MP 模型经过数学家的精心整理和抽象，最终发展成一种有限自动机理论，再一次展现了 MP 模型的价值，此模型沿用至今，直接影响着这一领域研究的进展。1949 年心理学家 D.O.Hebb 提出关于神经网络学习机理的“突触修正假设”，即突触联系效率可变的假设，现在多数学习机仍遵循 Hebb 学习规则。1957 年，Frank Rosenblatt 首次提出并设计制作了著名的感知机（Perceptron），第一次从理论研究转入过程实现阶段，掀起了研究人工神经网络的高潮。虽然，从 20 世纪 60、70 年代，MIT 电子研究实验室的 Marvin Minsky 和 Seymour Dapret 就开始对感知机做深入的评判，并于 1969 年他们出版了“Perceptron”一书，对 Frank Rosenblatt 的感知机的抽象版本做了详细的数学分析，认为感知机基本上是一个不值得研究的领域，曾一度使神经网络的研究陷入低谷。但是，从 1982 年美国物理学家 Hopfield 提出 Hopfield 神经网络，以及 1986 年 D.E.Rumelhart 和 J. L. McClelland 提出一种利用误差反向传播训练算法的 BP（Back Propagation）神经网络开始，在世界范围内再次掀起了神经网络的研究热潮。今天，随着科学技术的迅猛发展，神经网络正以极大的魅力吸引着世界上众多专家、学者为之奋斗，在世界范围内再次掀起了神经网络的研究热潮。难怪有关国际权威人士评论指出，目前对神经网络的研究其重要意义不亚于第二次世界大战时对原子弹的研究。

人工神经网络特有的非线性适应性信息处理能力，克服了传统人工智能方法对于直觉，如模式、语音识别、非结构化信息处理方面的缺陷，使之在神经专家系统、模式识别、智能控制、组合优化、预测等领域得到成功应用。人工神经网络与其他传统方法相结合，将推动人工智能和信息处理技术不断发展。近年来，人工神经网络正向模拟人类认知的道路上更加深入发展，与模糊系统、遗传算法、进化机制等结合，形成计算智能，成为人工智能的一个重要方向，将在实际应用中得到发展。

使用神经网络的主要优点是能够自适应样本数据，当数据中有噪声、形变和非线性时，它也能够正常地工作，很容易继承现有的领域知识，使用灵活，能够处理来自多个资源和决策系统的数据；提供简单工具进行自动特征选取，产生有用的数据表示，可作为专家系统的

前端(预处理器)。此外,神经网络还能提供十分快的优化过程,尤其以硬件直接实现网络时,可以加速联机应用程序的运行速度。当然,过分夸大神经网络的应用能力也是不恰当的,毕竟它不是无所不能的。这就需要在实际工作中具体问题具体分析,合理选择。

基于神经网络的控制称为神经网络控制(NNC),简称神经控制(Neuro Control, NC)这一新词是在国际自控联杂志《自动化》(Automatica)1994年 No.11 首次使用的,最早源于1992年 H.Tolle 和 E.Ersu 的专著《Neuro Control》。基于神经网络智能模拟用于控制,是实现智能控制的一种重要形式,近年来获得了迅速发展。本章介绍神经网络的基本概念、基本结构和学习算法等。

1.1 神经网络的基本概念

1.1.1 生物神经元的结构与功能特点

神经生理学和神经解剖学证明了人的思维是由人脑完成的。神经元是组成人脑的最基本单元,它能够接收并处理信息,人脑大约由 $10^{11} \sim 10^{12}$ 个神经元组成,其中每个神经元约与 $10^4 \sim 10^5$ 个神经元通过突触连接,因此,人脑是一个复杂的信息并行加工处理巨系统。探索脑组织的结构、工作原理及信息处理的机制,是整个人类面临的一项挑战,也是整个自然科学的前沿领域。

1. 生物神经元的结构

生物神经元,也称为神经细胞,是构成神经系统的基本单元。生物神经元主要由细胞体、树突和轴突构成,其基本结构如图 1-1 所示。

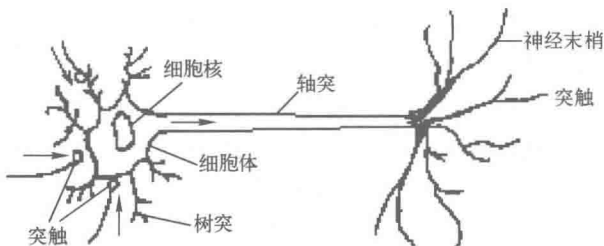


图 1-1 生物神经元结构

1) 细胞体

细胞体由细胞核、细胞质与细胞膜等组成。一般直径为 $5 \sim 100 \mu\text{m}$, 大小不等。细胞体是生物神经元的主体,它是生物神经元的新陈代谢中心,同时还负责接收并处理从其他神经元传递过来的信息。细胞体的内部是细胞核,外部是细胞膜,细胞膜外是许多外延的纤维,细胞膜内外有电位差,称为膜电位,膜外为正,膜内为负。

2) 轴突

轴突是由细胞体向外伸出的所有纤维中最长的一条分支。每个生物神经元只有一个轴突,长度最大可达 1m 以上,其作用相当于生物神经元的输出电缆,它通过尾部分出的许多神经末梢以及梢端的突触向其他生物神经元输出神经冲动。

轴突终端的突触,是生物神经元之间的连接接口,每一个生物神经元有 $10^4 \sim 10^5$ 个突触。一个生物神经元通过其轴突的神经末梢,经突触与另一生物神经元的树突连接,以实现信息的传递。

3) 树突

树突是由细胞体向外伸出的除轴突外的其他纤维分支, 长度一般均较短, 但分支很多。它相当于神经元的输入端, 用于接收从四面八方传来的神经冲动。

2. 生物神经元的功能特点

从生物控制论的观点来看, 作为控制和信息处理基本单元的生物神经元, 具有以下功能特点。

1) 时空整合功能

生物神经元对于不同时间通过同一突触传入的信息, 具有时间整合功能; 对于同一时间通过不同突触传入的信息, 具有空间整合功能。两种功能相互结合, 使生物神经元具有时空整合的输入信息处理功能。

2) 动态极化性

在每一种生物神经元中, 信息都是以预知的确定方向流动的, 即从生物神经元的接收信息部分(细胞体、树突)传到轴突的起始部分, 再传到轴突终端的突触, 最后再传给另一生物神经元。尽管不同的生物神经元在形状及功能上都有明显的不同, 但大多数生物神经元都是按这一方向进行信息流动的。

3) 兴奋与抑制状态

生物神经元具有两种常规工作状态, 即兴奋状态与抑制状态。所谓兴奋状态是指生物神经元对输入信息经整合后使细胞膜电位升高, 且超过了动作电位的阈值, 此时产生神经冲动并由轴突输出。抑制状态是指对输入信息整合后, 细胞膜电位值下降到低于动作电位的阈值, 从而导致无神经冲动输出。

4) 结构的可塑性

由于突触传递信息的特性是可变的, 也就是它随着神经冲动传递方式的变化, 传递作用强弱不同, 形成了生物神经元之间连接的柔性, 这种特性又称为生物神经元结构的可塑性。

5) 脉冲与电位信号的转换

突触界面具有脉冲与电位信号的转换功能。沿轴突传递的电脉冲是等幅的、离散的脉冲信号, 而细胞膜电位变化为连续的电位信号, 这两种信号是在突触接口进行变换的。

6) 突触时延和不应期

突触对信息的传递具有时延和不应期, 在相邻的两次输入之间需要一定的时间间隔, 在此期间, 无激励, 不传递信息, 这称为不应期。

7) 学习、遗忘和疲劳

由于生物神经元结构的可塑性, 突触的传递作用有增强、减弱和饱和三种情况。所以, 神经细胞也具有相应的学习、遗忘和疲劳效应(饱和效应)。

1.1.2 人工神经元模型

生物神经元经抽象化后, 可得到如图 1-2 所示的一种人工神经元模型, 它有三个基本要素。

1. 连接权

连接权对应于生物神经元的突触, 各个神经元之间的连接强度由连接权的权值表示, 权值为

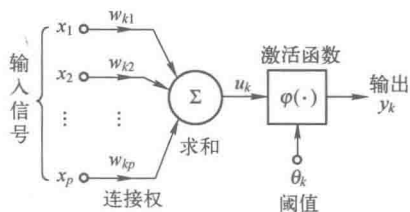


图 1-2 基本神经元模型

正表示激活, 为负表示抑制。

2. 求和单元

用于求取各输入信号的加权和 (线性组合)。

3. 激活函数

激活函数起非线性映射作用, 并将人工神经元输出幅度限制在一定范围内, 一般限制在 (0,1) 或 (-1,1) 之间。激活函数也称为传输函数。

此外, 还有一个阈值 θ_k (或偏值 $b_k = -\theta_k$)。

图 1-2 中的作用可分别以数学式表达出来, 即

$$u_k = \sum_{j=1}^p w_{kj} x_j, \quad y_k = \varphi(u_k - \theta_k)$$

式中, $x_1, x_2, \dots, x_p$ 为输入信号, 它相当于生物神经元的树突, 为人工神经元的输入信息; $w_{k1}, w_{k2}, \dots, w_{kp}$ 为神经元 k 的权值; u_k 为线性组合结果; θ_k 为阈值; $\varphi(\cdot)$ 为激活函数; y_k 为神经元 k 的输出, 它相当于生物神经元的轴突, 为人工神经元的输出信息。

在人工神经网络中, 为了简化起见, 通常将图 1-2 所示的基本神经元模型表示成图 1-3 (a) 或 (b) 所示的简化形式。

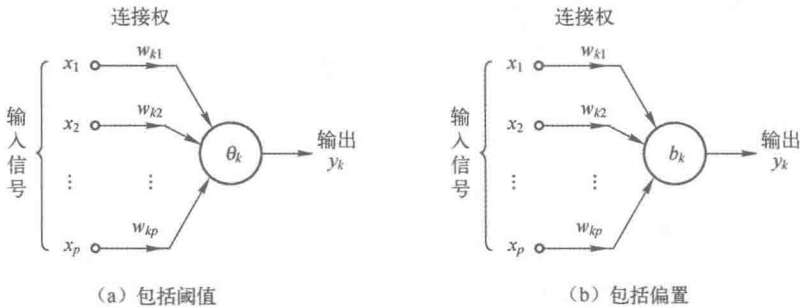


图 1-3 神经元模型的简化形式

图 1-3 (a) 和 (b) 中的输入与输出之间的关系, 可分别采用以下数学表达式进行描述

$$y_k = \varphi\left(\sum_{j=1}^p w_{kj} x_j - \theta_k\right) \quad \text{和} \quad y_k = \varphi\left(\sum_{j=1}^p w_{kj} x_j + b_k\right)$$

激活函数 $\varphi(\cdot)$ 一般采用以下几种形式。

1) 阶跃函数

函数表达式为

$$y = \varphi(x) = \begin{cases} 1 & (x \geq 0) \\ -1 & (x < 0) \end{cases}$$

2) 分段线性函数

函数表达式为

$$y = \varphi(x) = \begin{cases} 1 & (x \geq 1) \\ \frac{1}{2}(1+x) & (-1 < x < 1) \\ -1 & (x \leq -1) \end{cases}$$

3) Sigmoid 型函数

最常用的 Sigmoid 型函数为

$$\varphi(x) = \frac{1}{1 + \exp(-ax)}$$

式中, 参数 a 可控制其斜率。

Sigmoid 型函数也简称 S 型函数, 上式表示的是一种非对称 S 型函数。

另一种常用的 Sigmoid 型函数为双曲正切对称 S 型函数, 即

$$\varphi(x) = \tanh\left(\frac{1}{2}x\right) = \frac{1 - \exp(-x)}{1 + \exp(-x)}$$

这类函数具有平滑和渐近线, 并保持单调性。

1.1.3 神经网络的结构

人工神经网络 (Artificial Neural Networks, ANN) 是由大量人工神经元经广泛互连而组成的, 它可用来模拟脑神经系统的结构和功能。人工神经网络可以看成以人工神经元为节点, 用有向加权弧连接起来的有向图。在此有向图中, 人工神经元 (以下在不易引起混淆的情况下, 人工神经元简称神经元) 就是对生物神经元的模拟, 而有向加权弧则是轴突—突触—树突对的模拟。有向弧的权值表示相互连接的两个人工神经元间相互作用的强弱。

人工神经网络是生物神经网络的一种模拟和近似。它主要从两个方面进行模拟。一种是从生理结构和实现机理方面进行模拟, 它涉及生物学、生理学、心理学、物理及化学等许多基础科学。由于生物神经网络的结构和机理相当复杂, 现在距离完全认识它们还相差甚远。另外一种是从功能上加以模拟, 即尽量使得人工神经网络具有生物神经网络的某些功能特性, 如学习、识别、控制等功能。本书仅讨论后者, 从功能上来看, 人工神经网络 (以下简称神经网络或 NN) 根据连接方式主要分为两类。

1. 前馈型网络

前馈神经网络是整个神经网络体系中最常见的一种网络, 其网络中各个神经元接收前一级的输入, 并输出到下一级, 网络中没有反馈, 如图 1-4 所示。节点分为两类, 即输入单元和计算单元, 每一计算单元可有任意个输入, 但只有一个输出 (它可耦合到任意多个其他节点作为输入)。通常前馈网络可分为不同的层, 第 i 层的输入只与第 $i-1$ 层输出相连, 输入和输出节点与外界相连, 而其他中间层称为隐含层, 它们是一种强有力的学习系统, 其结构简单而易于编程。从系统的观点看, 前馈神经网络是一静态非线性映射, 通过简单非线性处理的复合映射可获得复杂的非线性处理能力。但从计算的观点看, 前馈神经网络并非是一种强有力的计算系统, 不具有丰富的动力学行为。大部分前馈神经网络是学习网络, 并不注意系统的动力学行为, 它们的分类能力和模式识别能力一般强于其他类型的神经网络。

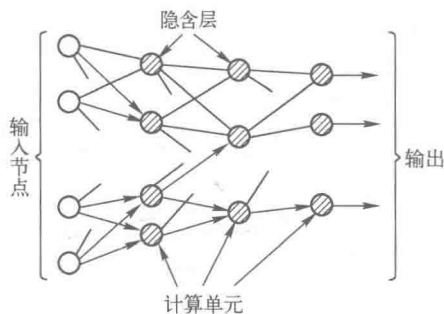


图 1-4 前馈网络

2. 反馈型网络

反馈神经网络又称为递归网络或回归网络。在反馈神经网络 (Feedback NN) 中, 输入

信号决定反馈系统的初始状态, 然后系统经过一系列状态转移后, 逐渐收敛于平衡状态。这样的平衡状态就是反馈网络经计算后输出的结果, 由此可见, 稳定性是反馈网络中最重要的问题之一。如果能找到网络的 Lyapunov 函数, 则能保证网络从任意的初始状态都能收敛到局部最小点。反馈神经网络中所有节点都是计算单元, 同时也可接收输入, 并向外界输出, 可画成一个无向图, 如图 1-5 (a) 所示, 其中每个连接弧都是双向的, 也可画成如图 1-5 (b) 所示的形式。若总单元数为 n , 则每一个节点有 $n-1$ 个输入和一个输出。

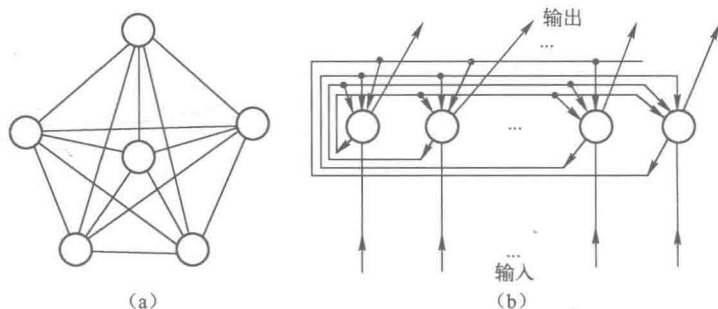


图 1-5 单层全连接反馈网络

1.1.4 神经网络的工作方式

神经网络的工作过程主要分为两个阶段: 第一阶段是学习期, 此时各计算单元状态不变, 各连接权上的权值可通过学习来修改; 第二阶段是工作期, 此时各连接权固定, 计算单元变化, 以达到某种稳定状态。

从作用效果看, 前馈网络主要是函数映射, 可用于模式识别和函数逼近。反馈网络按对能量函数的极小点的利用来分类有两种: 第一类是能量函数的所有极小点都起作用, 这一类主要用作各种联想存储器; 第二类只利用全局极小点, 它主要用于求解最优化问题。

1.1.5 神经网络的学习

1. 学习方式

通过向环境学习获取知识并改进自身性能是神经网络的一个重要特点, 在一般情况下,

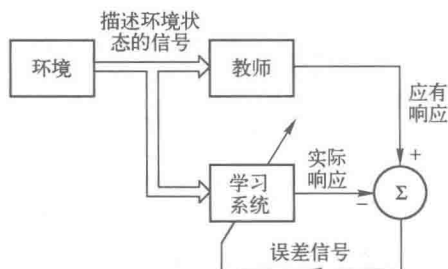


图 1-6 有监督学习框图

性能的改善是按某种预定的度量调节自身参数 (如权值) 随时间逐步达到的, 学习方式 (按环境所供信息的多少分) 有以下三种。

1) 有监督学习 (有教师学习)

有监督学习方式需要外界存在一个“教师”, 它可对一组给定输入提供应有的输出结果 (正确答案), 这组已知的输入/输出数据称为训练样本集。学习系统可根据已知输出与实际输出之间的差值 (误差信号) 来调节系统参数, 如图 1-6 所示。

在有监督学习当中, 学习规则由一组描述网络行为的训练集给出:

$$\{\mathbf{x}^{(1)}, \mathbf{t}^{(1)}\}, \{\mathbf{x}^{(2)}, \mathbf{t}^{(2)}\}, \dots, \{\mathbf{x}^p, \mathbf{t}^p\}, \dots, \{\mathbf{x}^N, \mathbf{t}^N\}$$

其中, $\mathbf{x}^p$ 为网络的第 p 个输入数据向量; $\mathbf{t}^p$ 为对应 $\mathbf{x}^p$ 的目标输出向量; N 为训练集中的样本数。

当输入作用到网络时, 网络的实际输出与目标输出相比较, 然后学习规则调整网络的权值和阈值, 从而使网络的实际输出越来越接近于目标输出。

2) 无监督学习 (无教师学习)

无监督学习时不存在外部教师, 学习系统完全按照环境所提供数据的某些统计规律来调节自身参数或结构 (这是一种自组织过程), 以表示外部输入的某种固有特性 (如聚类, 或某种统计上的分布特征), 如图 1-7 所示。在无监督学习中, 仅仅根据网络的输入调整网络的权值和阈值, 它没有目标输出。乍一看, 这种学习似乎并不可行: 不知道网络的目的是什么, 还能够训练网络吗? 实际上, 大多数这种类型的算法都是要完成某种聚类操作, 学会将输入模式分为有限的几种类型。这种功能特别适合于诸如向量量化等应用问题。

3) 强化学习 (或再励学习)

这种学习介于上述两种情况之间, 外部环境对系统输出结果只给出评价 (奖或罚) 而不是给出正确答案, 学习系统通过强化那些受奖励的动作来改善自身性能, 如图 1-8 所示。强化学习与有监督的学习类似, 只是它不像有监督的学习一样为每一个输入提供相应的目标输出, 而是仅仅给出一个级别。这个级别 (或评分) 是对网络在某些输入序列上的性能测度。当前这种类型的学习要比有监督的学习少见。它最为适合控制系统应用领域。



图 1-7 无监督学习框图

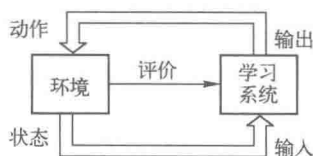


图 1-8 强化学习框图

2. 学习算法

1) δ 学习规则 (误差纠正规则)

若 $y_i(k)$ 为输入 $\mathbf{x}(k)$ 时神经元 i 在 k 时刻的实际输出, $t_i(k)$ 表示相应的期望输出, 则误差信号可写为

$$e_i(k) = t_i(k) - y_i(k)$$

误差纠正学习的最终目的是使某一基于 $e_i(k)$ 的目标函数达最小, 以使网络中每一输出单元的实际输出在某种统计意义上最逼近期望输出。一旦选定了目标函数形式, 误差纠正学习就成为一个典型的最优化问题。最常用的目标函数是均方误差判据, 定义为

$$J = E \left\{ \frac{1}{2} \sum_{i=1}^L (t_i - y_i)^2 \right\}$$

式中, E 是统计期望算子; L 为网络输出数。

上式的前提是被学习的过程是宽而平稳的, 具体方法可用最陡梯度下降法。直接用 J 作为目标函数时, 需要知道整个过程的统计特性, 为解决这一困难, 用 J 在时刻 k 的瞬时值 $J(k)$ 代替 J , 即

$$J(k) = \frac{1}{2} \sum_{i=1}^L (t_i - y_i)^2 = \frac{1}{2} \sum_{i=1}^L e_i^2(k)$$

问题变为求 $J(k)$ 对权值 w_{ij} 的极小值, 根据最陡梯度下降法可得:

$$\Delta w_{ij}(k) = \eta \cdot \delta_i(k) \cdot x_j(k) = \eta \cdot e_i(k) \cdot f'(W_i \mathbf{x}) \cdot x_j(k)$$

式中, η 为学习速率或步长 ($0 < \eta \leq 1$); $f(\cdot)$ 为激活函数。这就是通常说的误差纠正学习规则 (或称为 δ 规则), 用于控制每次误差修正值。它是基于使输出方差最小的思想而建立的。

2) Hebb 学习规则

神经心理学家 Hebb 提出的学习规则可归结为“当某一突触 (连接) 两端的神经元的激活同步 (同为激活或同为抑制) 时, 该连接的强度应增加, 反之则应减弱”, 用数学方式可描述为

$$\Delta w_{ij}(k) = F(y_i(k), x_j(k))$$

式中, $y_i(k)$, $x_j(k)$ 分别为 w_{ij} 两端神经元的状态, 其中最常用的一种情况为

$$\Delta w_{ij}(k) = \eta \cdot y_i(k) \cdot x_j(k)$$

式中, η 为学习速率。

由于 $w_{ij}(k)$ 与 $y_i(k)$ 、 $x_j(k)$ 的相关成比例, 故有时称之为相关学习规则。上式定义的 Hebb 规则实际上是一种无监督的学习规则, 它不需要关于目标输出的任何相关信息。

原始的 Hebb 学习规则对权值矩阵的取值未做任何限制, 因而学习后权值可取任意值。为了克服这一弊病, 在 Hebb 学习规则的基础上增加一个衰减项, 即

$$\Delta w_{ij}(k) = \eta \cdot y_i(k) \cdot x_j(k) - d_r * w_{ij}(k)$$

衰减项的加入能够增加网络学习的“记忆”功能, 并且能够有效地对权值的取值加以限制。衰减系数 d_r 的取值在 $[0, 1]$ 之间。当取 0 时, 就变成原始的 Hebb 学习规则。

另外, Hebb 学习规则还可以采用有监督的学习, 对于有监督学习的 Hebb 学习规则而言, 是将目标输出代替实际输出。由此, 算法被告知的就是网络应该做什么, 而不是网络当前正在做什么, 可描述为

$$\Delta w_{ij}(k) = \eta \cdot t_i(k) \cdot x_j(k)$$

3) 竞争 (Competitive) 学习规则

顾名思义, 在竞争学习时网络各输出单元互相竞争, 最后达到只有一个最强者激活。最常见的一种情况是输出神经元之间有侧向抑制性连接, 如图 1-9 所示。这样众多输出单元中如有某一单元较强, 则它将获胜并抑制其他单元, 最后只有比较强者处于激活状态。最常用的竞争学习规则有以下三种。

$$\text{Kohonen 规则: } \Delta w_{ij}(k) = \begin{cases} \eta(x_j - w_{ij}), & \text{若神经元 } j \text{ 竞争获胜} \\ 0, & \text{若神经元 } j \text{ 竞争失败} \end{cases}$$

$$\text{Instar 规则: } \Delta w_{ij}(k) = \begin{cases} \eta y_i(x_j - w_{ij}), & \text{若神经元 } j \text{ 竞争获胜} \\ 0, & \text{若神经元 } j \text{ 竞争失败} \end{cases}$$

$$\text{Outstar 规则: } \Delta w_{ij}(k) = \begin{cases} \eta(y_i - w_{ij})/x_j, & \text{若神经元 } j \text{ 竞争获胜} \\ 0, & \text{若神经元 } j \text{ 竞争失败} \end{cases}$$

3. 学习与自适应

当学习系统所处环境平稳时 (统计特征不随时间变化), 从理论上说通过监督学习可以学到环境的统计特征, 这些统计特征可被学习系统 (神经网络) 作为经验记住。如果环境是非平稳的 (统计特征随时间变化), 通常的监督学习没有能力跟踪这种变化, 为解决此问题需要网络有一定的自适应能力, 此时对每一个不同输入都作为一个新的例子对待, 其工作过程如图 1-10 所示。此时模型 (如 NN) 被当作一个预测器, 基于前一时刻输出 $x(k-1)$ 和模型在 $k-1$ 时刻的参数, 它估计出 k 时刻的输出 $\hat{x}(k)$, $\hat{x}(k)$ 与实际值 $x(k)$ (作为应有的正确答案) 比