

分类号：TP273

UDC：62

密级：不保密

学校代码：11065

青 岛 大 学

硕士学位论文

基于 SVM 和 K-均值聚类的 RBF 神经网络短时交通流预测

管硕

指 导 教 师

高军伟 教授

学科专业名称

控制科学与工程

论文答辩日期

2015 年 6 月 6 日



## 摘要

交通出行在日益繁忙的工作生活中愈发显得重要，如何解决道路拥挤，减少交通事故发生率，已经成为世界各国亟待解决的复杂难题。短时交通流的预测作为应对上述问题的方案，因为其具有实时准确等特点，已经被当作智能交通系统（Intelligent Transport Systems, ITS）的主要组成部分。

本文首先介绍了 K-均值聚类算法的基本理论，它的特点是简单快速，同时应用广泛。在处理密集样本的时候，同时样本中的类彼此线性可分，那么运用 K-均值聚类算法所得到的聚类效果就会很出色。在 RBF 神经网络的支持下，对短时交通流进行预测，得到了比较理想的数据。

支持向量机（Support Vector Machine, SVM）作为机器学习方法中的佼佼者，它能够出色的解决许多算法中的实际问题，比如 K-均值聚类算法很容易陷入局部极小值。通过引入支持向量机，我们能够在短时交通流的预测上，得到更加精确的结果。实验结果说明，利用基于支持向量机的 K-均值聚类算法，RBF 神经网络能够有效的预测短时交通流，证明该模型是有效的。

**关键词：**短时交通流预测；K-均值聚类算法；支持向量机；RBF 神经网络



## **Abstract**

Traffic travel in an increasingly busy work life all the more important, how to solve road congestion, reduce accident rates, the world has become a complex problem to be solved. Short-term traffic flow forecasting as a response to the above problem solution because it has the characteristics of real-time and accurate, has been used as a major component of the intelligent transportation system.

This paper introduces the basic theory of K- means clustering algorithm, which is characterized by simple and fast, while widely used. Time-intensive sample handling, and sample each other linearly separable classes, then use the clustering effect K- means clustering algorithm obtained will be very good. In support of RBF neural network, the forecasting of traffic flow, has been ideal data.

Support vector machine as the machine learning methods in the crowd, it can solve many excellent algorithms practical problems, such as K- means clustering algorithm is very easy to fall into local minima. By introducing support vector machines, we are able to predict traffic flow on, to get more accurate results. Experimental results demonstrate that the use of SVM K- means clustering algorithm, RBF neural network can effectively predict the traffic flow, show that the model is valid.

**Key words: short-term traffic flow forecasting; K- means clustering algorithm; Support Vector Machines; RBF neural network**



# 目 录

第一章 绪论	1
1.1 课题的研究背景及其意义	1
1.2 智能交通系统	2
1.2.1 先进的出行者信息系统 (ATIS)	3
1.2.2 先进的交通管理系统(ATMS)	4
1.3 交通流预测的研究与发展	5
1.3.1 交通流预测国内外研究动态	5
1.4 短时交通流预测的研究与发展	7
1.4.1 卡尔曼滤波	8
1.4.2 神经网络模型	9
1.5 论文研究的内容及目标	10
第二章 人工神经网络理论	11
2.1 神经网络的基本概念	11
2.1.1 生物神经元的结构与功能	11
2.1.2 神经网络的特点	12
2.1.3 常用的神经网络模型	12
2.2 BP 神经网络	13
2.2.1 BP 神经元结构模型	14
2.2.2 BP 神经网络拓扑结构	14
2.2.3 BP 算法的基本描述	15
2.2.4 BP 神经网络的优缺点	17
2.3 RBF 神经网络	18
2.3.1 RBF 神经网络结构	18
2.3.2 RBF 神经网络的学习算法	20
2.4 本章小结	21
第三章 聚类分析与 K-均值聚类算法	23
3.1 聚类分析与算法	23
3.1.1 聚类的定义	23
3.1.2 聚类分析中不同的数据类型	23
3.1.3 相似性度量	27
3.2 K-均值聚类算法	28
3.2.1 K-均值聚类算法流程和运行过程	28
3.2.2 K-均值聚类算法的优缺点分析	29

3.3 基于支持向量机的 K-均值聚类算法 .....	30
3.3.1 支持向量机 .....	31
3.3.2 最优分类面 .....	31
3.3.3 支持向量机的分类 .....	32
3.4 最小二乘法 .....	34
3.4.1 最小二乘法的概述 .....	34
3.4.2 最小二乘法的原理 .....	34
3.5 本章小结 .....	35
<b>第四章 基于支持向量机和 K-均值聚类的 RBF 短时交通流预测仿真实验 .....</b>	<b>37</b>
4.1 基于 K-均值聚类的 RBF 神经网络的学习算法 .....	37
4.1.1 K-均值聚类算法确定基函数中心的方法 .....	37
4.1.2 采用最小二乘法对 RBF 网络权值的处理 .....	37
4.2 交通流量的数据来源 .....	38
4.3 计算机仿真 .....	40
4.4 基于支持向量机和 K-均值聚类的 RBF 短时交通流预测 .....	42
4.5 本章小结 .....	43
<b>结论与展望 .....</b>	<b>45</b>
<b>参考文献 .....</b>	<b>47</b>
<b>攻读学位期间的研究成果 .....</b>	<b>51</b>
<b>致 谢 .....</b>	<b>53</b>
<b>学位论文独创性声明 .....</b>	<b>55</b>
<b>学位论文知识产权权属声明 .....</b>	<b>55</b>



## 第一章 绪论

### 1.1 课题的研究背景及其意义

随着社会的发展进步，人民生活水平的提高和生活节奏的加快，汽车保有量以及交通人流量的逐步攀升已经给城市带来了极大的拥堵问题。城市在不停的扩张，农村改造以及城市化建设正在如火如荼的进行中，大型城市上下班高峰时期堵车已经成为常态。城市道路拥堵的原因很多，例如机动车数量的激增，道路规划设计不合理甚至是驾驶员的不良驾驶习惯等等，都是造成交通拥堵的原因。

从 1994 年开始，我国陆续有城市对机动车的购买进行了限制，上海是第一个出台汽车限购令的城市，人们不能像以前一样没有限制的就可以给车辆上牌照了，客车的配额需要进行拍卖，同时针对私家车牌照的发放，相关部门实行不公开同时具有一定底价的拍卖政策。2010 年，北京市也发布了《北京市小客车数量调控暂行规定》条例，实施限购，2013 年 11 月，北京市再次发出通知，规定了 2014 年到 2017 年四年内的配额指标从 24 万辆减少到 15 万辆。

2011 年 7 月，贵阳市政府颁布了汽车限购令。

2012 年 7 月，广州市在前者的经验基础上出台了一个折中的处理办法，将限购规则划分为两部分，一种是有价竞拍，另一种是无偿摇号。

2013 年 6 月，石家庄市针对省会机动车数量激增的现实情况，要求限制每个家庭只允许购买两辆私家车，不允许购买第三辆。

2013 年 12 月，天津市对小客车的增加量进行管理，由相关部门分配后发放，小客车分配指标每年是 10 万辆，每月定额发放。

2014 年 3 月，杭州市也对小客车的增量配额进行管理，获取增量配额的方式有两种，一种是公开摇号，另一种就是竞价获取。

2014 年 12 月，深圳市开始实施汽车限购政策，有效期目前为五年，每年指标暂定为 10 万个，按月分配。

从上面的数据我们不难看出，以后会有越来越多的城市采取限购令来缓解日益严峻的交通问题，同时驾照也越来越难考，以前机动车驾驶证的考试一般分为科目一，科目二以及科目三，现在越来越多的城市增加了科目四和科目五，目的就是为了加强机动车驾驶员的安全文明驾驶常识，避免过多新手驾驶员对于交通规则理解不够深刻，减少交通事故的发生率。

随着城市化进程的推进，城市人口逐年攀升，目前城市民众对于交通出行需求增长过高，原有的交通资源已经逐渐无法满足市民们的出行要求，每天早晚班高峰

时段的拥堵成为了每个上班族的噩梦，城市交通系统网络承受着沉重的负担，交通系统的管理和信息服务无法彻底解决交通拥堵这一难题，仅靠加快城市基础设施建设已经无法满足人们对于道路交通的需求，交通堵塞、交通事故以及交通污染困扰着每一个城市公民，最近几年的雾霾现象频发，许多学者更是将其归结于汽车尾气的污染，因此高效的交通出行，畅通的道路不仅改善了人们的出行质量，更能减少汽车尾气的排放，减少二氧化硫和氮氧化合物的排放量，最终有效防止雾霾的发生。

仅仅依靠城市规划和基础设施建设的完善来解决交通运输领域的难题是远远不够的，在面对当今社会日益严重的交通问题时，许多发达国家对于城市的交通进行了科学的规划研究以及适当的引导疏通，来改善道路交通所面对的各种问题，提高道路交通效率。虽然这两项措施效果明显，同时也是政府在建设和管理城市交通运输网络中不可缺少的重要构成，但是依靠它们还是不能够经济、高效地处理交通拥挤和交通安全等问题<sup>[1]</sup>。

在现有的硬件设施资源下，通过加强道路基础建设、完善交通信号规划以及增加交警出勤次数，结合城市交通理论，合理运用智能交通系统（Intelligent Transportation Systems, ITS）<sup>[2,3]</sup>全面有效地解决交通运输领域问题，进而全方位的提升道路通行能力，改善城市的交通出行情况。

## 1.2 智能交通系统

1994年在法国巴黎，第一次世界智能交通大会召开，此次会议上，智能交通系统这个概念被学者们第一次提了出来，人们通过使用各种现代化技术对以前的交通系统网络进行改良，使得交通系统不是那么被动笨拙，保证对于道路车流量调节和道路通行状态更加智能，提高城市交通系统网络整体的运行效率。ITS技术能够让城市交通运输的管理者、运营者以及个体出行者及时的掌握道路信息，根据所得的信息相互协调，然后得出更合适的选择。通过ITS系统的建设和执行，我们就能够减少交通事故的数目，降低对环境的污染，减少运输成本，提高运输效率，这样一来，我们就能够营造起一个安全、环保、低消耗、高效率的智能型综合运输体系<sup>[4]</sup>。在国外，一些学者已经把ITS技术引进到航空运输以及铁路运输中，不仅能够给出行者提供快捷准确的出行信息，也能够给物流业带来足够全面详尽的信息服务。

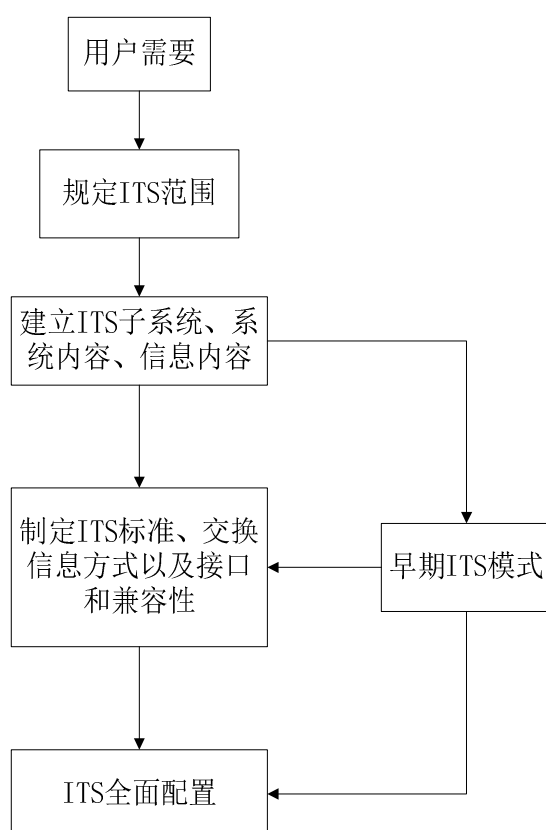


图 1.1 ITS 系统结构的标准化及其配置过程

面对日益严峻的交通问题，我国政府投入巨大的财力物力及高新科技技术，积极参与建设改进智能交通系统，道路监控系统日益完善，道路规划日趋合理，许多城市还初步建立了交通管理系统和交通信息服务平台，为市民传达实时的道路情况，大大提高了民众的出行效率以及交通安全。青岛市在前几年将无线通信技术、全球定位技术(GPS)以及互联网平台等一切可利用的资源，例如运用全球定位技术对任意一辆公交车定位，然后通过车上的软硬件将实时数据传达到控制中心，最后控制中心汇总处理数据后发布到网络上，这样就构建成了先进高效的智能公交调度。起初先从几条公交线路开始试验，利用无线通信技术以及相关硬件及时采集车辆到站信息，并且通过各种平台渠道为市民提供信息查询，比如手机端的 app “青岛公交查询”，人们就能通过手机或者平板电脑，轻轻松松查到许多线路公交的实时位置，合理安排自己的出行时间和出行路线，省去不必要的麻烦，为广大市民提供交通便利。公交公司也能够合理安排发车，适当增加或减少某公交线路的车次，这样既能避免出现公交车空跑浪费资源的情况，又能减少环境污染。

### 1.2.1 先进的出行者信息系统（ATIS）

先进的出行者信息系统（Advanced traveler Information System）作为智能交通系统的一个子系统，凭借其在智能交通系统中的出色管理，逐渐被世界各国重视发展，

并且被当作道路交通建设中重要的系统体系来进行建设。ATIS 可以给用户提供足够详细准确的实时路况信息，让用户可以选择合适的出行路线而不至于陷入拥堵，用户以及用户的车辆都装配有交通信息系统终端，车辆上的终端能够上传详细的车辆行驶情况，用户可以随时用文字向交通信息中心传达路况信息，交通信息中心也能够及时的反馈相应信息，这样就实现了双向信息传递。ATIS 系统即为智能交通系统的核心以及基础。

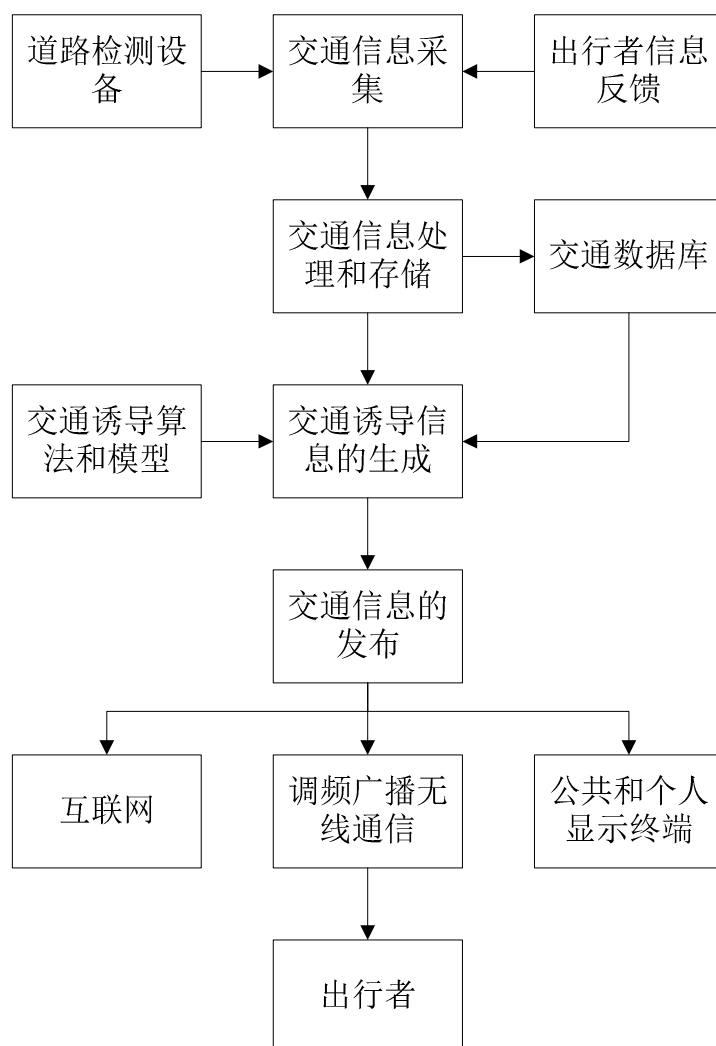


图 1.2 ATIS 系统结构

## 1.2.2 先进的交通管理系统(ATMS)

在先进的交通管理系统中，车辆不像以前一样，单纯的作为系统中被管理的某个个体，通过加载相关硬件以及软件的配合，连接到网络，形成了一个软硬件结合的局部管理系统。与以往的理论研究不同，通过先进的交通监控系统、配有微型计算机支配的交通运输设备以及科学的道路信息收集体系所构成的先进的交通管理系统，它将传统意义上的车辆作为自身的重要组成部分，车辆就是一个交通探测器，

能够实时收集和传送它本身甚至周围环境的数据，最终协助交通管理系统做出比较合理的判断与分析。

## 1.3 交通流预测的研究与发展

### 1.3.1 交通流预测国内外研究动态

交通流预测按照预测周期的长度，一般情况下被划分为两类<sup>[5]</sup>：短期预测（Short-term Forecasting）和中长期预测（Mid—long—term Forecasting）。通常情况下，短期预测的预测时间间隔为 5 分钟至 30 分钟以内，中期预测的预测时间间隔为 30 分钟到数个小时之间，长期预测的时间就比较长了，一般是时间跨度在 1 天以上。近几十年来，交通流预测在各国专家和学者的努力下，在不同的领域和学科中研究出了将近 300 多种预测模型和方法，在这些模型和方法中，有部分预测方法在中长期交通流预测的使用，得到了比较令人满意的预测结果<sup>[6]</sup>。短时交通流预测理论在 ATIS 和 ATMS 的推动下，逐渐受到学者们的重视，学者们把其他领域应用比较熟练的预测模型运用在了短时交通流预测上面，取得了大量具有科学意义的研究成果，得到了近百种预测方法<sup>[7]</sup>。

在国外，1974 年，Nicholson 和 Swamn 采用谱分析法进行交通量的预测<sup>[8]</sup>。1979 年，Ahmaed S.A. 和 Cook A.R. 首次将时间序列模型成功的应用在了交通流预测领域<sup>[9]</sup>。1981 年，Stephanedes 把历史平均模型运用在城市交通控制系统（UTCS-urban traffic control system）里面。1984 年，Stephanedes 和 Okutani 把 ARIMA（自回归指数移动平均）模型运用到 UTCS 里面<sup>[10]</sup>。1993 年，Hobeika 和 Kim 又将 ARIM 模型运用到高速公路交通流量预测模型里面，取得了不错的效果。1960 年，Kalman 提出了卡尔曼滤波理论，卡尔曼滤波是采用状态方程和观测方程组成的线性随机系统的状态空间模型来描述滤波器<sup>[11,12]</sup>；1984 年，Stephanedes 和 Okutani 根据此理论研究出用于交通流预测的卡尔曼滤波模型<sup>[13]</sup>。1991 年，Nihan 和 Davis 建立非参数回归预测模型，该模型在某些条件下的预测精度要比时间序列模型更加优秀<sup>[14]</sup>。1993 年，Brian L. Smith 和 Michael J. Demestky 利用神经网络来对交通流量进行预测，所得结果精度较高并且没有时滞现象<sup>[15]</sup>。1996 年，Maschavan der Voort 等人运用神经网络和 ARIMA 时间序列模型相结合来预测交通流量，使得 ARIMA 时间序列模型具有了更广泛的适应性和可移植性<sup>[16]</sup>。1997 年，Corinne Ledoux 建立了基于神经网络的交通流量预测模型，取得了较好的预测效果<sup>[17]</sup>。

在国内，针对城市交通日益拥堵的现状，很多学者也在进行对交通流量预测的研究。根据卡尔曼滤波选用了比较灵活可靠以及适应范围广阔的特性，杨兆升提出了利用 Kalman 滤波技术预测交通流量的方法<sup>[18]</sup>，验证了当运用卡尔曼滤波时，交通流量预测模型上就可以达到比较高的预测精度。根据神经网络所具备的能够识别

复杂非线性系统的特性，许多学者提出了利用神经网络来提高交通流预测准确性的方法<sup>[19-22]</sup>。

交通指数作为交通拥堵指数或者交通运行指数（Traffic Performance Index，即“TPU”）的缩写，是近几年来学者们提出的一种综合性指标，主要是将道路交通网络的总体运行状况进行量化后进行一定的分析和评估。例如，青岛市城市道路上下坡以及纵横交错的小路偏多，智能交通基础设施建设也比较薄弱，针对这两方面的特点青岛市在交通流量数据采集方法上运用了固定点的办法，与北京、上海等城市主要通过获取浮动车（例如出租车等）的 GPS 数据（车速、位置）来计算拥堵里程比、通行速度等路网交通流指标来定义各自城市的交通指数不同，克服了在严重拥堵或偏远的道路或者恶劣天气情况下出租车样本数量不足的缺点，青岛市的方案从根本上解决了浮动车样本不足的问题。交通指数取值范围是从 0 到 10，我们将它分成五个阶段，第一个阶段是严重拥堵阶段，用红色表示，交通指数为 8 到 10，这时一般不建议驾驶员开车上路，除非有紧急事件；第二个阶段是中等拥堵阶段，用橙色表示，交通指数为 6 到 8；第三个阶段是轻度拥堵阶段，用黄色表示，交通指数为 4 到 6，驾驶员可以上路，但是尽量选择自己熟悉的行车数量较少的路段，必要时可以选择绕路；第四个阶段是基本畅通阶段，交通指数为 2 到 4，用浅绿色表示，驾驶员可以放心上路；最后一个阶段是畅通阶段，交通指数为 0 到 2，用绿色表示，这个阶段道路上虽然车辆很少，但是很容易造成驾驶员行驶过快，引起不必要的交通事故。交通指数越大时，道路就越拥堵，例如当交通指数的数值属于“畅通”这一阶段时，车辆可以按照安全行驶速度自由行驶；当交通指数的数值属于“严重拥堵”这一阶段时，行车时间将会增加，相同长度路径的行车时间可能是畅通阶段的两三倍甚至更长，极容易造成驾驶员的疲劳驾驶状态，因此而产生安全隐患；当交通指数达到顶峰时，市区主干道一般会处在极其拥堵混乱的状态，这时建议民众选择等待后出行，避免在路上浪费太多时间。

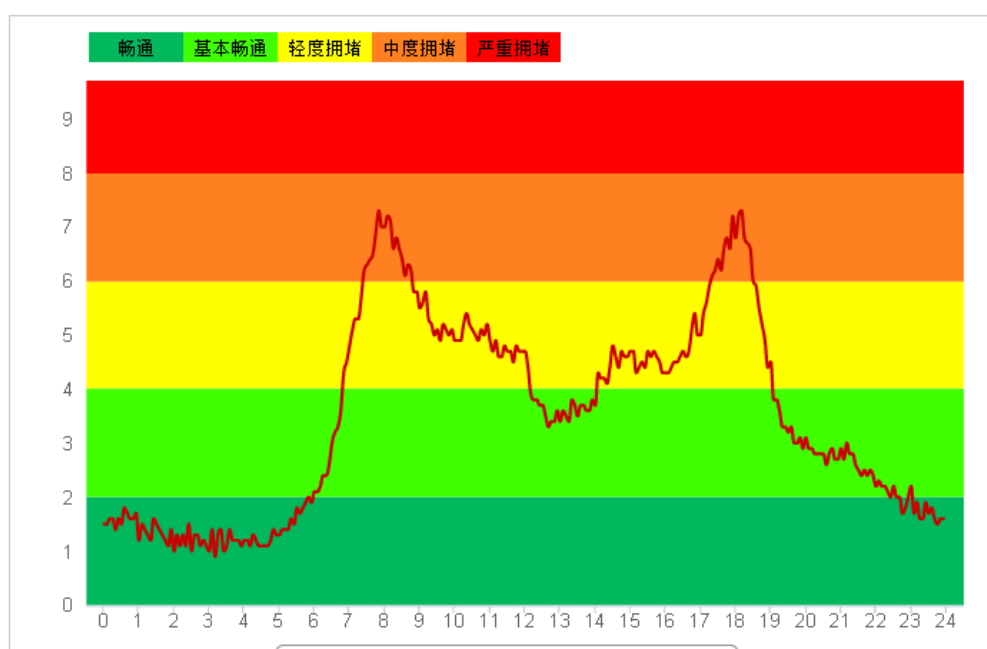


图 1.3 某天 24 小时交通指数趋势图

通过对历史数据统计分析、出行调查和现场验证，划分出对应畅通、基本畅通、轻度拥堵、中度拥堵、严重拥堵五种拥堵程度等级。

交通指数	(0, 2]	(2, 4]	(4, 6]	(6, 8]	(8, 10]
交通状态	畅通	基本畅通	轻度拥堵	中等拥堵	严重拥堵

图 1.4 交通指数阶段划分图

指数对应的交通拥堵现象如下所示：

畅通：交通运行状况很好，城市各个路段以及主干道车流量很少，车速快。；

基本畅通：交通运行状况较好，城市各个路段以及绝大多数主干道车流量偏少，车速快，少数主干道可能存在压车现象；

轻度拥堵：交通运行状况一般，但是城市整体路网运行平稳，局部路段会出现拥堵现象；

中度拥堵：交通运行状况较差，城市各个路段以及部分主干道车流量大，大部分主干道以及部分路段出现拥堵现象；

严重堵塞：交通运行状况很差，基本所有路段和全部主干道车流量很大，车速低，许多路段出现拥堵现象；

## 1.4 短时交通流预测的研究与发展

短时交通流的预测由于道路交通状况的时变性、非线性、不确定性以及复杂性等特点，同时因为有人参与、天气变化以及各种突发状况，很难进行准确预测<sup>[23,24]</sup>。因此研究短时交通流预测的目的，主要是对交通流的实时控制以及诱导。由于控制和引导的变化通常来说很快，交通诱导对于预测实时性的要求很高，预测时间间隔

一定要和控制与诱导的周期应该保持一致,所以短时交通流预测对于预测时间控制,一般来说会比较严格,通常情况下在 5-15 分钟以内<sup>[25]</sup>。当预测周期变小以后,交通流的变化情况就会随之变得复杂,在外界干扰的情况下会呈现出明显的不规律性,这就让短期交通流预测变得愈发困难。

国外的智能交通基础设施建设发展比较早,依靠他们先进的科学技术以及管理方法,很早就研究及探索智能交通系统,累积了很多经验。虽然国内在智能交通系统的研究方面,与国外相比,起步稍晚一些,但是有很多的现有方法和技术基础的前提下,整体研究和发展速度还是比较快的。

#### 1.4.1 卡尔曼滤波

Kalman 滤波是由 Kalman 在 1960 年提出的<sup>[26]</sup>,是现代控制理论中十分重要的组成部分。卡尔曼滤波采用的状态空间模型,是根据状态和观测方程构成的随机线性系统,运用递推算法运算的滤波器模型。在灵活的递推状态空间模型的支持下,卡尔曼滤波适用范围非常广泛,它既能够处理平稳数据,同时也能用在非平稳数据的处理上。当状态变量做出不同的假设时,可以处理不同的问题,与此同时还能够降低计算量。

卡尔曼滤波线性离散时间系统的状态空间表达式为:

$$\begin{cases} x(k+1) = \Phi(k+1, k)x(k) + \Gamma(k)w(k) \\ y(k) = H(k)x(k) + v(k) \end{cases} \quad 1-(1)$$

上式中,  $x$  是  $n \times 1$  维的状态变量,  $\Phi(k+1, k)$  是  $n \times n$  维的状态转移矩阵,  $\Gamma(k)$  是  $n \times r$  维的系统噪声转移矩阵,  $w(k)$  是  $r \times 1$  维的系统噪声,  $y$  是  $m \times 1$  维的观测向量,  $H(k)$  是  $m \times n$  维的观测矩阵,  $v(k)$  是  $m \times 1$  维的观测噪声。状态方程的系统噪声  $w(k)$  和观测噪声  $v(k)$ , 都是白噪声序列, 并且他们的均值是零。

系统噪声  $w(k)$  和观测噪声  $v(k)$  是相互独立的, 它们的均值也是零, 作为高斯白噪声序列, 它们的统计特性如下:

$$\begin{cases} \text{Cov}\{w(k), w(j)\} = \begin{cases} Q(k) & k = j \\ 0 & k \neq j \end{cases} \\ E\{w(k)\} = 0 \\ E\{w(k)v^T(j)\} = 0 \\ E\{v(k)\} = 0 \\ \text{Cov}\{v(k), v(j)\} = \begin{cases} R(k) & k = j \\ 0 & k \neq j \end{cases} \end{cases} \quad 1-(2)$$

式中,  $E$  表示数学期望,  $Q(k)$  是过程噪声  $w(k)$  的协方差矩阵, 且为对称半正定矩阵,  $R(k)$  是观测噪声  $v(k)$  的协方差矩阵, 且为对称正定矩阵。



初始条件满足：

$$\begin{cases} E\{x(0)\} = x_0 \\ E\{[x(0) - x_0][x(0) - x_0]^T\} = P_0 \end{cases} \quad 1-(3)$$

卡尔曼滤波一般情况下是按照线性无偏最小均方差进行估计计算，其准则：

$$E\{[x(k+1) - \bar{x}(k+1)]^T [x(k+1) - \bar{x}(k+1)]\} = \min \quad 1-(4)$$

卡尔曼滤波递推方程如下：

$$\begin{cases} \bar{x}(k+1|k) = \Phi(k+1, k)\bar{x}(k) \\ P(k+1|k) = \Phi(k+1, k)P(k)\Phi^T(k+1, k) + \Gamma(k)Q(k)\Gamma^T(k) \\ G(k+1) = P(k+1|k)H^T(k+1)[H(k+1)P(k+1|k)H^T(k+1) + R(k)]^{-1} \\ \bar{x}(k+1) = \bar{x}(k+1, k) + G(k+1)[y(k+1) - H(k+1)\bar{x}(k+1|k)] \\ P(k+1) = [I - G(k+1)H(k+1)]P(k+1|k) \end{cases} \quad 1-(5)$$

式中， $\bar{x}(k+1|k)$  是状态估计值， $P(k+1|k)$  是状态估计误差方差， $G(k+1)$  是卡尔曼增益， $\bar{x}(k+1)$  是状态预测值， $P(k+1)$  是状态预测误差方差。

相比于国内，国外学者们对于卡尔曼滤波研究进行的比较早。Iwao Okutani 在卡尔曼滤波理论的基础上，建立了交通流预测模型，得到了比较高的预测精度<sup>[27]</sup>。在国内，学者聂佩林、杨兆升等人，基于卡尔曼滤波提出了建立预测模型的方法，这种模型的预测数据效果令人满意<sup>[28,29]</sup>。

### 1.4.2 神经网络模型

1993 年，Vythoulkas PC<sup>[30]</sup>在研究交通流预测时，第一次运用神经网络来预测结果。Dochy Thierry 等把神经网络用来实现交通流预测时，既可以保证结果具有较高精度，同时也没有时滞现象的发生<sup>[31]</sup>。在国内，张晓利引进小波神经网络，然后对交通流进行预测并且分析了小波神经网络的优势所在<sup>[32]</sup>。姚亚夫通过对遗传神经网络的研究，进一步延伸到利用其来对交通流量进行预测，张九跃在利用 RBF 神经网络对预测模型进行改进后，运用到了短时交通流量的预测上面，得到的预测结果一般来说具有较高的精度<sup>[33,34]</sup>。陈雪平<sup>[35]</sup>等人探究了 BP 神经网络预测交通流的基本理论，并且选取某路段实际测量的数据，通过与试验数据的对比，保证了 BP 神经网络运用在短时预测模型时，能够得到可靠的输出结果。

作为一种非传统的比较新颖的数学建模方法，神经网络有许多优点，例如它能够识别复杂的非线性系统，交通流系统由于其具有多变的繁杂因素和条件，是个非常复杂的大型系统，所以神经网络被学者们运用在交通领域。在交通信息预测研究方面，神经网络的应用相对来说起步比较晚，现在研究的目标主要是短时交通流量预测以及路程行程时间预测等方面<sup>[36]</sup>。神经网络能根据实时传递得到的交通信息，

在进行一定的复杂计算之后，来对自身网络进行更新改善，确保预测的实时性和准确性。我们从系统建模的方向来看，神经网络通常情况下选取的学习模式，是典型的黑箱（Black-box）型学习模式，因为其通过学习后所获得的输入与输出之间的关系，很难用容易让人理解的方式表达，同时学习阶段的训练程序十分复杂，训练时的收敛速度很慢，消耗时间长，不能确保模型的全局收敛性。这样通过训练获得的模型适用范围很小，一般只用于预测某一段时间的交通流量，由此可见，该模型的推广能力比较差<sup>[37-39]</sup>。

## 1.5 论文研究的内容及目标

本文以城市交通为背景，运用支持向量机的 K-均值聚类算法，致力短时交通流预测方法的研究，通过 Matlab 仿真产生预测结果来帮助相关部门更好的对交通进行管理和控制。

本文章节安排如下：

第一章：绪论。在本章中着重介绍了短时交通流预测的背景，即现代城市交通网络建设的重要性，同时介绍了国内外学者对于短时交通流预测领域的研究状况，简单描述了几种算法的优点和应用。

第二章：人工神经网络理论。在本章中介绍了神经网络的构成，同时阐明了 BP 神经网络和 RBF 神经网络的优缺点，为后面的实验选择打下了理论基础。

第三章：聚类分析与 k-均值聚类算法。本章着重介绍分析了 K-均值聚类算法的原理和优缺点，同时在最后引入支持向量机学习算法。

第四章：基于支持向量机和 K-均值聚类的 RBF 短时交通流预测仿真实验。本章运用 Matlab 仿真，使用真实数据，建立了三种短时交通流预测模型，并对这三种模型产生的结果进行处理分析。

## 第二章 人工神经网络理论

将神经网络来运用在交通信息预测的研究起步较晚，不过应用在复杂的交通信息的非线性预测中运用却有着非常显著地效果。本章主要介绍人工神经网络的理论基础，着重介绍了 BP 神经网络、RBF 神经网络这两种网络模型的结构以及它们的学习算法。

### 2.1 神经网络的基本概念

#### 2.1.1 生物神经元的结构与功能

人工神经网络是在生物学研究的基础上，依照生物神经网络的特点演化得到的，每个神经元与其相关的神经细胞一一相对的。在人类大脑皮层中，差不多存在着 100 亿个神经元，比地球上所有人类的数量还多，更有 60 万亿个神经突触和它们的连接体。作为给人类大脑供给营养的主要媒介，胶质细胞和神经细胞构成了人类的大脑，其中胶质细胞占到了将近百分之八十的比例。其余的神经细胞是庞大的神经系统的根本组成部分和功能单元，也被叫做神经元<sup>[40]</sup>。

在神经网络里面，神经元作为基本单元来进行信息的处理与传递。生物神经元主要是由四部分构成的，它们分别是突触、轴突、细胞体和树突。

1)细胞体：细胞体作为神经元的主干，它的重要组成部分有细胞核、细胞质和细胞膜。作为神经元的重要组成，细胞体被作为给神经元活动提供必须的能量的场地，保证正常的细胞活动，同时也用来容纳各种新陈代谢和生化反应。

2)树突：细胞体的形状是不规则的，在它的外表面有很多细长的突起，这些从内向外伸展的像树枝一样的部分我们称之为树突。这些树突的作用是用来充当神经元的接收器，就像触摸屏对于智能手机一样，神经元依靠树突来接纳从其他神经元传入的信息。

3)轴突：和树突不同，细胞体外有一条特别长的独立的分支，由于它的形状和纤维类似，因此被学者们称作为神经纤维。神经纤维在信号的传递过程中有着不可替代的作用，等同于信号的输出端，其尾部的许多神经末梢和其他神经元的树突接触，作为输出端来传递信号，将神经冲动传出。

4)突触：神经元相互通过轴突末梢与其他神经元的细胞体或者树突进行信号传递连接，这样的连接我们把它称为突触。

现代生理学研究已经证明：一个生物神经元或者多个生物神经功能的简单累加，通常情况下是不可能保证人类大脑正常活动的，它是由多单元的非线性的动态处理系统链接在一起才能满足人类大脑的需求<sup>[41]</sup>。

## 2.1.2 神经网络的特点

神经网络具有如下特点<sup>[42-45]</sup>:

(1) 高度的并行性。神经网络运行时需要的处理单元很多, 这些处理单元的排列方式一般为并联组合, 并且这些处理单元有很高的相似度, 这样一来, 神经网络的并行结构就非常稳定, 能够并行处理所需要解决的问题, 因此在处理问题时即使有微小的扰动和干扰也不会造成太大的误差, 同时能够快速处理好相关问题。

(2) 非线性映射能力强。神经网络所处理的问题一般是非线性的, 因此它是一种高度非线性处理系统。神经网络里面的所有神经元, 正常情况下都有能力来接收来自其它神经元输入信号, 运用并行网络进行信号处理, 处理完毕后产生输出来对于其他神经元的进行影响。就像人类大脑里的神经网络一样: 当有输入信息传递过来, 神经元就会将传入的输入信息通过自身进行分析处理, 经过一系列的运算, 如果算出的结果超出了此神经元的阈值, 它所在的神经网络才会输出一个信号, 也就是说信号的输入和输出不是线性的, 神经元具有某种非线性的计算能力。所以, 神经网络不像以往的数字电子计算机一样只能处理传统的线性信号, 它也能够处理非线性信号, 这一特点表明了人类对于智能信息的分析判断, 以及利用计算机模拟人脑智能的一次突破。

(3) 自学能力和自适应能力强。通过研究系统过去的数据库结构和记录, 神经网络不断地训练, 当其接受足够的训练后, 神经网络的性能和能力就可以总结所有的数据。所以, 以前计算方式和描述方法比较难以解决的复杂性问题, 神经网络都能够处理。神经网络经过学习所存储的以前处理过的案例, 能够自发的识别相似问题, 最终快速解决相关案例。并且神经网络能够根据周边环境的变化来自主学习, 凭借自己极强的适应能力改进自身性能。

(4) 具有良好的容错性。人类大脑的神经网络如果一旦损失某个神经元, 也不会是去原有体系中的记忆, 神经网络亦是如此, 即容错率较高。即使软件操作或者是网络硬件中的某个非关键神经元无法工作, 总体的神经网络依然能够按照以前的进程平稳的运行。

## 2.1.3 常用的神经网络模型

通常意义的神经网络, 是由极大规模数量的神经元联接, 随后而构成的系统网络。在这个复杂多变的网络中, 每个神经元之间彼此相互连接, 形式多种多样。在研究学习的过程里, 我们从各种角度对于神经网络进行了各种形式的模拟和构造, 得到了许多令人满意的结果。目前被大量采用的模型, 主要是有以下四种<sup>[45]</sup>:

(1) 前馈型神经网络。前馈型神经网络在神经网络中的出现率很高, 是一种被学者们常用的网络结构, 在这种结构里面, 许许多多的神经元经过突触来进行相互

之间的连接，神经元之间互相传递着两种信息，一种是兴奋，可以传递到下一个神经元使之产生兴奋，另一种是抑制，作用与兴奋相反，神经网络中所有的神经元就组成了在拓扑构造上非常复杂的网络结构，此网络构造能够执行许多行为，例如学习与记忆、思维。信号以及信息从输入层传送到输出层时，没有任何反馈调节的产生和操作，仅仅是单向流通，神经元第一层连接到第二层，第二层连接到第三层，依次类推。前馈型网络是当下学者们研究采用最频繁的具有监督特性的神经网络。BP 神经网络是典型的前馈网络，感知器神经网络（MLP）、径向基函数（RBF）神经网络以及小脑模型联接控制（CMAC）网络等都属于前馈型神经网络。

（2）递归型神经网络。递归型神经网络也被叫做反馈网络，反馈网络有下面这些特点：第一，神经元相互之间是局部连接或者全连接方式构成的网络结构；第二，与前馈型神经网络不一样，反馈网络有反馈信息的流动，信号不仅仅是单向流通；第三，反馈网络能够通过自我监督的学习然后对自身网络进行训练。在它的网络结构里，互连神经网络的组成是依靠一定数量的神经元互相连接形成的，这种结构可以让一部分神经元的输出，可以被本层或者前面那层的神经元用作反馈进行接收后随后来自我调节。所以，信号可以在神经元之间相互流通。Hopfield 网络、Elmman 网络和 Jordan 网络都是比较典型的递归型网络。

（3）随机型神经网络。随机型神经网络和递归型神经网络类似，也具有反馈网络的三个特点。随机型神经网络的网络结构以及输入输出等都与 Hopfield 网络大体一致，但是随机型神经网络的运行规律是随机式的。

（4）自组织竞争型神经网络。自组织竞争型网络的学习方法在神经网络中比较特别，它的学习是没有监督的完全在自己内部进行网络训练。自组织竞争型神经网络由两层网络构成，第一层为输入层，用来接收输入数据，最后一层是竞争层。因为不存在隐含层，两层之间所有神经元能够进行纵向信息的双向传递，竞争层里面的每个神经元，也可能存在信息的相互的横向传递。

到目前为止，大约有 30 多种神经网络被人们研究并且加以利用，在这些网络中，比较常用的方便的神经网络模型是：感知器（Perception）、误差反向传播（BP）神经网络、径向基函数（RBF）神经网络和 Hopfield（HNN）神经网络等。

## 2.2 BP 神经网络

BP 神经网络是目前应用最广泛的一种神经网络模型，它由多层神经元构成，是一种前馈型神经网络。BP 神经网络中的神经元之间相互传达的信号不是线性的，通常情况下是 S 型函数，虽然输入输出不是线性的，S 型函数的输出是连续不断的，数值是在 0 到 1 之间连续变化的，因此能够将输入量非线性的映射到输出量中。权值的修正这里采取的是反向传播学习法，误差的调整是从后面向前方传递的，所以

也将这种神经网络称为 BP 神经网络<sup>[46]</sup> (Back Propagation Network)。

### 2.2.1 BP 神经元结构模型

图 2.1 是一个简单的 BP 神经元模型，从图中我们可以看出，它有  $R$  个输入，每个输入与一个权值  $w$  和相结合，然后通过一定的运算得到输出值，传递然后输出，网络输出表达式为：

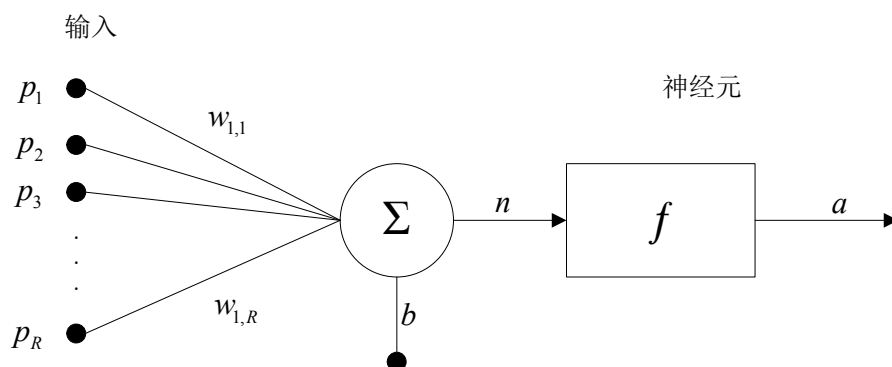


图 2.1 BP 神经元模型

$$a = f(wp + b)$$

$f$  就是表示输入和输出数值关系的传递函数。

### 2.2.2 BP 神经网络拓扑结构

BP 神经网络是一种多层前馈神经网络。与感知器仅有单层感知神经元相比，在网络结构的组成上，BP 神经网络更加复杂严谨，它的隐含层通常存在一层或者多层的隐含单元。使用 BP 神经网络来进行预测时，一般来说，可以按照实际情况的复杂程度来对神经网络的层数和节点数进行设计，按照我们事先对于数据的分析所得到的特征，然后针对此特征来构建神经网络的输入和输出方式，神经网络的输入输出方式决定了其性能的优劣，因此设计的时候一定要多次验证。其中包括隐含层的层数，以及每个隐含层所需要的神经元数目，最重要的就是对神经网络连接权值的计算，这关系到后期神经元学习记忆的速率以及准确性。BP 神经网络的误差信号是反向传播的，输入信号和其他网络一样，都是向前传递的，这种学习方式，我们称之谓有教师学习方式，是一种比较高效的学习方式。BP 算法的根本内涵在于：在信号传递的过程中，信号的数值不是不变的，它经过输入层、隐含层和输出层，每一层都会对信号进行相应的分析处理，最后验证预测输出和给定输出两者的数值误差。如果该误差的数值在经过对比之后，并没有达到事先设定的精度，就再次反向传播，信号从输入层传递到隐含层然后再传递到输出层，每层节点只会影响下一层节点所输出的数值大小，这样我们就依照测试得来的误差，根据此误差的数值，先调整输入层跟隐含层之间的权值，然后再对隐含层跟输出层之间的阈值进行调整，

经过这样一次次的调整和计算，最终能够让预测输出不断接近期望输出。下图是三层 BP 神经网络的拓扑结构：

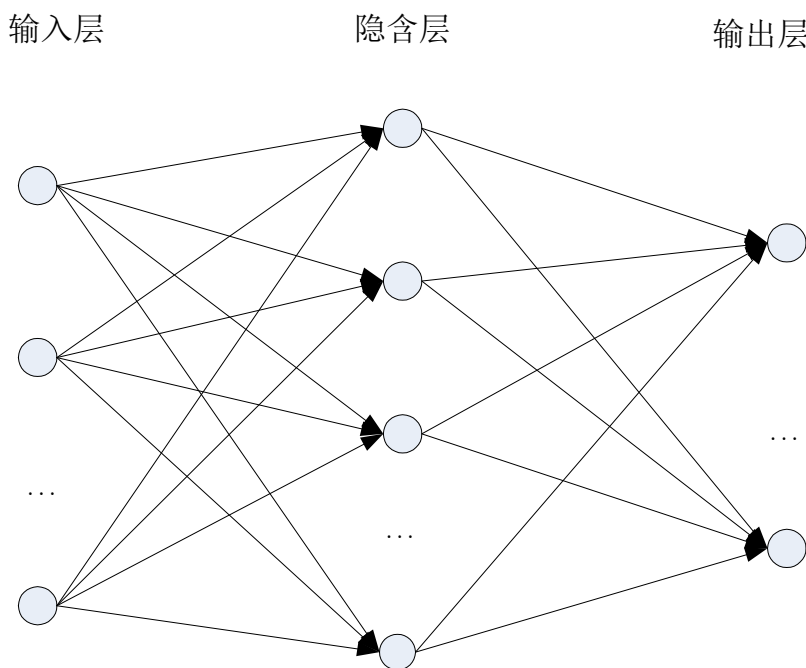


图 2.2 BP 神经网络拓扑结构

### 2.2.3 BP 算法的基本描述

三层 BP 神经网络学习算法的具体实现如下：

假设 BP 网络有  $n$  个输入层节点， $l$  个隐含层节点， $m$  个输出层节点，输入层与隐含层之间的权值为  $w_{ji} (i=1,2,\dots,n; j=1,2,\dots,l)$ ，隐含层与输出层之间的权值为  $u_{kj} (k=1,2,\dots,m; j=1,2,\dots,l)$ ，隐含层神经元和输出层神经元都是采用  $S$  型激活函数，*Sigmoidal* 函数，其表达式为：

$$f(x) = \frac{1}{1 + \exp(-\lambda x)} \quad 2-(1)$$

其中，假设训练样本的输入为  $X$ ，期望输出为  $Y$ ， $X = \{x_1, x_2, \dots, x_n\}$ ， $Y = \{y_1, y_2, \dots, y_n\}$ ，那么，隐含层各个神经元的输入为：

$$H_j = \sum_{i=1}^n w_{ji} x_i, j=1,2,\dots,l \quad 2-(2)$$

为了计算方便，我们取 *Sigmoidal* 函数中的参数  $\lambda$  为 1，那么隐含层各个神经元得的输出为：

$$net_j = f(H_j), j=1,2,\dots,l \quad 2-(3)$$

输出层各个神经元的输入为:

$$O_k = \sum_{j=1}^l u_{kj} net_j, k=1, 2, \dots, m \quad 2-(4)$$

输出层各个神经元的输出为:

$$y_k = f(O_k), k=1, 2, \dots, m \quad 2-(5)$$

假设输出层的第  $i$  个神经元的期望输出为  $d_i$ ，那么这个神经元的误差为:

$$e_i = d_i - y_i \quad 2-(6)$$

则输出层的误差函数为:

$$E = \frac{1}{2} \sum_i e_i^2 = \frac{1}{2} \sum_{i=1}^m (d_i - y_i)^2 \quad 2-(7)$$

输入层和隐含层两者之间的连接权值比较重要，它的误差信号方程为:

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}, i=1, 2, \dots, l, \quad j=1, 2, \dots, n \quad 2-(8)$$

隐含层与输出层之间的连接权值的误差信号方程为:

$$\Delta u_{ij} = -\eta \frac{\partial E}{\partial u_{ij}}, i=1, 2, \dots, l, \quad j=1, 2, \dots, m \quad 2-(9)$$

在上式中，有:

$$\frac{\partial E}{\partial u_{ij}} = \frac{\partial E}{\partial y_i} \frac{\partial y_i}{\partial u_{ij}} \quad 2-(10)$$

其中:

$$\frac{\partial E}{\partial y_i} = -(d_i - y_i) \quad 2-(11)$$

$$\frac{\partial y_i}{\partial u_{ij}} = \frac{\partial y_i}{\partial O_i} \frac{\partial O_i}{\partial u_{ij}} \quad 2-(12)$$

由于  $S$  型函数有:

$$f'(x) = f(x)(1 - f(x)) \quad 2-(13)$$

则 2-(12) 式中:

$$\frac{\partial y_i}{\partial u_{ij}} = y_i(1 - y_i) net_j \quad 2-(14)$$

因此，式 2-(9) 为:

$$\Delta u_{ij} = -\eta (d_i - y_i) y_i (1 - y_i) net_i \quad 2-(15)$$

而

$$\frac{\partial y_k}{\partial net_i} = \frac{\partial y_k}{\partial O_k} \frac{\partial O_k}{\partial net_i} = y_k(1 - y_k) u_{ki} \quad 2-(16)$$



$$\frac{\partial net_i}{\partial w_{ij}} = \frac{\partial net_i}{\partial H_i} \frac{\partial H_i}{\partial w_{ij}} = net_i(1 - net_i)u_j \quad 2-(17)$$

因此，式 2-(8)为：

$$\Delta w_{ij} = \eta \sum_{k=1}^m (d_k - y_k) y_k u_k net_i (1 - net_i) x_j \quad 2-(18)$$

由此可得到网络在训练过程中的权值迭代公式，这样一来，输出层与隐含层之间的权值迭代公式：

$$u_{ij}(t+1) = u_{ij}(t) + \eta(d_i - y_i)y_i(1 - y_i)net_j, \quad i = 1, 2, \dots, m, j = 1, 2, \dots, l \quad 2-(19)$$

那么，隐含层和输入层之间的权值迭代公式为：

$$w_{ij}(t+1) = w_{ij}(t) + \eta \sum_{k=1}^m (d_k - y_k) y_k (1 - y_k) net_i (1 - net_i) x_j, \\ i = 1, 2, \dots, l, j = 1, 2, \dots, n \quad 2-(20)$$

通过上述算法，训练样本即可以按照此方法：利用误差反向传递的特性，对每层神经元之间权值的系数数值大小进行不断的调整，然后多次重复输入全部的训练样本数据，就这样不断训练，直到所得结果的数值误差大小在设定的可接受范围内；抑或是达到提前计划好的网络最大训练程度。以上两种情况下任意一种发生后，网络训练就算终止了。

#### 2.2.4 BP 神经网络的优缺点

BP 神经网络在数学理论基础方面具有十分严谨的特性，同时它还有自主学习能力，并且具有优良的容错能力和良好的泛化能力。凭借上面所列举出的优点，目前为止，它仍然是前馈型网络所最需要的算法之一，BP 神经网络在自动控制、预测、模式识别等领域，都得到了学者们充分的应用。

与此同时，BP 算法也存在不少缺点，由于它是一种静态梯度下降法，性能的优劣很大程度上由初始条件决定，因此该算法仍具有以下局限性：

(1) 收敛速度慢、稳定性不足。因为网络结构的整体复杂程度不同，相对应的权值及阈值在对同一个样本的收敛效果也不尽相同，所以，可能导致网络的整个学习效率降低，学习速度不乐观。当我们在算法中使用不同大小的学习率，学习率小自然收敛速度缓慢，而学习率一旦提高，网络权值的改变量也会随之提高，虽然可以加快速率，更快的接近误差函数最小值。但是，如果神经网络算法中的学习率过高，某些情况下就有可能诱导网络权值的波动，进而产生由波动引起的震荡现象。

(2) 容易陷入局部极小。BP 算法对于起初给与的数值，也就是初始权值非常依靠，初始权值不同，那么其初始化的 BP 网络算法的收敛速度也会不同。假如我们选取的算法不合适，比如选用了误差平方和函数，误差曲面就会有很多极小值的

点，这样就导致了 BP 算法在梯度下降的过程中收敛于不同的局部极小值。所以，BP 神经网络算法的结果十分容易陷入局部极小无法跳出，这样就使得最终得到的输出值无法高度逼近期望的输出值。

(3) 泛化性能不足。BP 算法的泛化性能不足，主要原因在于：BP 神经网络在实现训练样本的输入输出映射上表现得很好，但是每当输入量变化时，即输入一组新的样本时，不一定能够得到理想的输出值。泛化性能的不足可能是因为训练数据的松散性决定的，例如所输入的训练数据不能够涵盖所有类型的特征，当然，也有原因是数据本身的非线性程度过高造成的。神经网络的结构也很重要，深刻影响着它自身的泛化能力。例如 BP 神经网络的隐含层层数，也是决定神经网络性能的关键，对于 BP 神经网络的各项性能也会产生较大的影响，例如泛化性能。一般来说，当隐含层的层数为一层或者两层时，BP 神经网络具有很好的收敛特性，当隐含层层数为零，或者隐含层层数多于两层时，其输入量在训练时的波动就会慢慢变大。隐含层所含有的神经元数量越复杂，计算量越大，收敛速度就愈加缓慢，但是仍旧可以获得令人满意的训练和调测误差，这样虽然数据收敛的速率降低，但是能有效地加强 BP 神经网络的预测和训练能力。当然，不是说 BP 神经网络中神经元数目越多越好，隐含层中的神经元数量一旦过多，BP 神经网络进行训练后的样本所得到的误差结果很令人满意，但是测试样本的波动就可能愈加强烈，这就表示着一定情况下，BP 神经网络训练能力的提高，反而会导致预测能力的下降。

## 2.3 RBF 神经网络

早在上个世纪，1985 年 Powell 研究并发表了关于多个变量的插值径向基函数 (Radical Basis Function, RBF) 算法。几年后，在 1988 年由 Moody 和 Darken 两位国外学者研究后，总结出了具有许多优良特性的神经网络构造，这种网络结构属于前向型神经网络，例如结构自适应确定，初始权值对于输出的影响没有那么明显等，这就是 RBF 神经网络<sup>[47,48]</sup>。与 BP 神经网络相对比，在面对同一个现实问题需要解决处理的时候，RBF 神经网络所需要的神经元数目，正常情况下会比 BP 神经网络庞大许多，网络结构也会更加复杂，但 RBF 神经网络计算所耗费的训练时间，一般来说和 BP 神经网络计算所耗费的训练时间相比，要少很多。

### 2.3.1 RBF 神经网络结构

RBF 神经网络的神经元结构组成，如下图所示：

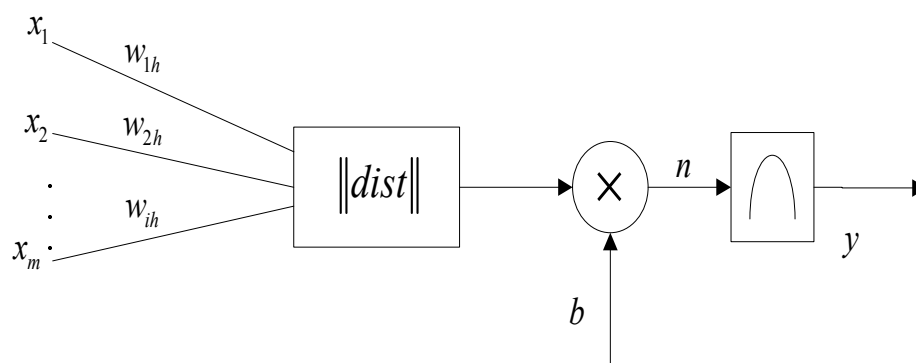


图 2.3 RBF 神经元模型

所有神经网络的性能优劣，基本都依靠于结构和优良算法的选择。径向基函数作为 RBF 神经网络的激活函数的算法选择，通常情况下，我们将其假定为空间中某点到某个中心点之间的欧式距离的单调函数。由上图我们能够得出，对于 RBF 神经网络的激活函数，不是固定不变的，而是将输入和权值向量两者之间  $\|dist\|$  的数值当成变量的。通常来说，作为 RBF 神经网络的激活函数，它的表达式如下所示：

$$R(\|dist\|) = e^{-\|dist\|^2} \quad 2-(21)$$

最基本的 RBF 神经网络包括三层，和 BP 神经网络大致相同的是，它也是一种由三层静态前向网络构成的。一般的 RBF 神经网络模型结构如下图所示：

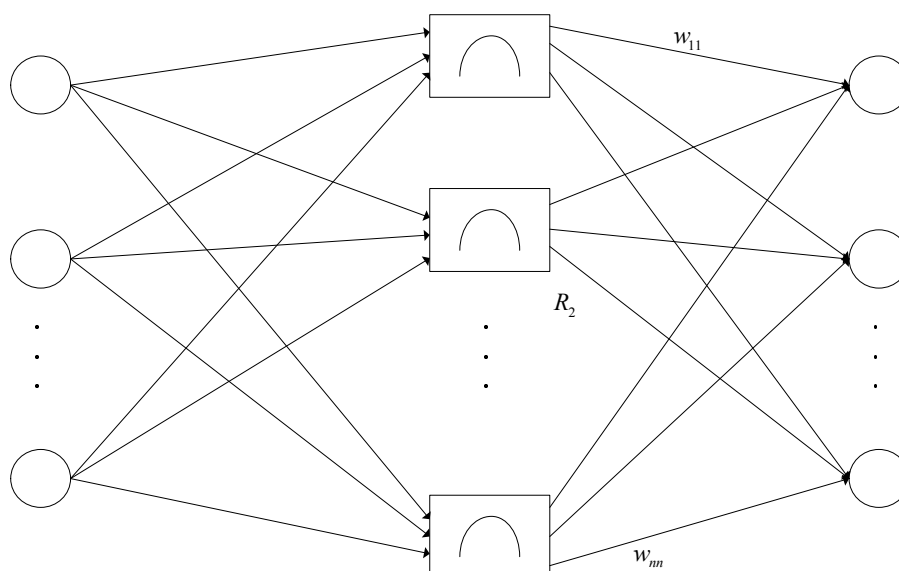


图 2.4 RBF 神经网络结构

其中，第一层被称作是输入层，作用是把外部环境和神经网络连接起来，同时接受外部输入数据，由信号源节点组成；第二层被称作是隐含层，主要用途是将输入层得到的数据，在层内进行非线性变换，隐含层的维数要按照实际情况的需要来确定，一般来说维数都比较高；第三层被称作是输出层，数据经过输出层后是线性

的，它的作用是为激活信号提供响应。上图描绘出了拥有前面为  $n$  个输入层节点，中间为  $k$  个隐含层节点，最后为  $m$  个输出节点，如图构成三层 RBF 的神经网络结构。

其中， $x=[x_1, x_2, \dots, x_n]^T$  为输入矢量； $W_{km}=[w_1, w_2, \dots, w_k]^T$  为网络输出权值矩阵，其中  $w_i=[w_{i1}, w_{i2}, \dots, w_{im}]^T$ ；

### 2.3.2 RBF 神经网络的学习算法

与前一种算法不同的是，RBF 神经网络学习算法最主要的参数有三个，第一个是基函数的中心，第二个是方差，最后一个就是隐含层到输出层的权值。RBF 神经网络从问世以来，科学家们对它所研究出的算法就有很多种，例如根据选择中心方式的区别，我们可以有多种算法，有随机选取中心法、自组织选取中心法和正交最小二乘法等等算法。接下来我们针对 RBF 神经网络对于基函数中心的确定方式差异，简单讲解几种使用率较高的 RBF 神经网络学习算法<sup>[49]</sup>。

(1) 自组织选取中心。此算法主要由两个阶段组成：第一个阶段是神经网络开展自我学习自我训练，这个时候不存在导师学习，只能依赖自身，隐含层基函数的两个关键点就是中心与方差的处理是否得当；第二个阶段是有导师学习的，也被称为有监督学习，介于隐含层和输出层之间的权值作为此时求解的关键。

RBF 神经网络很复杂，它所使用的径向基函数也是有很多种的，在这里我们只介绍运用了高斯函数的径向基函数，在此基础上，RBF 神经网络的激活函数可以用下式表达：

$$R(x_p - c_i) = \exp\left(-\frac{1}{2\sigma^2} \|x_p - c\|^2\right) \quad 2-(22)$$

式中， $\|x_p - c\|^2$  为欧式范数。

$c$  为高斯函数的中心。

$\sigma$  为高斯函数的方差。

由 2-(22) 式中 RBF 神经网络的结构，我们能够推得网络的输出为

$$y_j = \sum_{i=1}^h w_{ij} \exp\left(-\frac{1}{2\sigma^2} \|x_p - c\|^2\right) \quad j=1, 2, \dots, n \quad 2-(23)$$

式中， $x_p=(x_1^p, x_2^p, \dots, x_m^p)^T$  为第  $p$  个输入样本。

$p=1, 2, \dots, P$ ， $P$  表示样本总数。

$c_i$  为网络隐含层节点的中心。

$w_{ij}$  为隐含层到输出层之间的连接权值。

$i=1, 2, \dots, h$  为隐含层的节点数。

$y_j$  为实际输出，它所对应的节点就是输入样本系统里面的第  $j$  个输出节点。

在这里， $d$  是样本的期望输出值，因此我们可以用下式作为基函数的方差

$$\sigma = \frac{1}{P} \sum_j^m \|d_j - y_j c_i\|^2 \quad 2-(24)$$

(2) 随机选取固定中心。假定隐含层结点的激活函数是固定的，那么这就是最简单的方法，我们随机地从所需要学习的数据中挑选需要的中心位置，如果我们输入的学习样本是依照目前所需处理问题的典型方式分散布局的，一般来说，这种方法就比较合适。在这里，我们设定一个将  $c_i$  作为中心的径向基函数定义为：

$$\phi(x, c) = \exp\left(-\frac{1}{d_{\max}^2} \|x - c_i\|^2\right), \quad i=1, 2, \dots, k \quad 2-(25)$$

其中， $k$  是隐含层节点数， $d_{\max}$  是选取中心距离的最大值。通过上式我们能够得出，隐含层的节点的宽度都已经被确定了，如下式所示，这样我们就能确保任何一个径向基的数值都不会太突出或者太保守。

$$\sigma = \frac{d_{\max}}{\sqrt{2k}} \quad 2-(26)$$

在上面这个方法里，需要确定的学习参数，是输出层的线性权值  $W$ ，利用伪逆法即可得出，如下式所示。

$$W = G^+ Y \quad 2-(27)$$

其中， $Y$  是学习集合里面的期望输出向量。矩阵  $G^+$  是矩阵  $G$  的伪逆， $G$  的定义如下式所示。

$$G = \{g_{qi}\} \quad 2-(28)$$

其中

$$g_{qi} = \phi(x_q, c_i) = \exp\left(-\frac{k}{d_{\max}^2} \|x_q - c_i\|^2\right), \quad q=1, 2, \dots, N, i=1, 2, \dots, k \quad 2-(29)$$

上式中， $x_q$  是学习样本中的第  $q$  个输入向量， $c_i$  为第  $i$  个中心。

(3) 中心的监督选择。在这种方法中，凡是属于神经网络里面的自由参数，都不是无导师的，也就是说它们都会接受导师学习，也被称为监督学习。其优点就是因为选取的是误差修正学习过程，算法可以不断地进行自我调整改善，所以可以非常顺利的运用梯度下降法。

## 2.4 本章小结

本章对 BP 神经网络以及 RBF 神经网络的原理结构以及算法研究进行了介绍，并且对它们的优缺点进行了分析，为下一步选择适当的算法进行预测提供了理论基础。



## 第三章 聚类分析与 K-均值聚类算法

### 3.1 聚类分析与算法

在信息大爆炸的时代，由于每天都要面对纷繁杂乱的信息，人们对于事物分类的需求越来越高，这样才能更高效的处理日常生活中遇到的问题。以前人们都是依靠生活经验和累计的知识来进行分类，但是这样的分类很难做到十分精确。因此，数学工具和多元分析用在了分类学中，这两种工具大大提高了分类的准确性，慢慢的形成了聚类分析。聚类分析的应用在生活中无处不在，例如在购买手机的用户中找寻那些愿意花高价购买附属产品的用户；在买房的客户中标识那些是为了子女上学便利的客户；比对着雾霾的发生率将实施监测的城市进行是否宜居等分类。聚类分析对数据的处理非常有效可靠，将得到的数据进行深层次的挖掘，其中的聚类算法就是学者们探究的重要方向，是一种无监督数据分类的方法。聚类算法有许多种，例如系统聚类法、动态聚类法和基于划分聚类法等<sup>[50]</sup>。

#### 3.1.1 聚类的定义

聚类算法的基本定义是由 Evertt 提出的，简单来说，某些相似的实体集合在一起就是聚类，它们可能外形相同，可能大小相同，也可能重量相同，但是它们彼此之间肯定有共通点。不同类型的聚类之间，他们的实体也可能是不一样的。聚类的数学模型如下：

设论域  $X$  为没有分类的  $n$  个样本  $x_1, x_2, \dots, x_n$  的所有，各个样本  $x_i$  采用它的特征向量  $P(x_i) = (x_{i1}, x_{i2}, \dots, x_{is})$  来进行概述，此时，聚类分析的作用，通常来说就是用来分析论域  $X$  中的全部样本，把它们特征向量其中的相似的地方照出来，按照各个样本的相同程度的大小，将论域  $X$  分成了许多相互没有交集的子集  $X_1, X_2, \dots, X_k$ ，那么

$$X_1 \cup X_2 \cup \dots \cup X_k = X \quad 3-(1)$$

$$X_i \cup X_j = \phi \quad 3-(2)$$

将任意样本  $x_i (i=1, 2, \dots, n)$  设定为其中一类，这一类的样本中有它们共同相似的地方，并且这个类中任意样本不会隶属于任何其他类，此类与其他类的共通点不一致，这就意味着任意样本不会同时属于两类。由此我们可以得出，样本  $x_i$  对子集  $X_j$  的类属关系，我们可以采用下方的隶属度函数进行表示：

$$\mu_{x_j}(x_i) = \mu_{ij} = \begin{cases} 1 & x_i \in X_j \\ 0 & x_i \notin X_j \end{cases} \quad 3-(3)$$

#### 3.1.2 聚类分析中不同的数据类型

对于我们经常要在聚类算法中处理的数据对象，因此，许多聚类算法由于是基于内存方式构成的，他们有以下两种数据结构：

数据矩阵 (data matrix) 是将数据对象和它的数据属性一一对应结构，它是由  $n$  个对象构成的，假设这个聚类的问题中有  $n$  个对象：  $x_i = (i=1, 2, \dots, n)$ ，任意一个对象都有  $p$  个属性，其中第  $i$  个对象的第  $j$  个属性的观测数值为  $x_{ij}$ 。数据矩阵的关系表如下图所示，是一个  $n * p$  的矩阵：

$$\begin{pmatrix} x_{11}, x_{12} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1}, x_{n2} & \cdots & x_{np} \end{pmatrix} \quad 3-(4)$$

这个图一般被叫做数据矩阵，在这个矩阵里面第  $i$  个对象  $p$  个变量的观测数值记做：

$$x_i = (x_{i1}, x_{i2}, \dots, x_{ip}) \quad 3-(5)$$

下面我们将讨论如何计算和使用以下五种属性，这五种属性单独操作或者是将他们挑选搭配后用来计算得出某些数据对象之间的不同所在。对对象来进行聚类分析依赖的就是这些数据对象的差异值。

#### 1. 间隔数值属性 (interval-scaled attributes)

连续测量值如果他们的比例呈直线型，那么我们称之为间隔数值属性，这样我们用来表示他们的数量关系。比较常见的间隔数值有体重、湿度以及压力等。

在对同一件物体或者同一种情况进行测量时，如果选取不一样的测量单位，那么就可能对聚类分析的结果产生不同的效果。比如我们在测量一个游泳池的水深时，把测量单位（对于长度属性）从厘米变成英寸，就会产生不一样的聚类效果。一般情况下如果我们选取一个偏小的单位用来说明某个物体的某项特性的话，那么该特性的量取范围就会变得很大，如此一般会对聚类结构产生一定的波动。但是当我们提前对于数据进行稍微调整，也就是标准化后，那么就可以减少算法对于属性单位过度的依赖性。在这里我们将标准化测量定义为把同样的权值赋给所有的属性。

下面我们介绍一种方式来实现标准化测量，那就是把初始测量值替换成为没有单位的变量。通过下面的公式我们将一个给定的属性  $f$  来进行标准化，其中  $f$  是变量，公式如下<sup>[51]</sup>：

$$s_f = \frac{1}{n} (|x_{1f} - m_f| + |x_{2f} - m_f| + \dots + |x_{nf} - m_f|) \quad 3-(6)$$

上式中，  $x_{1f}, x_{2f}, \dots, x_{nf}$  是变量  $f$  的  $n$  个测量值。  $m_f$  是变量  $f$  的平均值，那么

$$m_f = (x_{1f} + x_{2f} + \dots + x_{nf}) / n \quad 3-(7)$$

#### 2. 二值属性



二值属性比较特别，它的变量取值只有两种情况，那就是 0 或者 1。其中，如果取值为 0，那么我们就称这个变量所表示的状态不存在，因此，如果取值为 1，那么相应的状态就表示为存在。对于一个假设用来记录成年人喝酒与否的二值变量 **drinker**，那么如果 **drinker** 取 1 时，这就意味着这位成年人喝酒，相反的，当 **drinker** 取 0 时就代表着这个成年人不喝酒。假如我们根据间隔数值变量这种情况来应对二值变量的情形，通常情况下会得到不甚理想的聚类分析结果。所以我们需要运行特殊的分析方式，将用二值变量表达的数据对象之间的相互差异给运算出来。

凭借二值数据来进行差异矩阵的运算，就是获得差异程度多种方法中的一例。假设我们认定全部二值变量它们所包含的权值都是一样的，就能够获得  $2 \times 2$  的一个可能性表格，如下表所示：

I取值 J取值	1	0	合计
1	q	r	q+r
0	s	t	s+t
合计	q+s	r+t	q+r+s+t

图 3.1 二值变量可能性表

在上面的表格里，{对象  $i$ ，对象  $j$ } 这两个对象的取值为  $\{1,1\}$  时，二值变量对数目为  $q$ ，{对象  $i$ ，对象  $j$ } 这两个对象的取值为  $\{1,0\}$  时，二值变量对数目为  $r$ ，{对象  $i$ ，对象  $j$ } 这两个对象的取值  $\{0,1\}$  时，二值变量对数目为  $s$ ，{对象  $i$ ，对象  $j$ } 这两个对象的取值为  $\{0,0\}$  时，二值变量对数目为  $t$ 。从表中我们能够看出，二值变量一共包含的所有的对数数目为  $p$ ，则  $p = q + r + s + t$ 。

假设我们称该二值变量是对称的，那么它的取值为 0 或者为 1 时，二值变量它所代表的内容，跟数值的大小或者其它特性没有关系，其作用是一样的。比如，上文中的 **drinker** 即为对称变量，用 0 或者 1 代表成年人是不是喝酒或者是不喝酒并不是必须的，1 也可以代表成年人不喝酒，因为这里只是代表喝酒与否的一种状态，与数值无关，并不意味着数值大的更有意义。

相反地，如果该二值变量并不是对称的话，那么它的取值为 0 或者为 1 时，二值变量所表示内容的重要意义是不同的。一般情况下，我们把概率较小的事件用 1 来表示，其他事件让 0 来代替。

### 3. 符号、顺序和比例数值属性

#### (1). 符号变量

二值变量仅仅只能对两个状态来进行描述，一旦状态数量多于两种，那么二值

变量就无法来进行描述。符号变量就是二值变量的一个延伸，能够描述表达多于两个的状态。比如红绿灯变量就是符号变量，它能够表示三种状态，那就是红色、绿色还有黄色。

假设符号变量所拥有的状态数量为  $M$ ，我们可以将其中的任何一个状态用字母、数字甚至符号来表示，这里为了方便数据处理，我们采用整数集合，没有数值上的大小关系，如： $1, 2, \dots, M$ 。

符号变量最简单的匹配方式，就是计算对象  $i$  和对象  $j$  之间的不同。具体描述如下式所示：

$$d(i, j) = (P - m) / P \quad 3-(8)$$

式中， $m$  代表对象  $i$  与对象  $j$  之间能够获得同样状态的变量数目， $p$  是所有符号变量的个数。

## (2).顺序变量

顺序变量是用来表达在不能高效的用相对客观的方式来展现主观质量价值这种情况的，比如：年纪高低作为一个顺序变量，它是从大学一年级、大学二年级、大学三年级、大学四年级这样的有序方式排列组成的。这样的连续顺序看起来很像一组不确定区域的连续的数值，但是绝对数值的意义仍然不大，有意义的只是他们的与彼此之间的位置差异。比如，在篮球比赛中，一只球队在所有球队中的排名位次通常情况下比他得了多少分更加重要。

## (3).比例数值变量

比例数值变量是在非线性的尺度上得到的正测量值，比如指数比例，我们能够用下面的公式进行表达：

$$Ae^{Bt} \text{ 或 } Ae^{-Bt}$$

式中， $A$  和  $B$  是正的常数。

## 4.混合类型属性

上面我们列出了几种变量类型，这些变量类型有：间隔数据类型、对称二值类型、符号类型、顺序类型以及比例数值类型。这只是理想状态，在通常情况下的数据库里面，数据对象通常是由以上几种变量类型一起混杂起来表示的。

处理方式包含两种，第一种就是把每一个类型的变量单独进行集合，然后逐个击破，每一个类型的变量都运行自己所对应的聚类进行分析。但是通常情况下，这种方法所得的结果不是那么令人赞同，因此我们一般选择第二种方法，那就是我们将所有类型的变量聚集在一起组织处理，在同一时间运行整体的聚类分析。这种情况下，我们要求把不同类型变量聚集在一起，然后放到同一个差异矩阵中去，最终把全部对我们有用的取值都映射到既定的区间  $[0, 1]$  中去。

## 5. 描述聚类的特征量

我们假设一个聚类  $G$ ，这个聚类中有  $N$  个数据对象，记做  $G(i=1,2,\dots,N)$ ，如果将其看成是随机变量，那么我们就能够用下面这些特征量对聚类进行表达：

$$\text{质心: } m = \frac{1}{N} \sum_{i=1}^N x_i \quad 3-(9)$$

离差矩阵与协方差矩阵：

$$A_G = \sum_{i=1}^N (x_i - m)(x_i - m)^T \quad 3-(10)$$

$$S_G = \frac{A_G}{N-1} \quad 3-(11)$$

直径的定义方法有许多种，相对而言较简单的如下式：

$$D_G = \max_{1 \leq i, j \leq N} (d(x_i, x_j)) \quad 3-(12)$$

或者

$$D_G = \sum_{i=1}^N (x_i - m)(x_i - m)^T \quad 3-(13)$$

### 3.1.3 相似性度量

在聚类算法中，一般情况下，某组数据的相似性程度大小，我们用此组数据相互之间的距离来阐释。通常来说，数据间的距离越大，数据愈稀疏，任意两者相互之间的相似性就越小。从另一个角度来看，数据间距离越小，数据愈紧密，则认为它们的相似性就越大，将它们归结到一类的可能性就无限大。不管在任何情况之下，数据之间的距离，如下面几组公式，都是零或者是正数，即不能为负数，而且可以构成三角不等式，符合数学逻辑。下面我们假设样本里有  $n$  个变量， $d(i, j)$  作为第  $i$  个样本和第  $j$  个样本间的距离，经常使用的距离公式如下：

(1) 明氏距离

$$d_{ij}(q) = \left( \sum_{k=1}^p |x_{ik} - x_{jk}|^q \right)^{\frac{1}{q}}, q > 0 \quad 3-(14)$$

此时  $q$  为  $1, 2, \dots, \infty$ ，所对应的明氏距离是：

1) 绝对值距离：

$$d_{ij}(1) = \sum_{k=1}^p |x_{ik} - x_{jk}| \quad 3-(15)$$

2) 欧式距离：

$$d_{ij}(2) = \left( \sum_{k=1}^p |x_{ik} - x_{jk}|^2 \right)^{\frac{1}{2}} \quad 3-(16)$$

3) 切比雪夫距离:

$$d_{ij}(\infty) = \max |x_{ik} - x_{jk}| (1 \leq k \leq p) \quad 3-(17)$$

(2) 马氏距离

一般情况下, 样本里面的每个变量的属性都不是固定的, 而根据随机变量组成的随机向量按照某些模式进行布局, 每个分量相互之间又有可能容纳相当程度的关联, 因此, 第  $i$  个向量和第  $j$  个向量根据下式, 我们能够得出它们之间的马氏距离是:

$$d_{ij}^2(m) = (x_i - x_j)^T \sum^{-1} (x_i - x_j) \quad 3-(18)$$

上式里面,  $\sum^{-1}$  是逆矩阵, 归属于随机变量的协方差矩阵。

(3) 曼哈顿距离

曼哈顿距离也被称作是城市区块距离, 也就是在欧几里得空间的固定直角坐标系上, 两个点形成的线段对各坐标轴产生的投影的距离总和, 距离公式如下:

$$D = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}| \quad 3-(19)$$

## 3.2 K-均值聚类算法

K-均值算法在聚类分析算法中, 是一类最基本的并且使用最频繁的划分算法, 它被广泛的应用于科学和工业中, 是由 MAC Queen 在 1967 年提出的一种非监督实时聚类算法<sup>[52]</sup>。

### 3.2.1 K-均值聚类算法流程和运行过程

K-均值算法的运行过程: 第一, 我们先安排聚类的数量  $K$ , 然后是设置好  $K$  个最开始聚类中心的初值, 最后是所需要的迭代的轮次, 抑或是收敛条件等初始条件, 以便于最终满足条件停止算法; 第二, 我们再依照某种相似性的度量准则, 将其量化, 更直白的利于我们理解分析, 比如距离, 把所有数据安排到距离各自最近的聚类中心, 这样相近的数据就形成了类, 它们就拥有了共性; 第三, 我们将每个类的平均矢量作为它们各自的聚类中心, 然后执行算法以及相关运算, 如此重复操作, 最终在满足我们前面设定好的条件时终结运算。

我们将大量样本数据  $X = \{x_1, x_2, x_3, \dots, x_n\}$  中所有的个体分成  $K$  类, 如此这般之后, 聚类中心就是  $C = \{c_1, c_2, c_3, \dots, c_k\}$ , 同时  $d_{ij}(x_i, c_j)$  表示样本  $x_i$  和匹配的中心  $c_j$  两者之间的距离, 这大量样本数据中所有的个体和它们匹配的质心距离数值的累加, 这个数据对于我们很重要, 因此我们用函数  $J$  进行表达, 它的表达式如下式:

$$J = \sum_{j=1}^K \sum_{i, j \in c_j} d_{ij}(x_i, c_j) \quad 3-(20)$$

目标函数  $J$  的数值大小很重要, 当它的数值很小时, 那么就表明此聚类越紧凑,

当它的数值偏大时，则表明该聚类很分散，因此其直接反映出聚类效果的优劣。为了得到更适合的结果，我们可以通过变换目标函数  $J$  的数值，从而对聚类方案进行调整，根据不同的结果来寻找最合适的数值，尤其是目标函数  $J$  达到极小值时，这个时刻我们所得到的聚类，效果最优异，就被称作最优聚类方案。

下面我们换一组特性来对聚类进行归类，这次我们挑选欧氏距离作为此次聚类的特性，它可以充分描述样本  $x_i$  以及聚类中心  $c_j$  之间的关系，组  $j$  包含  $x_i^{(f)}$  的数据，组  $j$  的样本的个数是  $n_j$ ，那么我们可以很轻易的得出目标函数，用下式表达：

$$J = \sum_{j=1}^K J_j = \sum_{j=1}^K \left( \sum_{i,j \in c_j} \|x_i^{(f)} - c_j\|^2 \right) \quad 3-(21)$$

解求最佳中心  $c_j$  的过程就叫做聚类过程，这样就能够让目标函数  $J$  为最小值，

$$\frac{\partial J}{\partial c_j} = \frac{\partial}{\partial c_j} \sum_{i=1}^{n_j} (x_i^{(f)} - c_j)^2 = -2 \sum_{i=1}^{n_j} (x_i^{(f)} - c_j) = 0 \quad 3-(22)$$

这时聚类中心为：

$$c_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_i^{(f)} \quad 3-(23)$$

### 3.2.2 K-均值聚类算法的优缺点分析

不管是最开始单纯的 K-均值聚类算法，或者是融合了聚类准则函数之后的 K-均值聚类算法，除了彼此的差异之外，它们也具有相同的优点，那就是都执行两个层次的反复迭代过程，算法终止的要求是没有数据样本被再次分配：1.指定聚类，就是把数据  $x_i$  分配到某一个聚类，能够让它和此聚类中心的距离达到最近。2.重新寻找聚类中心。K-均值聚类算法在运用的时候主要包含以下这些问题：

1) K-均值聚类算法中的聚类数目  $k$ 。

通常情况下，聚类个数  $k$  的数值是比较难以确定的，在算法执行之前，我们不知道该把事先给定的数据集究竟要分成多少类才能够让算法运行的更加顺利， $k$  的取值大小很大程度上决定了算法的精确性。关于 K-均值聚类算法中聚类个数  $k$  值的确定，有些依据方差进行分析验证，采用混合  $F$  统计量对最佳分类数进行确定，同时运用了模糊划分熵方法来检验最佳分类数的正确性。

2) K-均值聚类算法对初始值的选取。

一开始，我可以先挑选  $k$  个点来当作初始聚类中心，这些点不能是特意挑选的，而是无意识的选择，我们的目标是让算法达到收敛，然后运用迭代的重新定位技术，不断找寻最优聚类中心。但是不一样的初始数值所诱导的算法聚类收敛结果不是那么稳定。K-均值聚类算法经常把误差平方以及准则函数，当做聚类准则函数，

即目标函数。在空间状态中，目标函数往往作为一个非凸函数，根据以前对于 K-均值聚类算法的了解我们可以知道，这样的函数经常存在许多局部极小值，但是这么多局部最小值都是干扰的，因为里面通常只存在一个数值是全局最小值。所以如果每次在选取算法的初始值时如果一个不小心将初始聚类中心选在了非凸函数曲面的“位置”，经过不断地迭代之后，目标函数往往得到的是局部最小而不是全局最小。因此初始值的选择对于 K-均值聚类算法的聚类结果的影响是非常深刻的，这就诱导了聚类结果的不确定性。

3) 将簇的质心（即均值点）当作聚类中心，再次执行聚类计算，但是在距离大量数据所在的密集区很远的孤立点以及噪声点会产生干扰，就可能使聚类中心落在数据稀疏的区域，远离我们希望的数据密集区，因此 K-均值聚类算法在面对噪声点以及孤立点时通常束手无策。

#### 4) K-均值聚类算法通常情况下只能发现球状簇

在 K-均值聚类算法中，通常情况下，我们将误差平方和准则函数当作聚类准则函数（目标函数），考察目标函数之后，我们发现，倘若各个簇之间的差异比较大，而且数据排布不是很分散，那么基于欧氏距离的误差平方以及准则函数运行的结果相对比而言更加令人满意；不过，假如各个簇之间的外形以及区域大小差异很大的话，聚类簇分割的状况就会出现，原因是为了使误差平方和  $J_c$  值达到我们所期望的足够小的数值。

#### 5) K-均值聚类算法消耗大

根据 K-均值聚类算法的结构，我们能够发现，它必须不停地对样本分类，然后进行迭代调整，最后需要不停地处理经过运算之后的新聚类中心。由此我们不难看出，一旦数据集所包含的数据量非常多，算法进行计算所需要的时间也是令人叹为观止的。因此我们要改进算法的时间复杂度，来扩展算法在其他领域的应用。

### 3.3 基于支持向量机的 K-均值聚类算法

由于 K-均值聚类算法对于初始聚类中心的选取十分依赖，一般情况下我们都会挑选许多不一样的初始值，运行算法，最后从所有结果中选一个最优初始值。这样就会有一个明显的不足，那就是假如初始聚类中心的挑选数量比较多的话，就会增加大量的计算，使得效率大大降低；相反地，假如初始聚类中心的挑选数量偏少的话，那么即时计算结束，我们也不一定能够获得最优解。由此产生了许多改进的 K-均值聚类算法：P.S.Bradley 等人提出了一种利用聚类数据集，然后挑选许多子集样本的算法，利用 K-均值聚类算法来计算子样本的初始聚类中心。不过这算法还是受限于到样本子集挑选方法，不能完全脱离。另外的学者探究过其他方法来选择初始聚类中心，比如 CLARANS 算法，随机搜取初始聚类中心所对应的局部极小值，彼

此之间相互对比，然后选取出全局最优解。下面我们介绍另一种方法，那就是基于支持向量机的 K-均值聚类算法，初始值的选择对于这个算法没有那么强烈的影响，初始值的选择都是随机的无规律的。

### 3.3.1 支持向量机

支持向量机 (SVM) 是由 Vapnik 等人研究出来的一种学习算法，它是建立在统计学理论基础上的新型机器学习方法，用来处理模式识别的支持向量方法。支持向量机规避了学习算法执行的复杂程度同样本的维数的相互关联，运用核函数，控制超平面的间隔度量来抑制函数的过拟合。目前为止，支持向量机凭借其自身的特点在机器学习领域受到了普遍的关注，国内外学者都在研究这一新型的统计学习理论算法。

### 3.3.2 最优分类面

最大间隔分类器作为支持向量机里面，十分通俗易懂并且是最早被研究者们发布出来的模型，作为支持向量机的重要组成部分，是这一类型学习器的主要特性。最优分类面不可以在很多实际性的问题中被运用，因为它一般只是用在特征空间里面一些可以具有线性可分属性的数据中，但是支持向量机改善了这一不足，它就是根据求解线性可分条件下的最优超平面演变得出的。

下面介绍一下最优分类超平面，将不同的两种分类不出差错的分割开来，并且这两种分类之间的距离保证最大化。我们在确保经验风险能取到最小值，即  $R_{\text{emp}}(w) = 0$ ，同时能使得置信范围  $\Phi(h/n)$  的值最小，这样就能保证真实风险  $R(w)$  最小。下面我们用来表示一下这一思想：

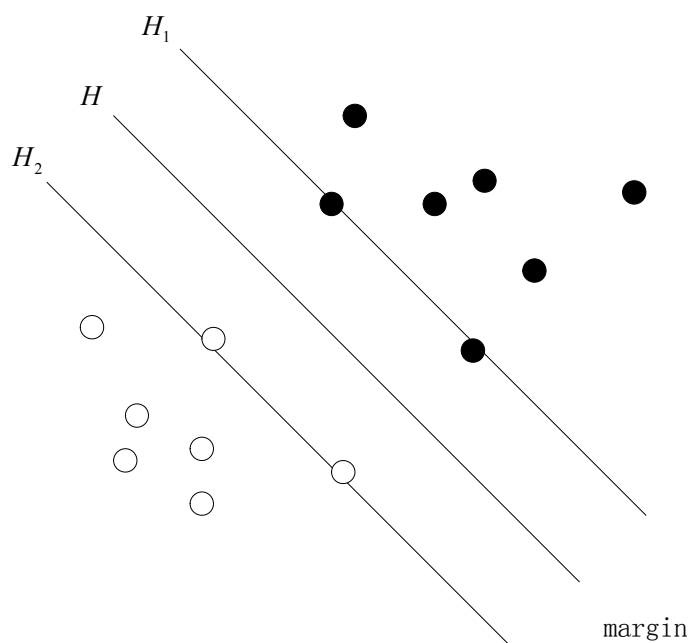


图 3.3 最优分类面示意图

对于样本集  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n), x \in R^n, y \in \{-1, 1\}$ ，支持向量机寻找一个最优超平面，这样能够让它的分类间隔最大。上图中，这两类样本分别用黑圆点和白圆点来表示，在三条直线中， $H$  是那条毫无偏差的将两类分割开来的分类线， $H_1$ 、 $H_2$  则各自作为这两类样本里面距离分类线最近同时与分类线相互平行的直线，他们之间的距离  $\text{margin}$ ，被称作分类间隔。假设分类线的方程式为  $x \cdot w + b = 0, y \in \{-1, 1\}, y_i[(w \cdot x_i) + b] \geq 0, i = 1, 2, \dots, n$ ，那么分类间隔为  $2/\|w\|$ ，分类间隔最大就意味着  $\|w\|$  的值是最小的，所以当满足  $2/\|w\|$  为最大值的时候的分类线，就是我们所需要的最优分类线，其中  $H_1$ 、 $H_2$  上面的训练样本点是支持向量。

### 3.3.3 支持向量机的分类

#### 1. 线性分类

支持向量机定义了最优线性超平面，然后利用求解二次规划问题来解决如何找到最优线性超平面这个问题，然后根据 Mercer 定理，将样本空间运用非线性映射来映射到高维特征空间，这样我们就通过线性方法处理了样本空间里面的高度非线性的难题。

支持向量机是根据二类别分类所研发出来的。如果给定训练样本  $\{x_i, y_i\}$ ， $i = 1, 2, \dots, l, x \in R^d, y \in \{-1, 1\}$ ，存在分类超平面  $x \cdot w + b = 0$ ，如果想要让分类面能够把所有样本进行正确的分类，同时兼顾分类间隔，那么就一定满足下面的条件：

$$y_i[(w \cdot x_i) + b] - 1 \geq 0 \quad 3-(24)$$



可以计算出分类间隔为：

$$\min_{\{x_i|y_i=+1\}} \frac{w \cdot x_i + b}{\|w\|} - \min_{\{x_i|y_i=-1\}} \frac{w \cdot x_i + b}{\|w\|} = \frac{2}{\|w\|} \quad 3-(25)$$

按照需求条件，如果最大分类间隔为  $2/\|w\|$ ，那就是说明  $\|w\|$  的值是最小值。因此解决最优分类平面我们就能将其转换为约束最优问题，即在上式的约束条件下，我们来把函数最小化

$$\psi(w) = \frac{1}{2} \|w\|^2 = \frac{1}{2} (w \cdot w) \quad 3-(26)$$

引入 *Lagrange* 函数后：

$$L = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^l \alpha_i \quad 3-(27)$$

式中， $\alpha_i > 0$  是 *Lagrange* 的系数。在上式中，分别对  $w$  和  $b$  求出其偏导数，然后让这俩偏导数变成 0，这样一来，我们就能把上面的问题转换成为相对容易的对偶问题。

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^l \alpha_i y_i x_i = 0 \quad 3-(28)$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^l \alpha_i y_i = 0 \quad 3-(29)$$

将式（3.5）和式（3.6）带入式子（3.4）里面去，就能够获得对偶最优化问题，即求解下面函数的最大值：

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad 3-(30)$$

$$s.t \quad y_i [(w \cdot x_i) + b] - 1 \geq 0 \quad 3-(31)$$

$$\sum_{i=1}^l y_i \alpha_i = 0 \quad 3-(32)$$

$$0 \leq \alpha_i \leq C, \xi_i \geq 0, i=1, 2, \dots, l \quad 3-(33)$$

其中， $C > 0$  是一个常数，被称作误差惩罚参数，它的作用是用来控制对错分样本惩戒的程度； $\xi_i$  是在训练样本线性不可分时，我们对样本加入的非负松弛变量。

## 2. 非线性分类

相比与线性分类，非线性分类没有那么负责，只要选取恰当的内积函数  $K(x_i \cdot x_j)$ ，我们一样能得到非线性变换之后的线性分类，这样的话我们需要改良的目标函数就变成了

$$Q(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j K(x_i \cdot x_j) \quad 3-(34)$$

与此同时，与之相对应的分类决策函数的表达式为

$$f(x) = \text{sgn} \left( \sum_{i=1}^l \alpha_i^* y_i K(x_i \cdot x_j) + b^* \right) \quad 3-(35)$$

上面的分类决策函数，即为支持向量机。

## 3.4 最小二乘法

### 3.4.1 最小二乘法的概述

最小二乘法在 18 世纪被高斯首先在其著作中发表，作为一个古老的算法，它被运用在天文观测等工作中。在随后的二百年中，最小二乘法展现出它在解决实际工程问题上的能力，被广泛运用在科学实验中。随着科学技术的发展，最小二乘法被运用到许许多多其他领域，比如神经网络、物理结构、经济和机械等等，展现出了它强大的生命力和适应性。

### 3.4.2 最小二乘法的原理

最小二乘法的基本原理是采用最小化算法对它所构造的误差函数进行计算，并得到最小值（不是局部最小值，而是最好的全局最小值）。

对于一个矩阵方程

$$Tf = g \quad 3-(36)$$

上式中， $T$  是  $M \times N$  阶矩阵， $f$  是  $N$  维列向量， $g$  是  $M$  维列向量。假设已经知道  $T$  和  $g$  需要求解，当  $T$  是非奇异矩阵时，我们能够对  $T$  求逆来直接进行求解。

通常情况下，矩阵  $T$  往往是奇异矩阵，而不是方阵，这样的情况下，我们就不能利用求逆的方式来直接求方程的解。但是我们可以用奇异值分解求解其最小二乘解的办法来对方乘求解，方法如下：

首先，我们对矩阵  $T$  进行奇异值分解，可得

$$T = U \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} V^H \quad 3-(37)$$

上式中， $U$  是  $M \times M$  阶矩阵， $V$  是  $N \times N$  阶矩阵，上标  $H$  表示复数共轭转置， $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$  是  $r \times r$  阶对角阵，同时  $r \leq \min\{m, n\}$ 。 $\sigma_i (i=1, 2, \dots, r)$  是矩阵  $T$  的奇异值。

如果  $U = [U_1 | U_2]$ ，其中  $U_1$  是  $U$  的前  $r$  列向量构成的  $M \times r$  阶矩阵，它包括矩阵  $T$  的左奇异向量； $V = [V_1 | V_2]$ ，其中  $V_1$  是  $V$  的前  $r$  列向量构成的  $N \times r$  阶矩阵，它包括矩阵  $T$  的右奇异向量。 $U_1$  和  $V_1$  是酉矩阵，即

$$U_1^{-1} = U_1^H \quad 3-(38)$$

$$V_1^{-1} = V_1^H \quad 3-(39)$$

这时，我们能够获得

$$T = U_1 \Sigma V_1^H \quad 3-(40)$$

结合上式，可以获得矩阵  $T$  的广义逆是

$$T^+ = V_1 \Sigma^{-1} U_1^H \quad 3-(41)$$

其中

$$\Sigma^{-1} = \text{diag}\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \dots, \frac{1}{\sigma_r}\right) \quad 3-(42)$$

因此，方程的最小二乘解为：

$$f = T^+ g = V_1 \Sigma^{-1} U_1^H g \quad 3-(43)$$

### 3.5 本章小结

本章主要介绍了 K-均值聚类和支持向量机的基本原理，详细描述了 K-均值聚类的优点以及缺点，其可以防止局部最优解的特点可以运用在神经网络里面对算法进行改进。支持向量机解决了在神经网络里面无法避免的局部极小值问题，为下文的实验做好理论基础。



## 第四章 基于支持向量机和 K-均值聚类的 RBF 短时交通流预测仿真实验

### 4.1 基于 K-均值聚类的 RBF 神经网络的学习算法

K-均值聚类算法有许多优点，比如运算速度快、思路清晰简单等等，因此被学者们大量运用于日常实验和科学研究之中。对于 RBF 神经网络中心的选取这个过程，K-均值聚类扮演着十分重要的角色。根据 K-均值聚类算法对初始聚类中心十分敏锐的特性，一般情况下我们可以挑选许多不一样的初始中心值，不需要特意挑选，随机即可，运行算法后从里面选取比较好的处理结果。

在这里我们选取  $L$  组输入样本  $x_p$ ，其中  $d_p$  为各个组里面的期望输出。然后把目标误差的函数用下式来作定义：

$$E = \frac{1}{2} \sum_{p=1}^L \|d_p - y_p\|^2 \quad 4-(1)$$

上式里面， $y_p$  为输出向量，并且它和输入向量  $x_p$  有关。

#### 4.1.1 K-均值聚类算法确定基函数中心的方法

K-均值聚类算法原理简单易用，下面我们运用它来将全部的输入样本，快速准确地分析处理然后完成统一聚类，求得全部隐含层节点的 RBF 中心值  $C_i$ 。计算步骤如下所示：

Step1: 给定各隐节点的初始中心  $C_i(0)$ ，通常取输入样本向量的前  $k$  个值。

Step2: 计算欧式距离：

$$d_i(t) = \|x(t) - C_i(t-1)\|; i = 1, 2, \dots, k. \quad 4-(2)$$

Step3: 求出最小距离节点：

$$d_r(t) = \min d_i(t) \quad 4-(3)$$

Step4: 调整重心：

$$C_i(t) = C_i(t-1); 1 \leq i \leq k, i \neq r \quad 4-(4)$$

$$C_r(t) = C_r(t-1) + \beta [x(t) - C_r(t-1)]; i = r. \quad 4-(5)$$

Step5: 判别：

如果  $C_i(t) = C_i(t-1)$ ，则终止迭代，得到最终的输出，倘若不满足，继续跳转 Step2。

#### 4.1.2 采用最小二乘法对 RBF 网络权值的处理

中心值  $C_i$  的确定，就意味着我们可以进行下一步权值的线性优化问题来处理了，此处的权值是位于隐含层和输出层之间的，因此可采用最小二乘法，公式如下：

$$w = \exp\left(\frac{M}{c_{\max}^2} \|x_p - c_i\|\right) \quad p = 1, 2, \dots, P; i = 1, 2, \dots, M \quad 4-(6)$$

其中， $w$  为权值，属于隐含层和输出层之间的神经元； $c_{\max}$  是我们所挑选的中心相互之间的距离的最大数值； $p = 1, 2, \dots, P$  代表的是样本的数量；与此同时， $i = 1, 2, \dots, M$  便是隐含层的所拥有的结点数目。

## 4.2 交通流量的数据来源

影响车流量的因素有很多，例如节假日、天气、是否周末等等因素都有可能造成某一段时间内车流量的波动较大，因此我们选取了六个工作日的车流量来进行仿真，这六个工作日均没有恶劣天气的影响，也不是正常的节假日，因此对车流量的影响不会十分剧烈。与此同时，我们还应该考虑到每天不同的时间车流量也是不同的，同时某些时段的车流量是我们不需要的，并且其波动性也比较大，比如凌晨这段时间，这段时间的路况通常来说不会很差，基本是畅通无阻的，因此我们在数据采集的时候，对于晚上九点至凌晨六点这 9 个小时的车流量数据是忽略的。因此这次数据样本的组成，是早上 6 点到晚上 9 点的实际车流量，样本采集地点为青岛市某主干道路，总共采集了连续六个周二的车流量，这六天都是正常的工作日，没有法定的节假日等情况。数据太多显得太过冗杂，数据太少又无法说明算法的优劣，因此我们设定每隔 10 分钟统计一次车流量，每小时统计 6 次，每天统计 90 次。每天的统计数据都在当天校验完毕，保证其有效性。

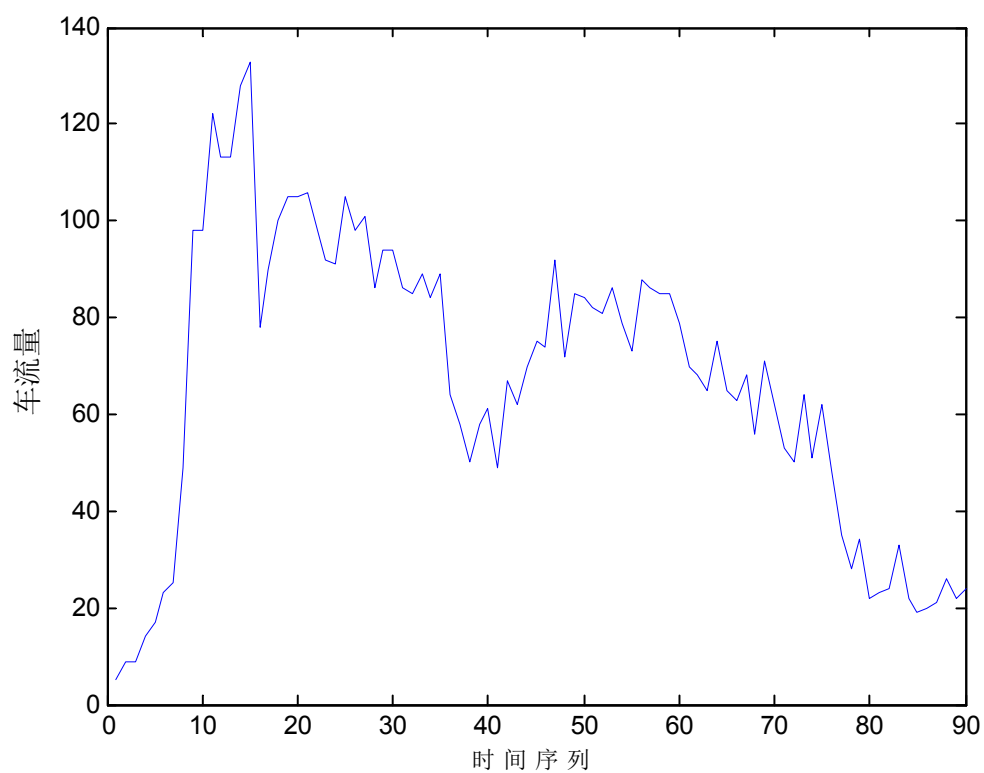


图 4.1 某一个周二的车流量

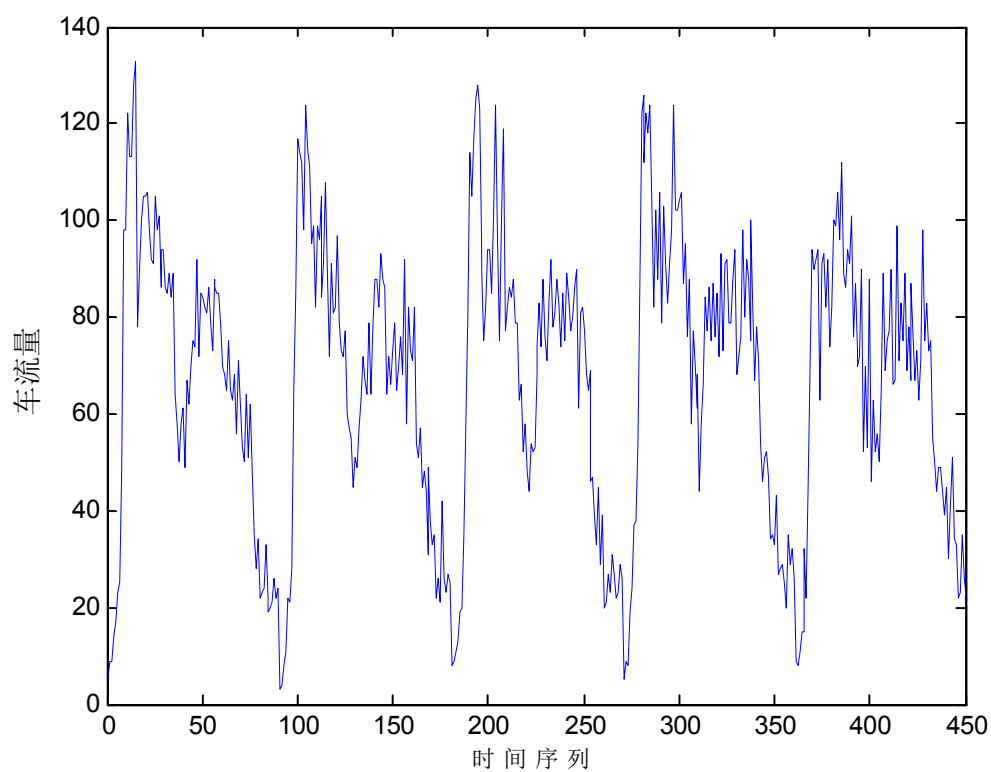


图 4.2 五个周二的车流量

其中，选取前五个周二的车流量数，用来当作 RBF 神经网络的训练数据；同时将最后组的后 86 个数据，当作校验预测输出的实际输出。

### 4.3 计算机仿真

#### （1）对 RBF 神经网络建立合适模型

step1: 确定网络参数。在这里需要设定的参数有三个，分别是输入层节点数、隐含层节点数以及输出层节点数。我们把利用每天 90 个数据的前四个来预测第五个数据，因此设定输入节点数为 4；隐含层节点数不能过高或者过低，根据经验我们将隐含层节点数设置为 11；最后的输出只有一个，所以很明显输出层节点数为 1。根据上面三个参数，我们就可以建立预测模型为 4-11-1 的 RBF 神经网络。

step2: 首先我们将聚类中心给初始化，随后依照前面的 K 均值聚类算法，通过计算得出基函数中心  $C_i$ ；然后再判断下中心的位置是否发生改变，倘若没有变化，我们就利用上面的公式，选择最小二乘法算出权值  $w$ ；倘若发生变化，那么就重复上面的步骤，直到确定基函数中心  $C_i$  不会发生改变。

step3: 将算法和 MATLAB 编程相互结合，编写程序后，读取相应数据，得到运算后的图像，最后用输出的期望数据与实际数据相互对比。

#### （2）进行初步试验

第一次实验我们分为两个阶段：第一阶段，我们的算法未加入 K 均值聚类，单独的使用 RBF 神经网络来进行短时交通流的预测，来验证 K-均值聚类算法是否有效；第二阶段，我们将 K-均值聚类融合到 RBF 神经网络中去，其它条件没有变化，这样一来就能够对两种结果进行比对分析，可以更好地理解 K-均值聚类算法在短时交通流预测上所发挥的作用。

其中，没有融合 K-均值聚类算法的 RBF 神经网络，我们编程然后通过 MATLAB 仿真运算，结果如下图所示：



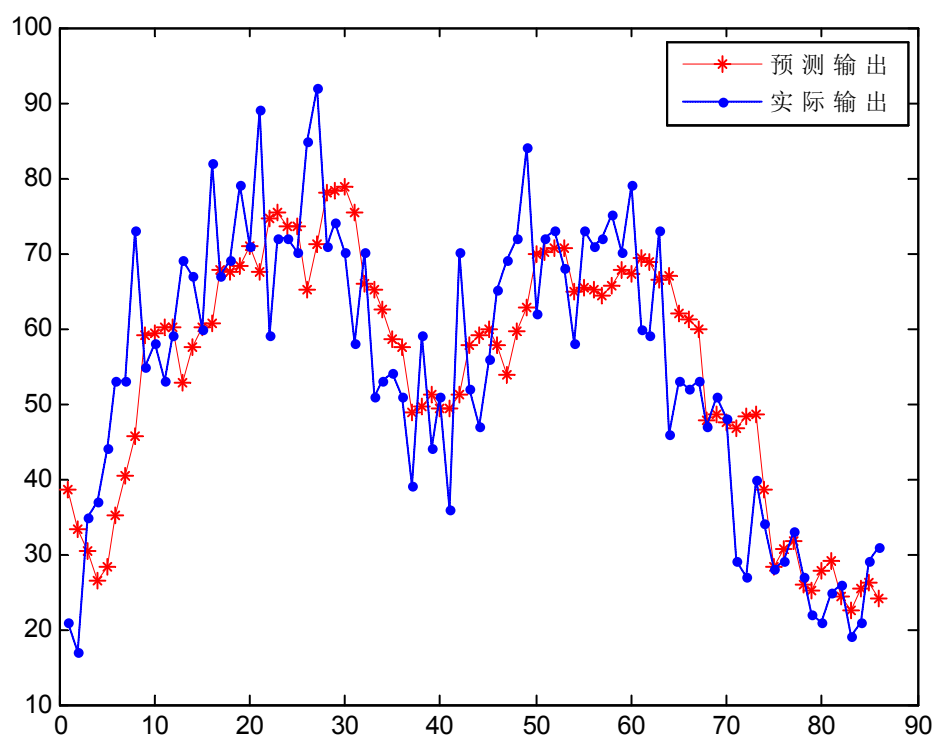


图 4.3 未采用 K 均值聚类算法的预测输出和实际输出

图中，原点为实际输出，星点为预测输出，很明显预测输出在实际输出波动较大的时候，他们之间的偏差也很大。虽然大致曲线走向一致，但是拟合程度不高，预测曲线在收敛速度方面差强人意。

下面我们将 K-均值聚类算法融合到 RBF 神经网络中去，所得数据图像如下：

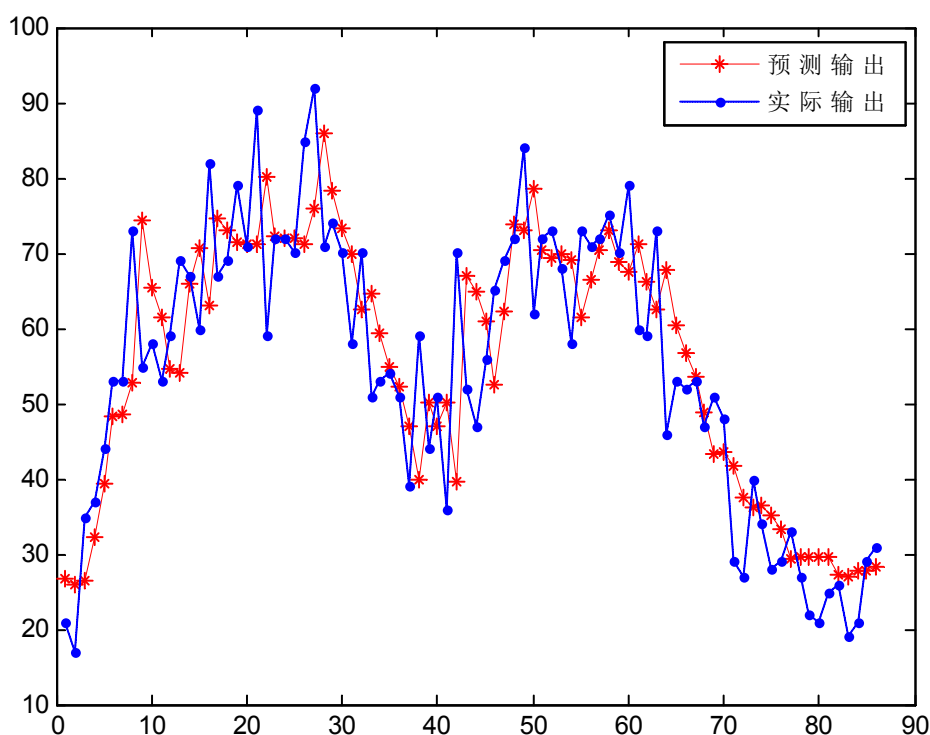


图 4.4 基于 K 均值聚类算法的 RBF 神经网络的预测输出和实际输出

经过基于 K 均值聚类算法的 RBF 神经网络的训练后,预测输出和实际输出的曲线它们之间的拟合程度明显增高不少,在实际输出数据的大幅波动的情况下,预测输出的曲线无论是收敛速度,还是预测的准确度都有着明显提高,但是在小范围数据波动的情况下,其准确性还是有待加强,尤其是最后几组预测,数据点聚拢在一起,明显是算法出现偏颇导致的。

#### 4.4 基于支持向量机和 K-均值聚类的 RBF 短时交通流预测

针对上面的不足,我们利用支持向量机来对 K-均值聚类算法进行优化,将两者相互结合,这样就形成了一种改进的 K-均值聚类算法。下图为改进算法后的预测和实际输出:

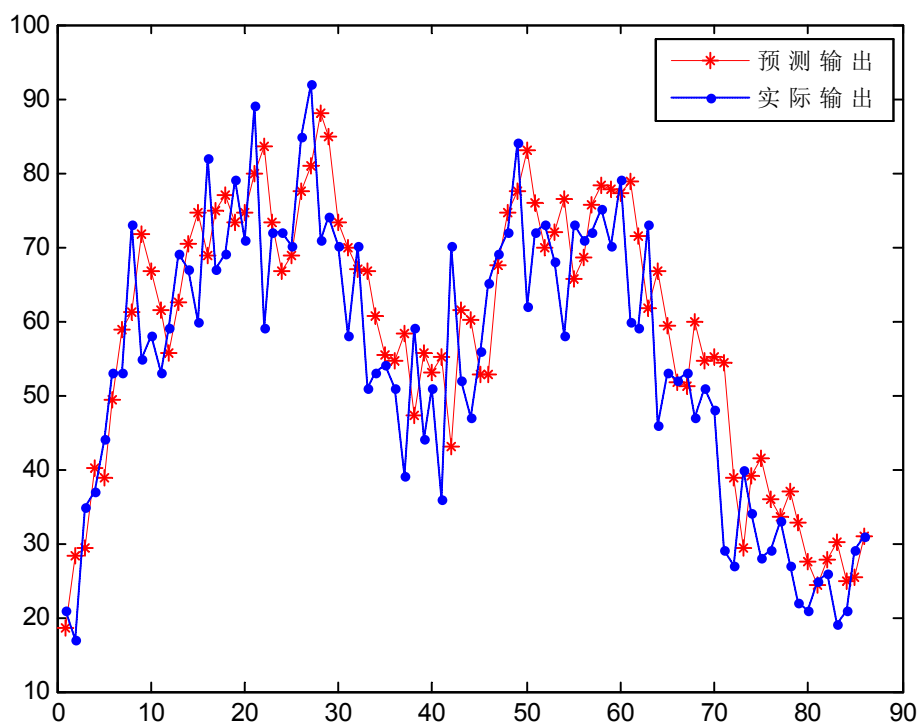


图 4.5 加入 SVM 算法后的预测输出和实际输出对比

因此我们能够得出一个结论，经过支持向量机改进后的 K-均值聚类算法所预测的数据更加准确可靠，预测值更接近实际的交通流，效果最佳。

#### 4.5 本章小结

本章中，我们通过对真实数据的仿真预测，然后将预测结果和实际数值相对比，同时结合相关的理论基础，首先验证了 RBF 神经网络在短时交通流预测上是可行的；其次我们克服了 K-均值聚类算法的某些缺点，与支持向量机相结合，建立了预测性能更好的神经网络模型，验证了算法的可行性



## 结论与展望

短时交通流预测已经逐渐成为改善城市交通环境的一种有效的措施，具有十分重要的研究意义。凭借神经网络强大的学习能力，能够很好的实施短时交通流的预测，为智能交通系统提供可靠的数据支持，帮助其进行路径规划、车流引导等等，具有一定的实际意义。本文主要总结工作如下：

（1）介绍了短时交通流预测的背景，即现代城市交通网络建设的重要性，同时介绍了国内外学者对于短时交通流预测领域的研究状况，简单描述了几种算法的优点和应用。

（2）介绍了神经网络的构成，同时阐明了 BP 神经网络和 RBF 神经网络的优缺点，为后面的实验选择打下了理论基础。

（3）介绍分析了 K-均值聚类算法的原理和优缺点，同时在最后引入支持向量机器学习算法。

（4）最后运用 Matlab 仿真，使用真实数据，建立了三种短时交通流预测模型，并对这三种模型产生的结果进行处理分析。

虽然本文对于支持向量机和 K-均值聚类算法在短时交通流预测的应用研究取得了一定的进展，但是限于个人能力以及时间原因，在课题研究的过程中仍存在一些亟待解决的问题。比如对现在算法中依旧存在的问题实施改善，几天的实时交通流数据是否过少等问题。



## 参考文献

- [1] 杨兆生. 智能运输系统概论[M]. 北京: 人民交通出版社. 2003.
- [2] 魏文. 基于混沌和 PSO 神经网络的短时交通流预测研究[D]. 成都:西南交通大学, 2009.
- [3] 秦楠. 面向出行者的出行诱导系统研究与设计[D].青岛: 青岛大学,2011.
- [4] 中国智能运输系统体系框架专题组. 中国智能运输系统体系框架[M]. 人民交通出版社.2003.
- [5] Smith, B. L., Williams, B .M. Oswald, R. K-Comparison Of Parametric And Non-Parametric Models For Traffic Flow Forecasting[J]. Transportation Research Part C, 2002, 10:303-321.
- [6] 谭国贤, 翁小雄, 姚树申. 城市交叉路口的短时交通流建模预测[J]. 交通与计算机, 2005, 23(1): 27-28.
- [7] 王正武, 黄中祥. 短时交通流预测模型的分析与评价[J]. 系统工程, 2003, 21(6): 97-100.
- [8] H. Nicholson and C. D. Swann. The prediction of traffic flow volumes based on spectral analysis. [J]. Transportation Research, 1974, 8(6):533-538.
- [9] Ahmaed Mohamed S, Cook Allen R. Analysis of freeway traffic time-series data by using Box-Jenkins technique[J]. Transportation Research Record 722, 1979: 1-9.
- [10] Guoqiang Yu, Changshui Zhang. Switching ARIMA model based forecasting for traffic flow[J]. IEEE, 2004.
- [11] 李嘉, 刘小兰. 基于卡尔曼滤波理论的交通信息预测模型及其应用[J]. 中南公路工程.2005, 12, 30(4): 108-109.
- [12] 佩林, 余志, 何兆成. 基于约束卡尔曼滤波的短时交通流量组合预测模型[J]. 交通运输工程学报. 2008, 10, 8(5): 86-90.
- [13] Okutani I, Stephanedes Y J. Dynamic Prediction of traffic volume through Kalman filtering theory[J]. Transportation Research(B), 1984: 1-11.
- [14] Davis Gary A.and Nihan Nancy L. Nonparametric Regression and Short-term Freeway Traffic Forecasting[J]. Journal of Transportation Engineering, 1991, 117(2): 178-188.
- [15] Smith Brian L, Demrsky Michael J. Short-term Traffic Flow Prediction:Neural Network Approach[J]. Transportation Research Record 1453, Washington, D.C.TRB, 1993.
- [16] der Voort Maschavan, Dougherty Mark and Watson Susan. Combining Kohonen Maps with ARIMA Time Series Models to Forecast Traffic Flow[J]. Transportation Research, 1996, 4(C): 307-318.
- [17] orinne Ledoux. An Urban Traffic Flow Model Integrating Neural Network[J]. Transportation System Research, 1997, 5(5): 287-300.
- [18] 杨兆升. 基于卡尔曼滤波的事实交通流量预测模型. 中国公路学报[J]. 1998.
- [19] 朱中, 杨兆升.实时交通流量人工神经网络预测模型. 中国公路学报[J]. 1998.
- [20] 李存军, 杨儒贵, 靳蕃. 基于神经网络的交通信息融合预测方法[J]. 系统工程. 2004, 22(3): 80-83.
- [21] 刘洁, 魏连雨, 杨春风. 基于遗传神经网络的交通量预测[J]. 长安大学学报. 2003, 55-57.
- [22] 王凡, 谭国真等. 并行广义神经网络的交通流预测[J]. 计算机工程与应用, 2010, 46(17): 229-231.
- [23] 蔡珥. 短时交通流量预测的 IPSO-BPNN 算法[J]. 计算机工程与应用, 2012, 48(27): 239-243.
- [24] J. W. Gao, G Q. Cai, Q. C. Li, et al. Short-term urban road traffic flow forecasting based on ANTIS[J]. ICIC Express Letters, 2011, 5(10):3671-3675.

- [25] 华冬冬. 短时交通流的混沌性分析及其基于神经网络的预测模型研究[D]. 南京: 东南大学, 2005.
- [26] 王进, 史其信. 短时交通流预测模型综述[J]. ITS 通讯, 2005, (1): 10-13.
- [27] Iwao Okutani. Dynamic Prediction of traffic volume Through Kalman Filtering Theory[J]. Transportation Research, 1984, 1: 1-11.
- [28] 杨兆升, 朱中. 基于卡尔曼滤波理论的交通流量实时预测模型[J]. 中国公路学报, 1999, 12(3): 63-67.
- [29] 聂佩林, 余志, 何兆成. 基于约束卡尔曼滤波的短时交通流量组合预测模型[J]. 交通运输工程学报, 2008, 8(5): 6-9.
- [30] Vythoulkas P C. Alternative approaches to short term traffic forecasting for use in driver in-formation[A].University of Oxford 12th International Symposium on the Theory of Traffic Flow and Transportation. Elsevier Science, 1993: 485-506.
- [31] Dochy Thierry,Danech-Pajouh Mehdi and Lechevallier Yees. Short-term Traffic Forecasting Using Network[J]. Theory and Application of Advanced Technology, 1994: 633-639.
- [32] Ledoux Corinne. An Urban Traffic Flow Model Integration Neural Networks[J]. Transportation Research, 1997, 5: 307-318.
- [33] 姚亚夫, 刘侃. 基于遗传神经网络的交通流量预测研究[J]. 公路与汽运, 2007, (6): 28-30.
- [34] 张九跃, 焦玉栋. 基于 RBF 神经网络的短时交通流量预测[J]. 山东交通学院学报, 2008, 16(3): 22-26.
- [35] 陈雪平, 曾盛, 胡刚. 基于 BP 神经网络的短时交通流预测[J]. 公路交通技术, 2008, 3:115-117.
- [36] Chang S. C, Kim. R. S, Kim. S. J, Ann B. H. Traffic-flow forecasting using a 3-stage model[J]. IEEE Intelligent Vehicles Symposium Proceedings 2000, 5: 422-430.
- [37] 周欣荣.高速公路短时段交通流预测方法研究. [D]. 江苏. 东南大学. 2006. 3-1.
- [38] 聂佩林.高速公路交通流预测方法研究. [D]. 广州. 中山大学. 2009. 9-18.
- [39] 彭信林. 城市快速路交通状态预测研究[D]. 上海交通大学, 2008.
- [40] D. O. Hebb. The organization of behavior [M]. New York: Wiley,1949.
- [41] F. Rosenblatt The perception: a probabilistic model for information storage and organization in the brain[J]. Psychological Review, 1958, 65(6): 386-408.
- [42] Simon Haykin. Neural networks, a comprehensive foundation, second edition, Prentice Hall, 1998. 161-175, 400-438.
- [43] 戴葵. 神经网络实现技术[M]. 长沙: 国防科技大学出版社, 1998.
- [44] 焦李成. 神经网络系统理论[M]. 西安: 西安电子科技大学出版社, 1996.
- [45] 袁曾任. 神经网络原理及其应用[M]. 北京: 清华大学出版社, 1999.
- [46] Rumelhart D.E,Hinton G.E and Williams R.J. Learning Internal Representations by Error Propagation. Parallel Distributed Processing: Explorationsin the Microstructures of Cognition.MIT Press, MA.1986. 318-362.
- [47] 朱铁夫, 李明, 邓建华. 具有重构功能的基于 RBF 神经网络直接自适应飞控系统[J]. 西北工业大学学报, 2005. 23(3): 311-315.
- [48] 海金 S 著. 叶世伟等译. 神经网络原理(第二版)[M]. 北京: 机械工业出版社, 2004.
- [49] 穆云峰. RBF 神经网络学习算法在模式分类中的应用研究 [D]. 大连: 大连理工大学, 2006.
- [50] 新波. 模糊聚类分析及其应用[M]. 西安电子科技大学出版, 2004.



- [51] 吴晓蓉. K-均值聚类算法初始中心选取相关问题的研究 [D]. 湖南大学硕士学位论文, 2008.
- [52] 刘炜. 聚类分析在高校就业工作中的应用研究[D]. 东北林业大学, 2011.



## 攻读学位期间的研究成果

### 攻读学位期间发表的学术论文

1. 管硕, 高军伟, 张彬, 等. 基于 K-均值聚类算法 RBF 神经网络交通流预测[J]. 青岛大学学报(工程技术版), 2014, 29(2): 20-23.



## 致 谢

三年的研究生生活即将结束，在这里首先要对我的导师高军伟教授表示感谢，不管是在学术研究中还是日常生活中，高老师都给予了我很大的帮助。在论文即将完结之际，感谢高老师从论文的开题、中期答辩、论文的架构以及算法的讲解中对我的帮助，高老师渊博的学识，正直的人品，工作中的一丝不苟，生活中的勤劳朴实，都在我心里刻下深深地烙印，这些美好的品质会让我在以后的工作和生活中向老师看齐，做一个正直勤劳踏实肯干的人。

感谢北京交通大学的姚德臣师兄。在广州地铁学习的那段日子感谢姚师兄对我专业知识方面的解答以及日后找工作上的指点迷津。

感谢实验室中的冷子文、马增涛和李刚师兄，感谢成亮、阚枢和刘赫同学，感谢师弟张盼栋、林怀民和张海峰，感谢师妹赵红、季雪美和郑依，感谢你们在生活和学习上的帮助。

感谢我的父母，对于我学业的支持对于我生活的支持对于我一切一切的支持。

感谢评审论文的专家和老师们。



## 学位论文独创性声明

本人声明，所呈交的学位论文系本人在导师指导下独立完成的研究成果。文中依法引用他人的成果，均已做出明确标注或得到许可。论文内容未包含法律意义上已属于他人的任何形式的研究成果，也不包含本人已用于其他学位申请的论文或成果。

本人如违反上述声明，愿意承担由此引发的一切责任和后果。

论文作者签名：吕硕

日期：2015年6月9日

## 学位论文知识产权权属声明

本人在导师指导下所完成的学位论文及相关的职务作品，知识产权归属学校。学校享有以任何方式发表、复制、公开阅览、借阅以及申请专利等权利。本人离校后发表或使用学位论文或与该论文直接相关的学术论文或成果时，署名单位仍然为青岛大学。

本学位论文属于：

保密 ☐，在          年解密后适用于本声明。

不保密 ☒。

（请在以上方框内打“√”）

论文作者签名：吕硕

日期：2015年6月9日

导师签名：高伟

日期：2015年6月9日

（本声明的版权归青岛大学所有，未经许可，任何单位及任何个人不得擅自使用）