

天津大学

硕士学位论文

基于支持向量机参数优化的群智能优化算法研究

姓名：李艳英

申请学位级别：硕士

专业：模式识别与智能系统

指导教师：路志英

20070601

摘要

群智能算法是人们从生物行为中得到启发,与计算机技术相结合用于解决现实生活中遇到的复杂问题的方法,具有容易理解、易于实现等特点,已在很多领域得到了广泛应用,本文的研究是针对沙尘暴预测进行的,是对其应用领域的拓展。

沙尘暴的频繁发生给人类的生活和社会生产带来了巨大危害,沙尘暴的预测问题早就引起了研究者的关注,目前已出现了将支持向量机分类器应用到沙尘暴预测中的研究,但对支持向量机参数的选取仍停留在依靠经验手动调节的水平上,影响对沙尘暴预测的准确率。在以上研究背景下,本文首次将遗传算法、粒子群算法引入沙尘暴的预测模型——RBF核支持向量机参数优化过程中,对以遗传算法、粒子群算法为代表的群智能优化算法进行了进一步的研究。

本文所作的主要研究工作如下:

1. 对遗传算法和粒子群算法进行了进一步的分析,并分别运用遗传算法和粒子群算法对支持向量机的参数进行了优化。仿真结果表明粒子群算法具有比遗传算法更好的性能。

2. 对粒子群算法进行了收敛性分析,得出了使得算法收敛其参数之间满足的条件,在此基础上提出了粒子群算法的改进算法。针对粒子群算法容易陷入局部极值点的缺点,在迭代之初选用比较大的惯性权值,随着迭代的进行惯性权值线性减小,保证了算法在开始时进行的是全局搜索,在后期进行的是小范围的局部搜索。学习因子的设定在满足收敛条件的约束下进行。仿真结果表明:改进粒子群算法具有更好的全局收敛性,寻优准确率提高。

本文实现了运用智能优化方法对支持向量机的参数进行寻优,提高了沙尘暴预测的准确率,推动了群智能研究的发展。

关键词: 支持向量机 核函数 遗传算法 粒子群算法 群智能 沙尘暴预测

ABSTRACT

Swarm intelligence algorithm is a method enlightened from the behaviors of biology and combined with computer technology to solve complicated problems appearing in the real life. It is easy for us to comprehend and achieve so that it has been used in many fields. The study in this paper aiming at the sand-dust storm forecasting enlarges the application fields of swarm intelligence algorithm.

The high frequency of the sand-dust storm destroyed people's daily life and social production, the sand-dust storm forecasting is attracting for researchers. Support vector machine has been used to forecast the sand-dust storm, but the SVM's parameters are still selected using manual method, depending on experience, this effects the forecasting accuracy. In the background of the research mentioned above, for the first time, in this paper GA and PSO are used in the parameter's optimization of sand-dust storm forecasting model——RBF-SVM, and take GA and PSO as examples to make a deep research about swarm intelligence algorithm.

The main researching contents of this dissertation are as follows:

1. A deep research of GA and PSO is made and used respectively in the parameter's optimization of SVM. The results of the experiment show that the performance of PSO is better than GA.

2. The constringency of PSO is analyzed, the condition of the algorithm's constringency and the relationship of the parameters is obtained. An improved PSO algorithm is proposed. aiming at the shortcoming that PSO is easy to get into part extremum, greater ω is used at the beginning, as the algorithm is running ω is decreasing according to a linear rule, this ensures the algorithm searches entirely at the beginning and searches partly in the end. The study parameters c_1 , c_2 are selected under the constringency condition. The simulations' results show that the improved algorithm is better in the entire constringency and accuracy.

Searching the best parameters of SVM using swarm intelligence algorithm comes true in this paper, this improves the accuracy of the sand-dust storm forecasting, promotes the development of swarm intelligence algorithm.

KEY WORDS: Support Vector Machine, kernel function, generation algorithm, particle swarm optimization, the sand-dust storm forecasting

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 天津大学 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名：李艳英 签字日期：2007年6月7日

学位论文版权使用授权书

本学位论文作者完全了解 天津大学 有关保留、使用学位论文的规定。特授权 天津大学 可以将学位论文的全部或部分内容编入有关数据库进行检索，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：李艳英

导师签名：路志英

签字日期：2007年6月7日

签字日期：2007年6月7日

第一章 绪论

遗传算法 (Genetic Algorithm, GA)、粒子群 (Particle Swarm Optimization, PSO) 算法都是模拟自然界的生物现象, 用来解决现实生活中遇到的复杂问题的群智能算法。鉴于各自在不同方面所特有的优点, 以它们为代表的群智能算法已被广泛应用于计算机科学、人工智能、信息技术及工程实践等领域。虽然它们都有鲜明的生物学基础, 但理论研究尚不够完善, 算法的性能方面也存在很多的不足, 这就促使了研究者对群智能算法的理论和性能及应用领域作进一步的研究。

1.1 引言

实际生活中有很多优化问题需要在复杂而庞大的搜索空间中寻找最优解或准最优解, 其中包括线性的, 非线性的; 连续的, 离散的; 单峰值的, 多峰值的等多种类型。求解最优化问题的方法有多种形式, 不同的优化方法具有不同的优化特性。有些优化方法可以快速求解到局部最优解, 有些优化方法可以得到全局最优解, 但寻优的时间耗费会长一些。最优化问题求解的理想情况是快速有效的得到全局最优解, 但这种情况只有在有限的条件下才能实现。随着对优化问题研究的不断深入, 对于复杂函数优化问题, 很难找到收敛性好且全局最优的优化方法, 因此人们就把着眼点放在了寻求近似最优解上。在求解优化问题时, 若不能利用问题的固有知识来缩小搜索空间, 则会产生搜索的组合爆炸。因此, 研究能在搜索过程中自动获得和积累有关搜索空间的知识, 并能自适应地控制搜索过程, 从而得到能寻得最优解的通用搜索算法一直是令人瞩目的课题。

人们从自然界中很多的生物演化现象中得到启示解决了难以实现的优化问题。近年来, 对自然界演化现象的研究与应用已经成为了人工智能领域的一大研究热点。通过对生物自然选择的进化过程、对蚂蚁觅食现象、对鸟群飞行规律以及对人类的免疫系统和生物的神经网络的一系列研究, 计算机学者们分别提出了遗传算法、蚁群算法、粒子群算法、免疫算法以及人工神经网络等人工智能领域的新的研究方向。

遗传算法和粒子群算法都是经实践证明特别有效的搜索算法。遗传算法采用简单的编码技术来表示各种复杂的结构, 并通过对编码进行简单的遗传操作来指导学习、确定搜索方向。粒子群算法也是一种基于种群的搜索算法, 它利用生物

群体内个体的合作与竞争等复杂行为产生群体智能,在解决优化问题时,该算法利用了生物群体中信息共享会产生进化优势的思想,通过个体之间的协作来寻找最优解,其中的每个个体都是一个潜在的最优解,经过反复迭代最终得到所需要的全局最优解。本文的研究在沙尘暴预测领域对遗传算法和粒子群算法进行的更加深入的研究。

1.2 本文的研究背景

1.2.1 沙尘暴预测

由于人类生活和社会生产对生态环境的破坏,沙尘暴已成为我国北方春季较为常见的一种灾害性天气。沙尘暴的危害极大,一旦发生,其携带沙砾的强劲气流所经之处,通过沙埋、狂风袭击、降温霜冻和污染大气等方式,便会使大片农田受到沙埋或被刮走沃土,或者农作物受霜冻之害,致使有的农作物绝收,有的大幅度减产;它能加剧土地沙漠化,对大气环境造成严重污染,对生态环境造成巨大破坏,对交通和供电线路等基础设施产生重要影响,使人民生命财产遭受严重损失。沙尘暴的形成与地球温室效应、厄尔尼诺现象、森林锐减、植被破坏、物种灭绝、气候异常等因素有着不可分割的关系。其中,人口膨胀导致的过度开发自然资源、过量砍伐森林、过度开垦土地是沙尘暴频发的主要原因。沙尘暴的形成有两个条件,一个是大风,另一个是沙漠。随着近几年沙尘暴的频繁发生,人类已经越来越认识到沙尘暴是不可忽视的生态环境的重要问题之一。

研究者在分析其成因的基础上加强了对沙尘暴预测的研究,但由于沙尘暴发生原因的复杂性,一直未能取得比较理想的预测结果。目前已出现了将支持向量机理论应用于沙尘暴预测的研究,但支持向量机的参数会对其分类效果产生很大的影响,而对其参数的调节尚停留在依靠经验手动调节的水平上,影响了对沙尘暴预测的准确率,本文的研究就是建立在基于支持向量机沙尘暴预测模型的基础上展开的。

1.2.2 支持向量机的参数优化

支持向量机(Support Vector Machine, SVM)理论作为近几年发展起来的一门新技术,是建立在统计学习理论的VC维理论和结构风险最小原理基础之上的,它能很好地解决小样本、非线性、高维数和局部极小点等实际问题^[1],目前已成为机器学习界的研究热点。鉴于以上优点,支持向量机已被应用于沙尘暴的预测

中^[2], 仿真结果表明: 无论是从效率上还是准确率上都取得了比BP网络好得多的预报结果, 验证了该分类器在沙尘暴预测中的有效性。但大量的实验也证明, 支持向量机的参数会对其分类效果产生很大的影响。因此, 支持向量机的参数优化是该领域学者一直关注的热点问题。目前常用的优化方法有试凑法、双线性搜索法、网格搜索法、交叉验证法、混沌优化算法等。

文献[2]中用于沙尘暴预测的支持向量机参数的选取是根据网格搜索法的原理, 用手动调节的方式来实现的, 随着参数个数的增加和调节幅度的增大, 手动调节的效率将会大幅度降低, 而且寻找到最优参数组合变得几乎不可能。因此寻求一种能以有限的代价来解决上述优化问题的通用方法具有重要的意义。遗传算法和粒子群算法等群智能算法为我们解决上述优化问题提供了一条有效的途径和新的思路。

1.2.3 群智能的发展

受社会性昆虫行为的启发, 计算机工作者模拟社会性昆虫行为产生了一系列解决传统问题的新方法, 这些研究就是对群智能的研究。群智能不是忠实的模拟自然现象, 而是从自然现象中得到启发来实际问题。群集智能中的群指的是“一组相互之间可以进行直接或间接通信的主体, 这些主体之间能够相互协作进行问题求解”。已有的对群智能理论和应用的研究表明群智能是一种能有效解决优化问题的方法, 它潜在的并行性和分布式等特点为处理大规模的数据提供了保证。

以遗传算法、粒子群算法等为代表的群智能算法, 主要有以下几个特点:

- (1) 群体中相互合作的个体是分布式的, 适合当代网络环境下并行的工作状态;
- (2) 不会由于某一个或某几个个体的故障而影响整个问题的求解, 具有鲁棒性;
- (3) 个体之间可以通过非直接通信进行相互合作, 系统中因个体的增加而增加的系统通信开销十分小, 使得系统具有很好的可扩充性;
- (4) 系统中每个个体的执行时间比较短, 实现起来比较简单。

目前, 群智能以其分布性、自组织性、强的鲁棒性等优点使得对它的研究虽然还处于初级阶段, 但已经被成功的应用于函数优化、神经网络等领域。其中遗传算法和粒子群算法作为群智能算法中的代表, 从一出现便引起了研究者的广泛关注, 其理论研究在不断深入, 其应用领域也在随之不断拓展, 充分说明了群智能发展所蕴藏的巨大潜力。

1.2.3.1 遗传算法的发展现状

遗传算法是群智能算法中的一种,在其基础理论研究方面,为了提高算法的性能,出现了很多改进算法。关于遗传算子, Syswerda提出了一致交叉算子^[4],通过随机的交换位序列来保持群体的多样性; Eshelman提出了通过阻止类似的个体之间的杂交来防止过早收敛^[5]的方法。Goldberg提出了混乱遗传算法,通过生成 k 阶短模式,用规定每次至少改变 k 位的方式来避免陷入局部解;另外,为了提高遗传算法的寻优性能还出现了混合遗传算法,将遗传算法与局部搜索相结合,增强了在解空间中的寻优能力。对遗传算法的大量改进方法的出现在大大的推动了其理论的发展的同时,也为其应用领域的扩展提供了理论支持。

20 世纪末以来,遗传算法在两大领域得到了广泛的应用。其一,在工程领域,特别是人工智能和控制领域,不断涌现出超大规模的非线性系统,在这些系统的研究中存在着大量的经典优化方法所不能有效求解的优化问题,诸如神经网络连接权重及网络拓扑结构的优化、模糊系统中模糊规则的选取及隶属函数的确定等等;其二,将遗传算法以独立的或与其它方法相结合的形式用于智能机器学习系统的设计中。

近年来,遗传算法的研究领域出现了新的研究方向:一是基于遗传算法的机器学习,这一新的研究课题把遗传算法从历来在离散搜索空间中的优化搜索算法扩展到具有独特的规则生成功能的机器学习算法^[6];二是遗传算法和神经网络、模糊推理以及混沌理论等智能计算方法相互渗透和结合^[7];三是并行处理的遗传算法研究开始活跃;四是遗传算法和人工生命的研究领域之间正在不断渗透^[8,9]。

1.2.3.2 粒子群算法的发展现状

1995 年, James Kennedy和Russell Eberhar^[10]受他们早期对鸟类群体行为建模和仿真研究结果的启发,提出了著名的微粒群优化(Particle Swarm Optimization, PSO)算法。粒子群算法保留了基于种群的全局搜索策略,采用简单的速度调节模型,利用特有的记忆功能使其动态地跟踪当前搜索状态,具有较强的鲁棒性和快速的寻优速度。PSO作为一种高效的并行搜索算法,非常适合于对实时性要求较高的复杂优化问题进行求解,一经提出,立刻引起了学者们的广泛关注。目前, PSO算法已用于求解带约束的优化问题、动态优化问题、多目标优化问题等大多数的优化问题,此外PSO算法在神经网络训练^[11,12]、电力系统^[13,14]、图像处理^[15,16]等领域也得到了广泛应用。

任何事物都有它的两面性,粒子群优化算法也不例外,在具备自身特有的优

点的同时, 它也存在着容易出现早熟、收敛速度比较慢、容易陷入局部极值点、全局搜索能力不强等缺点。针对这些缺点, 近几年出现了大量粒子群算法的改进方法, 有力的推动了 PSO 算法的发展。文献[17]中 F. van den Bergh 提出了具有局部收敛性能的 GCP SO (Guaranteed Convergence PSO) 算法, 在全局最优微粒的运动方程中引入了随机搜索机制; 文献[18]中 Shi 和 Eberhart 探讨了 PSO 算法中参数的设置对算法性能的影响, 提出了线性递减的惯性权重设置方法; 文献[19, 20, 21]中将模拟生态平衡的仿生技术——小生境技术应用于粒子群算法; 文献[22]中 Shi 和 Eberhart 提出了使用模糊控制系统自适应调整惯性权值和针对动态优化问题的随机惯性权值等方法, 提高了收敛速度, 在单峰值问题上表现出了更高的性能; 文献[23]中 Angeline 用遗传算法中的选择操作来提高收敛到局部极值的速度, 但容易导致算法无法搜索到全局最优解; 文献[24]中 Lovbjerg 等人进一步将进化算法中的交叉操作引入 PSO 算法, 提高了种群的多样性; 文献[25]中 Clerk M 引入收敛因子 K 以保证算法的收敛性, 来达到提高局部收敛速度的目的; 文献[26]中 N.Higashi 和 Kennedy 对粒子速度或者位置引入了一个小概率随机变异操作来提高种群多样性; 文献[27]中 Eberhart 和 Kennedy 提出了二进制 PSO 算法; M. Lovbjerg 和 T.Krink 提出了自组织临界点控制法来提高种群多样性; Higashi 等人提出了变异 PSO 算法[28], 通过引入变异算子来提高算法的全局搜索能力; 在文献[29]中 Kennedy 分析了 PSO 算法中的拓扑结构对算法收敛性的影响; 针对离散的优化问题, 比如组合优化, 文献[30]Yoshida.H 提出了混合编码的粒子群算法。除此之外, 还出现了其它一些 PSO 的改进算法, 所有的改进算法都在提高了 PSO 性能的同时一定程度上增加了算法的复杂性。

1.3 本文所作的工作

本文的研究成果体现在以下几个方面:

1. 对遗传算法和粒子群算法的原理、发展情况及应用领域进行了介绍, 在学习分析支持向量机分类原理的基础上, 构建了沙尘暴预测模型。
2. 分析了支持向量机的参数的取值不同对沙尘暴预测准确率的影响, 设定了参数的寻优范围, 以沙尘暴预测的准确率作为评价函数, 并将遗传算法和粒子群算法分别应用于支持向量机的参数优化, 通过比较训练过程的仿真结果和测试结果, 得出了在此应用领域粒子群算法的性能优于遗传算法的结论。
3. 在研究分析了粒子群算法存在的缺点的基础上, 针对其优化特性会受到惯性权值, 学习因子的影响以及容易陷入局部极值点的特性, 对算法的收敛性进行了分析, 得出了算法收敛与惯性权值、学习因子之间需要满足的约束关系。在

此基础上提出了改进的粒子群算法,采取随着迭代的进行惯性权值以线性方式逐步减小的控制策略,以此来避免算法陷入早熟收敛;另外,对学习因子的选取综合考虑惯性权值的选取方法和算法收敛的条件来进行,保证了算法的收敛性。将改进后的粒子群算法应用于支持向量机的参数优化中,仿真结果表明改进算法的性能优于基本粒子群算法。

本文主要工作结构如图 1-1 所示:

1.4 本章小结

由于人们对生态环境的破坏,近几年沙尘暴天气的发生呈现愈演愈烈的趋势,给人们的生命健康和社会生产带了很大的危害,提高沙尘暴预测的准确率具有重要的现实意义,但是由于沙尘暴的成因复杂,对沙尘暴的预测还停留在较低的水平。鉴于支持向量机分类器的优点,基于支持向量机的沙尘暴预测方法已经崭露头角,但支持向量机的参数仍是制约其预报准确率的重要因素,目前对支持向量机参数的选择仍然是凭借经验、实验对比、大范围的搜寻或者利用交叉验证功能进行寻优,比较常用的是双线性搜索法和网格搜索法等手动调节方法。本文的研究打破了以往基于支持向量机的沙尘暴预测模型中为了提高预报准确率,依靠手动调节支持向量机参数的历史,将群智能算法引入参数寻优过程,提高了预报的准确率。

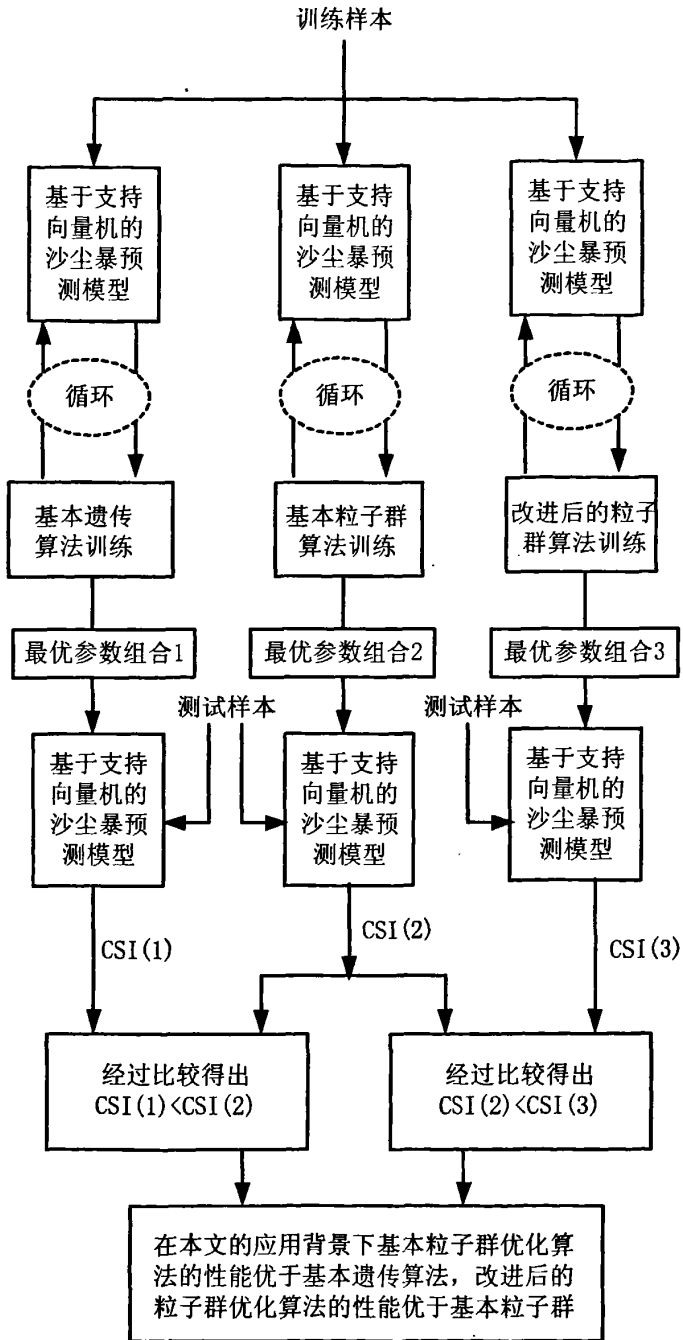


图 1-1 主要工作结构

第二章 遗传算法与粒子群算法

2.1 遗传算法

自然界的生物体按照达尔文“适者生存，优胜劣汰”的进化规律，不断的由低级向高级进化和发展，受此启发，美国密执安大学的Holland教授提出了遗传算法，将这种自然遗传规律采用简单的编码技术和繁殖机制来表现，并用于解决复杂的问题。70年代De Jong基于遗传算法的思想在计算机上进行了大量的纯数值函数优化计算实验。80年代Goldberg在一系列研究工作的基础上，进行归纳总结，形成了遗传算法的基本框架^[31]。

遗传算法是借鉴生物的自然选择和遗传进化机制而开发出的一种全局优化自适应概率搜索算法。它通过对当前群体施加选择、交叉、变异等一系列遗传操作，产生新一代的群体，并逐渐使群体进化到接近或包含最优解的状态。遗传算法具有较强的鲁棒性，特别是对于一些大型、复杂非线性系统，它表现出了比其他传统算法更加独特和优越的性能。

2.1.1 遗传算法的生物学概念

世间的生物由其亲代继承特性或性状，生物遗传物质的主要载体是染色体，而基因则是控制生物性状的遗传物质的功能单元和结构单位。多个基因组成染色体，染色体中基因的位置叫做基因座，而基因座上的基因叫做等位基因。生物个体的性状由基因和基因座共同决定。生物的进化是以集团的形式共同进行的，这个集团被称之为群体(Population)，组成群体的单个生物称之为个体(Individual)，每个个体对其环境的适应能力称为个体的适应度(Fitness)。不同的生物个体间的交配现象类似于遗传算法中的交叉过程；由于多种原因的存在，生物在进化过程中可能会发生基因突变从而产生新的生物基因，这部分变异了的基因也将遗传到下一代，且发生这种变异的概率是可以预测的，但对个体而言变异的发生却是偶然的，这个过程类似于遗传算法中的变异操作过程。基因在环境中表现出的适应能力决定其繁殖能力，适应能力强的个体更容易得以生存，适应能力差的个体将逐渐被环境所淘汰，通过这种自然的选择，物种将逐渐向着适应于环境的方向进化，从而不断产生优良个体。

在标准遗传算法中，通常用二进制编码方式将染色体表示成一维的二进制串

结构，串上各个位置对应的是基因座，各个位置上的 0、1 取值对应等位基因。在遗传算法的运行过程中，一定数量的个体组成群体，群体中个体的数目称为群体规模。遗传算法中通常有一个目标函数与生存环境相对应，由它来确定每个个体的适应度值。遗传算法中需要进行数据转化操作，一个是把具体问题空间中的参数转换成遗传空间中的染色体，即编码过程；另一个是把染色体转换成问题空间中的实际参数值，即译码过程。与生物一代又一代的自然进化过程相类似，遗传算法也是一个反复迭代的过程，由群体不断的经过遗传和进化操作，按照优胜劣汰的规则将适应度较高的个体更多地遗传到下一代，这样最终会得到一个优良个体占多数的群体。

2.1.2 遗传算法的构成要素

遗传算法由染色体编码、个体适应度函数、遗传算子、运行参数的设定四个要素组成。

1 染色体编码 遗传算法采用编码的方式来表示群体中的个体，常用的有实数编码、二进制编码等，以二进制编码为例，其等位基因由二值符号集{0,1}组成。初始个体的基因值也用均匀分布的随机值来生成，染色体的长度根据所面临的实际问题来规定。

2 适应度函数 适应度函数与目标函数相关联，用来表示个体的优良特性，可以理解为个体所产生效益的表示，是个体空间到正实数空间的映射。遗传算法中每个个体遗传到下一代群体中的概率与个体适应度值成正比。为正确计算这个概率，要求所有个体的适应度值必须大于等于零。因此需要确定由目标函数值到个体适应度之间的转化规则，特别是预先确定好当目标函数值为负数时的处理方法。基本遗传算法中一般采用下面两种方法之一将目标函数 $f(X)$ 变换为个体的适应度 $F(X)$ 。

方法一：对于求目标函数最大值的优化问题，变化方法为：

$$F(X) = \begin{cases} f(X) + C_{\min}, & (f(X) + C_{\min} > 0) \\ 0, & (f(X) + C_{\min} \leq 0) \end{cases} \quad (2-1)$$

式(2-1)中， $C_{\min}$ 为一个适当的比较小的数。

方法二：对于求目标函数最小值的优化问题，变化方法为：

$$F(X) = \begin{cases} C_{\max} - f(X), & (f(X) < C_{\max}) \\ 0, & (f(X) \geq C_{\max}) \end{cases} \quad (2-2)$$

式(2-2)中， $C_{\max}$ 为一个适当的比较大的数。

3 遗传算子：遗传算法涉及以下三种遗传操作：

➤ 选择操作(Selection Operation)选择操作是从一个旧种群中选择生命力强的个体产生新种群的过程。适应度值高的个体更有可能在下一代中产生一个或多个子孙。选择操作可以通过随机方法来实现。较典型的几种方法为赌轮盘法、期望值法、排位次法等。其中赌轮盘法的具体执行过程为：

首先计算出所有个体适应度的总和；

其次计算出每个个体适应度的大小，确定每个个体在所有适应度之和中所占比例，由此获得各个个体被遗传到下一代群体中的概率；

最后使用模拟赌轮盘法操作来确定各个个体被选中的次数。

设群体大小为 M ，个体 i 的适应度为 F_i ，则个体 i 被选中的概率 P 为：

$$P = F_i / \sum_{i=1}^M F_i \quad (i=1,2,\dots,M) \quad (2-3)$$

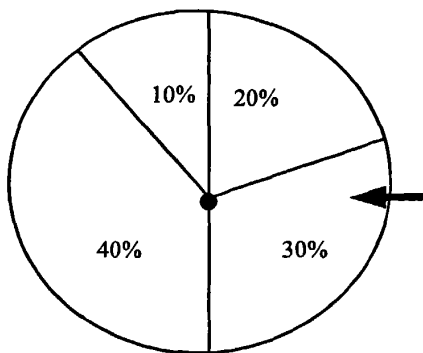


图 2-1 赌轮盘法示意图

➤ 交叉操作(Crossover Operation)交叉操作模拟了生物繁殖现象，通过两个染色体的交换组合，产生新的优良品种。它的操作过程是：

- (1) 任选两个染色体，随机选择一个或多个交换点位置；
- (2) 交换双亲染色体交换点右边的部分，即可得到两个新的染色体编码串。

➤ 交叉体现了自然界中信息交换的思想，它可以是一点交叉、多点交叉、一致交叉、顺序交叉和周期交叉。单点交叉的具体执行过程为：首先对群体中的个体进行两两随机配对，然后对每一对相互配对的个体随机设置交叉点，若染色体的长度为 n ，则有 $n-1$ 个可能的交叉点位置，最后相互交换交叉点之后的部分染色体，从而生成了新的个体。

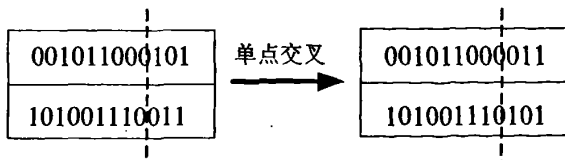


图 2-2 单点交叉示意图

► 变异操作(Mutation Operation)变异操作是用来模拟生物在自然的遗传环境中由于各种偶然因素引起的基因突变, 它以很小的随机概率改变遗传基因(表示染色体的符号串的某一位)的值。在染色体以二进制编码的系统中, 它随机的将染色体的某一基因由 1 变为 0 或由 0 变为 1。变异操作使得搜索可以在初始基因组合以外的空间进行, 避免了寻优过程在早期就陷入局部解而终止。为了在尽可能大的搜索空间中获得质量较高的优化解, 变异操作是必须的, 一般使用基本位变异算子或均匀变异算子实现。具体执行过程是: 对个体的每一个基因座, 用随机概率指定其为变异点, 然后对每一个指定的变异点, 对其基因座取反运算, 从而产生新的个体。

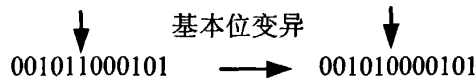


图 2-3 基本位变异示意图

4 基本遗传算法的运行参数:

M : 群体大小, 即群体中所含个体的数量, 一般取为 20~100;

G : 遗传算法的终止进化代数, 一般取为 100~500;

P_c : 交叉概率, 一般取为 0.49~0.99;

P_m : 变异概率, 一般取为 0.0001~0.1。

这四个运行参数对遗传算法的求解结果和求解效率都有一定的影响, 但目前还没有合理地选择它们的理论依据, 在实际应用中, 需要经过反复尝试依靠经验的积累来确定出这些参数的合理取值。

2.1.3 遗传算法的应用步骤和流程

2.1.3.1 GA 的应用步骤

面对一个实际的优化问题, 遗传算法的使用步骤如下:

第一步: 根据要解决的实际问题确定决策变量和约束条件, 确定出个体的表现型 X 和问题的解空间;

第二步: 分析问题, 建立合适的优化模型, 确定出目标函数的类型、数学描述形式和量化方法;

第三步: 选择合适的染色体编码方法, 完成个体由表现型 X 向基因型 x 的转变, 确定出遗传算法的具体搜索空间;

第四步: 对个体进行解码, 找出由个体基因型 x 与表现型 X 之间的转换关系,

并确定出转换方法。

第五步，确定出目标函数到个体适应度函数之间的转换规则；

第六步：设计选择运算、交叉运算和变异运算等遗传算子的具体操作方法；

第七步：确定出遗传算法的有关运行参数。

2.1.3.2 GA 的操作流程

基本遗传算法可以定义为： $GA=(P_0, f, M, P_c, P_c, P_m, T)$ 。其中， P_0 ——初始种群； f ——适应度函数； M ——群体规模； P_c ——选择算子； P_c ——交叉算子； P_m ——变异算子； T ——遗传算法终止条件。算法的具体流程如图 2-4 所示：

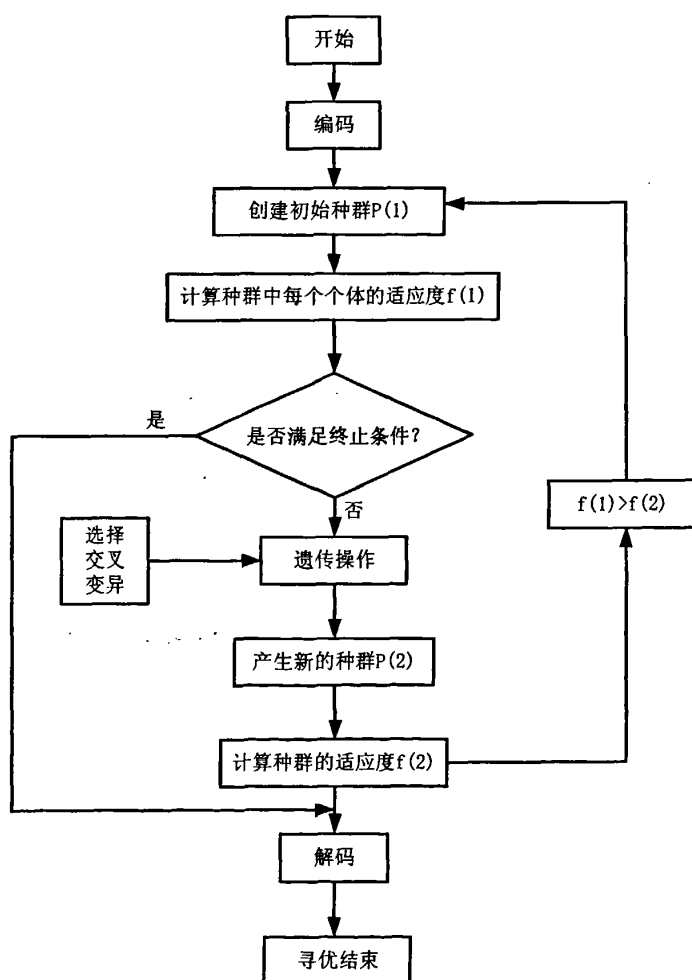


图 2-4 基本遗传算法流程图

2.1.4 遗传算法的特点

遗传算法以决策变量的编码作为运算对象。这种决策变量的编码处理方式,使得我们在优化计算过程中可以借鉴生物学中的染色体和基因等概念,模仿自然界中生物的遗传和进化等机理,也使得我们可以方便的应用遗传操作算子。特别是对一些无数值概念而只有代码概念的优化问题,编码处理使得问题的求解具有良好的可操作性和简单性。

遗传算法利用适应度函数值作为搜索信息,来确定进一步的搜索方向和范围,无需其他辅助信息。正是由于这个特性,使遗传算法避开了求导过程,因而遗传算法适用于任何大规模、高度非线性的不连续多峰函数的优化以及无解析表达式的目标函数的优化问题,具有很好的通用性。

遗传算法同时使用多个搜索点的信息,由多个个体组成的一个初始群体开始最优解的搜索过程,通过对这个群体所进行的选择、交叉、变异等运算,产生新一代的群体。由于群体初始化过程的随机性,尽管每次寻优的过程不尽相同,但实践和理论都已经证明了在一定的条件下遗传算法最终总是能以概率 1 收敛于问题的最优解。

遗传算法采用概率搜索技术,属于一种自适应概率搜索方法,其选择、交叉、变异等运算都是以概率的方式来进行的,从而增强了其搜索过程的灵活性。虽然这种概率特性也会使得群体产生一些适应度不高的个体,但随着进化过程的进行,适应度不高的个体将会逐渐被舍弃,优良个体得以存活下来,使得算法最终收敛于最优解。

遗传算法作为一种随机搜索算法在具备上述优点的同时也存在一些不足。在解决具体问题时,影响算法的几种参数的取值很难确定其最优值及最优组合,往往要根据经验和试验来自行确定,另外,不同的编码方案也会影响算法的执行效率。所有这些,如果处理不当,都会造成遗传算法的搜索效率低下。早熟收敛是遗传算法的另一个重要问题。因为伴随着交叉和选择的作用,搜索到的最优个体就会以指数速度繁殖,该个体将很快占据种群的大部分,通过染色体的交叉和变异,种群很难产生优于亲本的子代个体;另外,在均匀种群中经常发生这种情况,所有标准形式的交叉简单地重复产生当前的亲本,使得进一步的优化完全依赖于位变异,而且可能非常缓慢,也会产生早熟收敛。

2.2 粒子群算法

粒子群优化算法是一种新的仿生学进化计算方法,最早是由Kennedy与

Eberhart于1995年提出的^[32,33,34]。PSO算法的运行机理不是依据个体的自然进化规律,而是源于对鸟群捕食行为的研究。同遗传算法类似,PSO算法也是一种基于种群思想的迭代优化算法。

在生物群体中存在着个体与个体、个体与群体间的相互作用、相互影响的行为,这种行为体现的是一种存在于生物群体中的信息共享机制。PSO算法就是对这种社会行为的模拟,即利用信息共享机制,使得个体间可以相互借鉴经验,从而促进整个群体的发展。在PSO算法中,首先系统初始化为一组随机解,在迭代中搜寻最优值,在解空间中追随最优的粒子,该算法将群体中的每个个体视为多维搜索空间中一个没有质量和体积的粒子,这些粒子都具有很好的感知力,能够感知自己周围的局部最好位置的个体和整个群体中全局最好位置的个体的存在,他们在搜索空间中以一定的速度飞行,并根据粒子本身的飞行经验以及同伴的飞行经验对自己的飞行速度进行动态调整,即每个粒子通过迭代过程中自身的最优值和群体的最优值来不断的修正自己的前进速度和方向,从而形成群体寻优的正反馈机制。

PSO算法与其他进化计算方法相比,具有以下主要特点:

- 1) 每一个粒子都被赋予了一个随机速度并在问题空间中流动;
- 2) 个体具有记忆能力;
- 3) 个体的进化主要是通过个体之间的合作与竞争来实现;
- 4) PSO在多维空间函数寻优、动态目标寻优等方面有着收敛速度快、解质量高、鲁棒性好等优点,它在搜索的初期收敛速度很快,而在后期容易陷入局部最小^[35,36]。

5) PSO算法编码方式简单、容易理解、易于实现。

正是由于上述特点,PSO算法发展很快。粒子群算法最初是用以处理连续优化问题的,目前已被广泛应用于函数优化、组合优化、神经网络训练、模糊系统控制以及其他应用领域。

2.2.1 PSO算法的描述^[37,38,39,40]

粒子群算法的优化问题可以看作在空中鸟群觅食问题,空中飞行的每一只鸟就相当于粒子群算法解空间中的一个“粒子”(Particle),也是优化问题的一个解。食物的位置就相当于优化问题的最优解。PSO算法中,每个粒子具有位置和速度两个特征,粒子位置坐标所对应的目标函数值可作为粒子的适应度值,该算法通过适应度值的大小来衡量粒子的优劣程度。

假设一个包含 m 个粒子的粒子群,粒子在 D 维搜索空间中“飞翔”,粒子群

中第 i 个粒子的参数表示如下：

$$\begin{aligned}x_i &= (x_{i,1}, x_{i,2}, \dots, x_{i,D}) \\v_i &= (v_{i,1}, v_{i,2}, \dots, v_{i,D}) \\P_i &= (P_{i,1}, P_{i,2}, \dots, P_{i,D}) \\P_g &= (P_{g,1}, P_{g,2}, \dots, P_{g,D})\end{aligned}$$

x_i 、 v_i 、 P_i 、 P_g 分别表示第 i 个粒子在 D 维空间中的当前位置、当前速度、历史最优位置及整个粒子群中的所有粒子在 D 维空间中的历史最优位置。

基本粒子群算法的迭代公式为：

$$v_{i,d}(t+1) = \omega v_{i,d}(t) + c_1 \text{rand}(1)(P_{i,d} - x_{i,d}(t)) + c_2 \text{rand}(2)(P_{g,d} - x_{i,d}(t)) \quad (2-4)$$

$$x_{i,d}(t+1) = x_{i,d}(t) + v_{i,d}(t+1) \quad (2-5)$$

$$i=1,2,\dots,m$$

式(2-4)中，惯性权重(inertia weight) ω 用来控制前一次迭代产生的速度对本次迭代速度的影响； c_1, c_2 为学习因子，通常取 $c_1 = c_2 = 2$ ； $\text{rand}(1)$ 、 $\text{rand}(2)$ 均为分布在(0,1)之间的随机数。式(2-4)的第一部分为粒子先前的速度乘一个权值进行加速，表示粒子对当前自身运动状态的信任，依据自身的速度做惯性运动；第二部分为“认知”部分，表示粒子对自身的思考；第三部分为“社会”部分，表示粒子与所有粒子信息的共享与合作。式(2-5)体现了粒子群算法根据当前的速度情况将群体中的个体移动到好的区域的过程。

2.2.2 PSO 算法的应用步骤

1. 随机初始化粒子种群，也就是随机初始化种群中每一个粒子的位置和速度；
 2. 根据优化问题的目标函数定义出适应度函数，并计算出所有粒子的适应度值；
 3. 根据计算出的适应度函数值更新种群中每一个粒子的最优历史位置；
 4. 根据计算出的适应度函数值更新种群中所有粒子中的最优历史位置；
 5. 根据迭代公式(2-4)、(2-5)对种群中的所有粒子进行速度和位置的迭代更新；
 6. 重复 1-5 的步骤直到满足算法的迭代条件终止。
- 流程图的形式如图 2-5 所示。

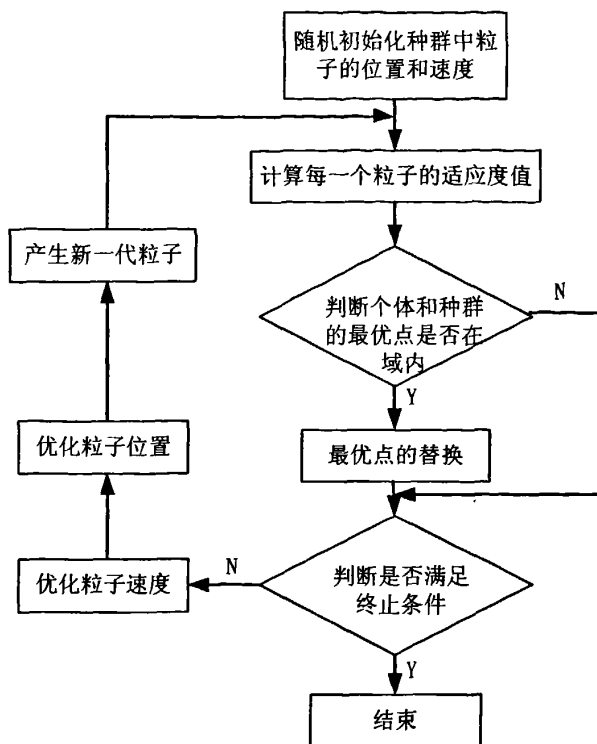


图 2-5 基本粒子群优化算法流程图

2.3 本章小结

PSO 和 GA 都是基于群体迭代的启发式随机优化算法，它们都是对自然中随机系统的仿真，都具有并行性，但它们有很多不同之处。首先它们所模拟的自然随机系统不一样，GA 所模拟的是生物系统的进化过程，其最基本的单位是基因，遗传信息在生物体的每一代之间传播；PSO 是源于对动物群体之间通过协作解决问题行为的模拟，它主要强调对社会系统中个体之间相互协同作用的模拟。其次，GA 依据的是“适者生存，优胜劣汰”的达尔文进化论，不好的个体在竞争中会被淘汰，最好的个体通过产生更多得的后代来传播自己的优良基因；PSO 强调的是“个体协作”，表现不好的个体通过学习信息来改进自己，从而向好的方向发展，其中优秀的个体吸引其他个体向它靠近来传播自己的信息。再次，GA 的迭代是由选择、交叉、变异等操作算子组成，而 PSO 迭代中隐含的操作是对最优点的跟随。

本章分别详细叙述了遗传算法和粒子群算法的原理、应用步骤和算法流程。表明了遗传算法的关键步骤是在种群的初始化、遗传进化过程中选择算子、交叉

算子和变异算子的设计以及适应度函数的建立上；鉴于粒子群算法容易陷入局部极值点的缺点，其关键在于群中个体位置及速度的初始化、惯性因子及学习因子等参数的设定等方面。

对群智能算法的研究，目前国内外研究者的着眼点大多数集中在这些关键问题的修改上，试图寻找求解质量和速度的理想的平衡点，然后将这些改进应用到具体问题中来说明算法的有效性。

第三章 基于支持向量机的沙尘暴预测模型

目前对沙尘暴的预测，主要方法有K-最近邻法、人工神经网络法等，其中BP神经网络预测模型被广泛应用，但BP神经网络稳定性差，容易产生过学习现象，预测效果并不理想。而九十年代中期发展起来的支持向量机，是以统计学习理论(SLT)为基础的新的机器学习技术。它基于结构风险最小化（SRM）理论基础之上，结构简单、技术性能和泛化能力强，能很好的解决小样本、非线性和局部极小点等问题，具有和数据维数无关的特性，避免了“维数灾难”，特别在处理高维数据的方面具有好的性能。鉴于支持向量机在分类方面的优点，本文用支持向量机构建了沙尘暴的预测模型。

3.1 支持向量机理论

3.1.1 支持向量机原理^[41]

支持向量机（SVM）是基于统计学习理论提出的一种新型的分类技术，它根据有限的样本信息在模型的复杂性和学习能力之间寻求最佳折衷，以期获得最好的推广能力。由于SVM是建立在结构风险最小化基础之上的，不存在过学习问题，因此具有更好的泛化能力，另外，SVM的求解是凸二次优化问题，解决了神经网络方法中不可避免的局部极值点问题。

支持向量机分类通常需要训练集和测试集，分类的目标就是根据对训练集分类的情况创建一个模型，该模型能正确识别测试集中输入向量所归属的标签类别。

对于支持向量机的研究是从最初的二分类问题展开的，多分类模型由多个二分类支持向量机来构造。设分类样本为 n 维向量，其中 k 个样本为训练样本集，即 (x_i, y_i) ， $i=1,2,\dots,k$ ，输入向量 $x_i \in R^n$ ， $y_i \in \{-1,1\}$ 是类别标签。

3.1.1.1 线性可分问题

最早的SVM多类分析多数采用one-against-one(一对一)思想,是根据模式识别中的两类线性可分问题提出来的,基本思想如图 3-1 所示。其中,实心点和空心点代表两类样本, H 为最优分类超平面, H_1 , H_2 分别为经过各类中离最优分类超平面最近的样本平行于最优分类超平面的平面, H_1 与 H_2 之间的距离叫做分类间隔(margin)。最优分类超平面不但能将训练向量集正确地分开,而且使得分类间隔最大。距离最优分类超平面最近的样本向量称为支持向量^[42]。

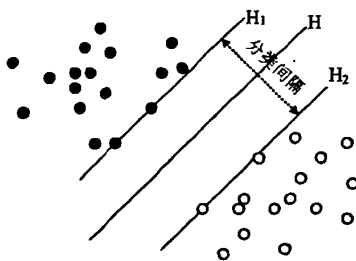


图 3-1 支持向量机分类原理示意图

n 维空间中线性判别函数的一般形式为:

$$g(x) = \omega \cdot x + b \quad (3-1)$$

对于线性可分的二分类问题一定存在一个超平面将两类样本正确的分开。假设超平面的方程为

$$\omega \cdot x + b = 0 \quad (3-2)$$

将判别函数归一化, 等比例调节 ω 和 b , 使得所有的样本点满足:

$$y_i(\omega \cdot x_i + b) \geq 1, \quad i=1, 2, \dots, k \quad (3-3)$$

根据分类间隔的定义, 得两类训练样本的分类间隔为:

$$\text{margin} = \min_{x_i | y_i=1} \frac{\omega \cdot x_i + b}{\|\omega\|} - \max_{x_i | y_i=-1} \frac{\omega \cdot x_i + b}{\|\omega\|} = \frac{2}{\|\omega\|} \quad (3-4)$$

线性SVM的目标是寻找一最优分类超平面 H , 使分类间隔具有最大值。在图 3-1 中, 分类平面 H_1 、 H_2 上的样本点满足 $|g(x)|=1$, 它们离最优超平面 H 的距离最小, 所以构造最优超平面的问题转化为:

$$\begin{cases} \min \phi(\omega) = \frac{1}{2} \|\omega\|^2 = \frac{1}{2} (\omega \cdot \omega) \\ \text{s.t. } y_i [(\omega \cdot x_i) + b] \geq 1, \quad i = 1, 2, \dots, k \end{cases} \quad (3-5)$$

式(3-5)为一个正定的二次规划问题, 通过引入Lagrange系数, 根据约束优化理论中的Kuhn Tucker(KT)条件^[43], (3-5)式中的极值问题转化为它的对偶问题, 如式(3-6)所示,

$$\begin{cases} \max \phi(\alpha) = \sum_{i=1}^k \alpha_i - \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^k \alpha_i \alpha_j y_i y_j x_i^T x_j \\ \text{s.t. } \sum_{i=1}^k \alpha_i y_i = 0, \text{ 且 } \alpha_i \geq 0, i = 1, 2, \dots, k \end{cases} \quad (3-6)$$

根据 KT 条件可知, 对于大多数的样本 $\alpha_i = 0$, 那些 $\alpha_i \neq 0$ 对应的样本即为支持向量, 这样得到最终的分类判别函数为:

$$f(x) = \text{Sgn} \left(\sum_{i=1}^m y_i \alpha_i (x_i \cdot x) + b_0 \right) \quad (3-7)$$

其中 m 为支持向量的个数。

由此可见, 只有支持向量在分类器的构建上才是有意义的, 仅由支持向量就能寻得分类面。由于训练集中大部分样本都不是支持向量, 因此约减训练集, 删除最不可能成为支持向量的训练样本, 将大样本的二次规划问题转化为小样本的二次规划问题具有一定的研究意义。

3.1.1.2 线性不可分问题

实际情况中, 考虑到噪声或其他干扰因素造成的影响, 少数线性可分的样本也会变的线性不可分, 这种情况下就会有部分样本被错误地分类。为提高了样本分类的准确率, 引入松弛变量 $\varepsilon_i \geq 0$, (3-5)式中的约束条件变为

$$y_i [(\omega \cdot x_i) + b] + \varepsilon_i \geq 1, i = 1, 2, \dots, k \quad (3-8)$$

这样, 式(3-5)就转化为

$$\begin{cases} \min \theta(\omega, \varepsilon) = \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^k \varepsilon_i \\ y_i [(\omega \cdot x_i) + b] + \varepsilon_i \geq 1, \quad i = 1, 2, \dots, k, \text{ 且 } \varepsilon_i \geq 0 \end{cases} \quad (3-9)$$

其中 C 是对线性不可分的极少数样本点施加的惩罚系数, C 越大表示对错误分类的惩罚越大, 当 $C \rightarrow \infty$ 时, 采用拉格朗日乘子法将(3-9)式转化为与(3-6)式一样

的对偶问题来求解，相应的约束条件变为

$$0 \leq \alpha_i \leq C, i = 1, 2, \dots, k.$$

α_i 为支持值，满足约束条件时所对应的 x_i 称为支持向量。

3.1.1.3 核函数

由于现实世界的复杂性，多数的分类情况需要比线性函数更富有表达能力的假设空间，早在 20 世纪 60 年代 Minsky 和 Papert 就已明确指出了线性学习机的学习能力有限。

针对样本在原特征空间的不可分问题，Vapnik 引入了核函数，将分类问题转化到新的特征空间进行，如果选用适当的映射函数能在 Z 空间中寻找一个最优的分界面(分类器)，那么就将样本的不可分问题转化为线性可分问题。

以二维空间中的样本为例，设输入空间为 $x=(x_1, x_2, \dots, x_n)$ ，且样本在输入空间中线性不可分，通过函数 ϕ 将输入映射到新的特征空间 $\phi(x)=(\phi_1(x), \phi_2(x), \dots, \phi_n(x))$ ，这样将图 3-2a 所示的线性不可分问题转化为图 3-2b 所示的线性可分问题。

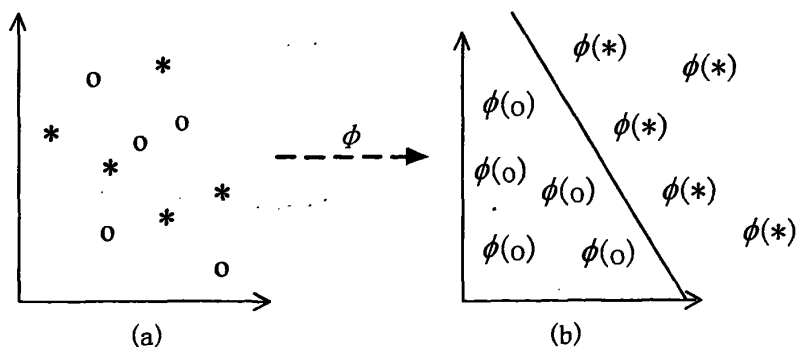


图 3-2 二维空间的特征映射

映射函数 ϕ 可以将输入向量从低维特征空间映射到高维特征空间 Z ，但在映射的过程中空间维数会急剧增长，导致多数情况下很难在新的特征空间 Z 中直接计算出最优分类超平面。

核函数 $K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle$ 的引入，把本来需要在高维空间中的计算转化到低维空间中进行，巧妙地解决了维数灾难问题。根据泛函的有关理论，只要

满足 Mercer 条件的函数就可以作为核函数，常用的核函数有三种：

① Sigmoid 核函数： $K(x, x_i) = \tanh(v(x \cdot x_i) + c)$ ；

② RBF 核函数： $K(x, x_i) = \exp(-\frac{|x - x_i|^2}{2\sigma^2})$ ；

③ Polynomial 核函数： $K(x, x_i) = [\gamma(x \cdot x_i) + 1]^q$

对于线性可分问题引入核函数后，寻优目标函数式(3-6)变为

$$\phi(\alpha) = \sum_{i=1}^k \alpha_i - \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^k \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (3-10)$$

相应的分类函数式(3-7)变为

$$f(x) = \text{sgn}[\sum_{i=1}^m \alpha_i y_i K(x_i, x) + b_0] \quad (3-11)$$

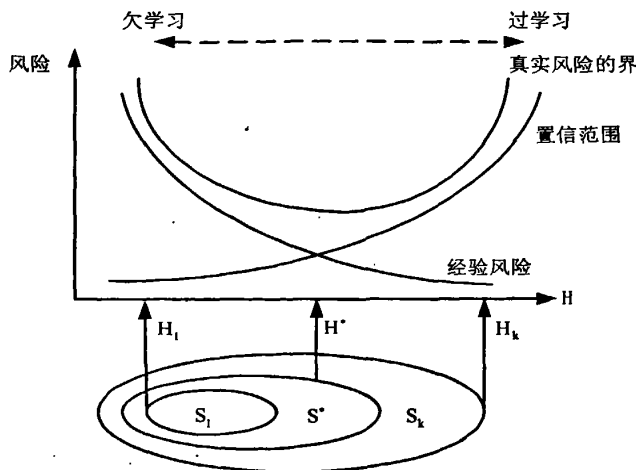
根据VC维理论，Vapnik^[44]给出了支持向量机分类器的误差上界：

$$\varepsilon = \frac{1}{k} R^2 / \gamma^2 \quad (3-12)$$

其中， R 为在特征空间中包含所有训练样本的最小球半径； γ 为超平面之间的间隔。

3.1.2 SVM 核函数的选择及其参数对分类的影响

支持向量机的核心是结构风险最小化原则，它把函数集构造成一个函数子序列，各个子集按照 VC 维的大小排列，在子集中寻找经验风险和置信范围的最佳折衷，如图 3-3 所示。



函数集子集: $S_1 \leq S^* \leq S_k$ VC 维: $H_1 \leq H^* \leq H_k$

图 3-3 结构风险最小化

结构风险最小化一方面要求函数子集具有确定的 VC 维, 另一方面要求在给定经验风险情况下, 置信范围最小。对于 SVM 算法就要试图找到既要使经验风险最小, 又要使置信范围达到最小的函数。不同子空间会得到不同的最优参数组合, 这就是 SVM 算法中对其核参数和误差惩罚参数同时进行优化的意义所在。

我们除了在确定子空间优化 C 以获得最优SVM外,还要优化核参数以获得全局最优的SVM。根据统计学习理论,每一种参数对的取值组合通过比较它们的风险上界就可以找到适合该数据集的最优SVM^[45],图 3-3 中 H^* 处,既不会出现“欠学习”现象,也不会出现“过学习”现象,SVM的经验风险和置信范围接近最佳的组合,此时的SVM具有最好的推广能力。

3.1.2.1 核函数的选择

核函数是 SVM 中能够调节的参数之一。核函数的形式及其参数的取值决定了支持向量机分类器的类型和复杂程度，核参数、映射函数以及特征空间是一一对应的，核函数也决定了输入空间与高维空间的非线性映射的本质，控制了映射过程的复杂度。分类子空间的维数过高或过低情况下得到的 SVM 都不会有较好的分类效果，只有首先选择了合适的核函数将数据映射到合适的特征空间才可能得到泛化能力好的 SVM，选取合适的核参数是取得满意分类效果的关键问题之

一,但目前核函数选择的理论依据很少。

RBF 核函数只含一个参数 σ ,易于参数优化, σ 为 RBF 函数的宽度参数,控制了核函数的径向作用范围,如果选择合适的 σ ,RBF-SVM 在实际分类问题中就会表现出良好的学习性能。

Polynomial 核函数和 Sigmoid 核函数有两个参数,会使参数优化更复杂,实验证明得到的 SVM 的性能却并不比 RBF 核函数得到的 SVM 好。

Polynomial 核函数,当次数很大时,它的内积值可能趋于无穷或者趋于 0,而 RBF 核函数的取值范围为(0, 1),所以 RBF 核函数实现起来比较容易。

鉴于以上特点,本文选择了 RBF 核函数。

3.1.2.2 参数对分类的影响

在 SVM 的学习中,虽然支持向量机的参数选择会对其分类效果产生很大的影响,但对 SVM 核函数参数的选择目前仍没有形成一个统一的模式。

RBF 核 SVM 性能的优劣直接受核参数 σ 的影响,当 σ 很小时,训练误差很小,测试误差接近于 1,此时几乎所有的训练样本都是支持向量,SVM 的推广能力极差;随着 σ 的逐渐增大,支持向量数减少,训练误差增大,SVM 的推广能力在增强;当 σ 达到一定值时,支持向量数又开始增加,对应的测试误差和训练误差也同时开始增加,SVM 对训练样本的分类能力和推广能力都变差。在 σ 由小变大的过程中,RBF-SVM 的推广能力经历了由低到高再到低的过程。

惩罚参数 C 的作用是实现在错分样本的比例和算法的复杂度之间的折衷,即在确定的数据子空间中调节支持向量机经验风险和置信范围之间的比例,直到学习机器达到较好的推广能力,子空间不同对应 C 的最优取值也不同。在数据子空间确定的情况下, C 的取值越小表示对经验误差的惩罚力度越小,SVM 的复杂度小而经验风险值较大;反之亦然。每个数据子空间至少存在一个合适的 C 使得 SVM 的推广能力最好。当 C 超过一定值时,SVM 的复杂度就会达到数据子空间所允许的最大值,导致经验风险和推广能力变化迟缓。

误差惩罚参数 C 和 RBF 核参数 σ 共同影响着支持向量机对测试样本的分类效果和推广能力, σ 决定了样本特征子空间分布的复杂程度,而误差惩罚参数 C 通过对错分率的控制来影响 SVM 的推广能力。

3.2 实验样本及数据预处理

3.2.1 实验样本

本文所采用的数据集是美国环境气象中心(NCEP)提供的,每天一个样本,每个样本的结构是相同的。鉴于我国沙尘暴发生的时间和地区特点,将NCEP资料圈定在我国西北部从初春到初夏的范围之内(东经 $70^{\circ}\sim 115^{\circ}$,跨度 45° ;北纬 $35^{\circ}\sim 55^{\circ}$,跨度 20°)。所圈定的地域范围内生成格点场。按照NCEP资料每 2.5° 记一格的数据格式,东西向跨 45° 分为18格、南北向跨 20° 分为8格。东西向和南北向的交叉记为一个格点,这样,一个格点场的数据量就是 $19\times 9=171$ 个,即每个格点场提供 19×9 的数据阵。考虑到沙尘暴形成的基本条件,选取4个物理场即500hpa的高度(Hgt)场、700hpa的两个风(UV)场和850hpa的位温(θ_{se})场。这样,一个样本的总维数为 $171\times 4=684$ 。选取1981年至1997年从初春到初夏共采集到2027个样本^[46]。表3-1所示为本文所用的所有样本。

表 3-1 本文所用的沙尘暴样本

样本采集时间	总个数	非沙尘暴日	沙尘暴日	沙尘暴日频数/%	单样本维数
1981-1997(初春~初夏)	2027	1454	573	28.27	56

3.2.2 样本数据预处理

由于高维数据量太大,很难用来预报,需要对数据集进行削减。几何上,主成份分析法是一种数据驱动方法,可以看作是坐标轴的旋转,将原始坐标系的坐标轴旋转成一组新的正交坐标轴,并按照它们占原始数据方差的多少排列这些坐标轴,数据削减的原则是在降维的过程中使得关键特征能保留下来,即使得维数特征占原始数据方差的多数。该方法事前没有假定数据内部存在不同的类,因此是一种非监督的特征提取方法。

假设要压缩的数据由 N 个 d 维元组或数据向量组成,主成份分析法搜索 c 个 d 维正交向量,它们能够在保证不损失大量有用信息的基础上最佳的表示数据,其中 $c\leq N$ 。这样原始数据就被投影到一个小得多的数据空间。本文所用数据是将原684维样本降维处理后得到的56维样本。另外,由于源样本数据的个数繁多,若将所有的样本都用于支持向量机的训练,则会大大的降低训练的速度。

本文采用随机抽样的方法，在 2027 个样本中随机抽取部分作为训练样本，剩余样本中随机抽取 1000 个样本用作测试样本，以评价训练结果的优劣程度。样本的示意图如表 3-2：

表 3-2 沙尘暴样本特征

样本	特征(56 维)				标签(0、1)
A_1	A_{11}	A_{12}	...	$A_{1(56)}$	1
A_2	A_{21}	A_{22}	...	$A_{2(56)}$	0
A_3	A_{31}	A_{32}	...	$A_{3(56)}$	0
$\vdots$	$\vdots$	$\vdots$	...	$\vdots$	$\vdots$
A_{2027}	$A_{2027(1)}$	$A_{2027(3)}$	...	$A_{2027(56)}$	1

注：0——没有沙尘暴发生 1——有沙尘暴发生

3.3 SVM 训练过程示意图

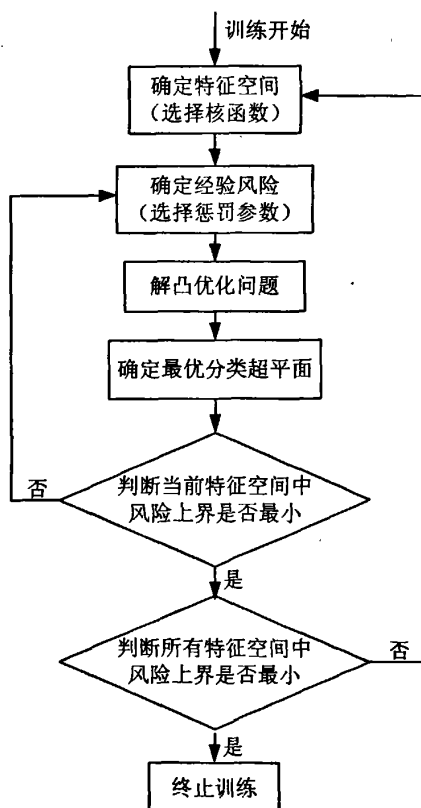


图 3-4 SVM 训练过程

3.4 本章小结

本章内容首先从线性可分和线性不可分两种情况出发介绍了支持向量机分类的基本原理,然后结合结构风险最小化理论,分析了核函数的选择依据及核参数和惩罚因子分别对支持向量机分类特性造成的影响,构造了基于 RBF-SVM 的沙尘暴预测模型。接下来介绍了沙尘暴源样本数据集及通过对源样本数据集进行处理得到的样本集情况,最后以图示的方式展示了支持向量机的训练过程。

第四章 遗传算法、粒子群算法在 SVM 参数优化中的应用

本文选用的沙尘暴预测模型是 RBF-SVM, 其中惩罚参数 C 和核参数 σ 的取值都会对系统的性能造成一定的影响, 目前参数确定的方法还处于手动、凭经验确定阶段, 故在参数的调节过程中存在一定的盲目性, 而且当在较大的寻优范围内调节时需要调整的次数较多, 调节过程复杂, 效率低。因此运用群智能优化算法对 SVM 的参数进行自动优化具有重要的研究意义。

4.1 遗传算法在 SVM 参数优化中的应用

遗传算法作为一种基于生物遗传和进化的群智能算法, 具有操作简单、速度快等优点, 不需要复杂的规则, 只通过对编码串进行简单的复制、交叉、变异操作便可达到寻优的目的; 另外, 它的搜索过程是从多点开始, 在解空间进行高效启发式搜索, 克服了从单点出发进行搜索的盲目性, 避免了过早的陷入局部最优解。鉴于以上优点, 本文将遗传算法用于支持向量机的参数优化中, 试图寻找一组最佳的 SVM 参数对组合, 以提高其分类性能, 从而提高对沙尘暴预测的准确率。

4.1.1 参数的确定及其编码和解码

由于本文选用的沙尘暴预测模型是 RBF-SVM, 所以误差惩罚参数 C 和 RBF 核参数 σ 为被优化的参数, 为此需要指定参数的寻优范围及编码。

4.1.1.1 参数寻优范围

参数的寻优范围如果设置的过大, 则会降低寻优的准确率, 如果设置的范围过小则容易错过最佳参数组合, 造成最优解的遗漏, 本文参数的寻优范围设定为: $C[50, 100]$, $\sigma[0, 150]$ 。

4.1.1.2 染色体编码

本文采用二进制编码方式对染色体进行编码。编码的长度决定着寻优的精

度,但编码过长则会降低搜索效率。本文结合实际问题,在精度和效率之间寻求折衷,采用了十位编码来表示每一个参数,两个参数表示成一个二十位的二进制编码串,前十位是对惩罚参数 C 的编码,后十位是对核参数 σ 的编码。10 位二进制编码串可以表示从 0~1023 之间的 1024 个不同的数,故将 C 和 σ 的定义域离散化为 1023 个均等的区域。对参数 C 来说,从点 0 到点 50,依次让它们分别对应于 0000000000~1111111111 之间的二进制编码串。同理对参数 σ 进行编码。将两个 10 位长的二进制编码连接在一起,就构成了优化问题的染色体编码。这样,解空间和搜索空间就具有了一一对应的关系。

4.1.1.3 解码

编码方式决定解码方式,解码时需要将 20 位长的二进制编码串切断为两个 10 位长的二进制编码串,然后分别将它们转换成对应的十进制整数代码,记为 y_1 、 y_2 。本文中对应的解码公式为:

$$x_1 = (50 - 0) \times \frac{y_1}{1023} + 0 \quad (4-1)$$

$$x_2 = (150 - 50) \times \frac{y_2}{1023} + 50 \quad (4-2)$$

4.1.2 种群初始化

种群的规模和产生方式是种群初始化中的两个重要问题。种群的大小应结合具体问题中计算的复杂度来决定。由于本文中支持向量机参数对的优劣反映了对大量的沙尘暴样本数据进行分类后的结果,计算量大,故不适合建立大规模的初始种群。本文的种群数目定为 20 个。

编码方式决定了本文中的种群是由一组 20 位的二进制编码串组成的,因此初始种群的产生方式为:随机产生 0~1 之间均匀分布的随机数,其中 0~0.5 之间的随机数代表 0,0.5~1 之间的随机数代表 1。这样,初始种群就是一个 20 行 20 列的 0、1 矩阵。

4.1.3 适应度函数的确定

在气象上,沙尘暴预报的准确率由成功界限指数 CSI 来衡量预报模型的性能。CSI 的定义如下:

$$CSI = \frac{c_f}{c_f + w_f} \quad (4-3)$$

c_f 为正确报出的沙尘暴日数, w_f 为漏报与空报数之和。

由成功界限指数的定义可知, CSI 的值为非负的实数, CSI 值越大代表预报的准确率越高, 代表支持向量机分类的效果越好。可见支持向量机的参数对组合越理想, CSI 的值就越大。本文将 CSI 设定为适应度函数。

4.1.4 遗传算子及运行参数的设定

本文中遗传运算采用比例选择算子, 交叉运算使用单点交叉, 变异运算使用基本位变异算子。终止进化代数 $G=50$, 交叉概率 $P_c=0.60$, 变异概率 $P_m=0.01$ 。

► 在选择操作中, 已知种群的规模为 20 个, 适应度函数表示为 CSI, 选择的次数为种群中粒子的个数, 每个粒子被选中的概率为 $P = CSI_i / \sum_{i=1}^{20} CSI_i$, 这样适应度值大的个体就会被重复选择, 而且重复出现的次数会随着适应度值的增大而增大, 而那些适应度值小的个体就会在选择操作中被淘汰, 可见, 选择操作实现了那些性能优良的个体存活下来的目的。

► 在交叉操作中, 交叉概率控制着样本发生交叉的概率, 具体实现方式为: 每一次进化过程中, 在交叉操作运行之初, 都会产生一个 0 到 1 之间的随机数 $rand(1)$, 若 $rand(1) < P_c$, 则交叉发生。

对于交叉点位置的确定也是通过随机数来控制的, 具体的实现方式为: 产生 0 到 1 之间的一个随机数 $rand(2)$, 然后由 $n=20*rand(2)$ 决定交叉点的位置。

► 变异操作也是通过随机数来控制的, 具体的实现过程与交叉操作相似, 不再赘述。

可见, 选择操作是对性能优良的个体的筛选, 交叉和变异是使待优化的参数在规定的寻优范围内发生改变的“推动力”。

4.2 粒子群算法在支持向量机参数优化中的应用

虽然遗传算法有很多优点, 求解优化问题一直是该算法的经典应用领域, 但是它也有着自身难以克服的缺点, 首先, 对于不同的优化问题, 遗传算法往往要重新设计“交叉”、“变异”算子, 甚至要开发新的进化操作; 另外遗传算法操作

复杂、大多数情况下效率比较低。所以寻求新的解决优化问题的算法一直是研究的热点。粒子群优化算法由于其容易理解、易于实现，在许多优化问题中得到成功应用,并且很多情况下要比遗传算法更有效率，所以 PSO 算法在优化问题中的应用是一个很有意义的研究方向。

为了增强与遗传算法性能之间的可比性，与 GA 的应用一样，PSO 算法的初始种群数目也规定为 20 个，终止循环代数为 50。

4.2.1 种群初始化

种群的初始化包括位置和速度的初始化两部分，结合本文的实际优化问题，由粒子群算法的原理和迭代公式(2-4)、(2-5)可知，种群中每个粒子的寻优过程是在两维空间中的“运动”，初始化中惩罚参数 C 和核参数 σ 的具体取值与种群中每个粒子的位置相对应，惩罚参数 C 和 σ 采用实数编码方式被初始化为一个 2 行 20 列的矩阵 A ，第一行代表 C 的 20 种不同取值，第二行代表 σ 的 20 种不同取值，这样矩阵的每一列就是 C 和 σ 的一对组合。矩阵中的每一个数是按下列公式得到的，即

$$A(1, j) = 0 + (50 - 0) * rand(0,1), \quad j = 1, 2, \dots, 20 \quad (4-4)$$

$$A(2, j) = 50 + (100 - 50) * rand(0,1), \quad j = 1, 2, \dots, 20 \quad (4-5)$$

速度初始化方法很简单，令每个粒子的初始速度都为 0 到 1 之间的随机数。这样每个粒子就被给予了一定的初始运动状态。

4.2.2 运行参数的设定

PSO 算法中，惯性权值 ω 决定着粒子前一次迭代时的速度对本次迭代的影响（参见公式(2-4)），较大的 ω 取值会扩大粒子的搜索空间，有利于使其跳出局部极小点，较小的 ω 会增强粒子的局部搜索能力，有利于算法收敛。大量的实验研究表明：面对不同的具体问题，对 ω 的取值有着不同的要求，但大致来说， ω 的取值在 (0.6, 1.0) 之间时算法会有比较好的性能。学习因子 c_1, c_2 分别表示粒子“自学习”和“社会学习”的能力。本文优化中，取 $\omega=0.8$, $c_1=c_2=2$ 。

4.3 仿真结果

4.3.1 训练过程

为了保证训练过程的可靠性和稳定性,本文将沙尘暴样本采用随机抽样的方式划分为大小不同的几部分分别作为训练集。本节内容将针对不同的训练集分别采用两种算法进行训练。

由于两种算法都是全局寻优的概率搜索算法,其种群初始化结果均决定着每次的寻优过程,尽管同一种算法针对同一组训练集,由于初始种群的不同,每次训练的过程也不同,但最终都能稳定到一个最优的平衡点。

本节给出的仿真图都是具有代表意义的,是多次仿真中出现频率最高的情况,这种带有大众化色彩的典型图像之间的对比是有一定现实意义的。

情况一: 400 个样本作为训练集。

遗传算法和粒子群算法的训练结果分别如图 4-1、4-2 所示。

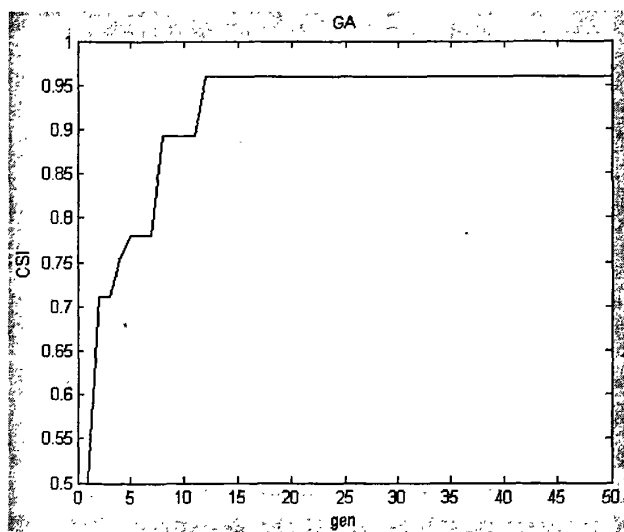


图 4-1 遗传算法的最优解随进化代数的变化情况

$(C, \sigma) = (110.9376, 2.0274)$, $CSI = 0.9587$;

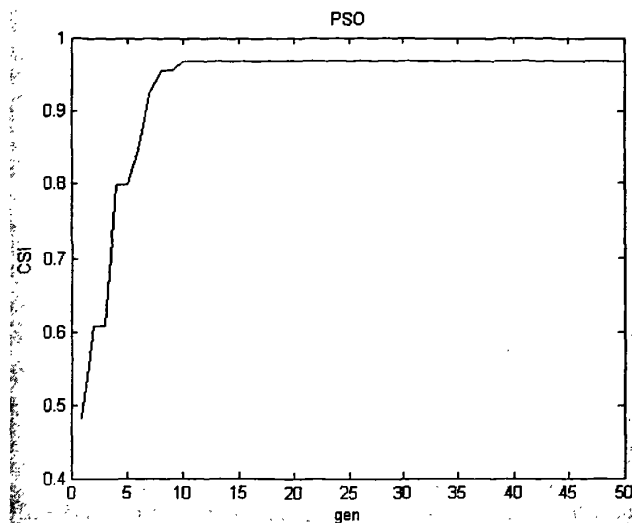


图 4-2 粒子群算法的最优解随进化代数的变化情况
(C, σ)=(119.9078,1.7708), CSI=0.9678

遗传算法和粒子群算法的训练结果对比图如图 4-3 所示:

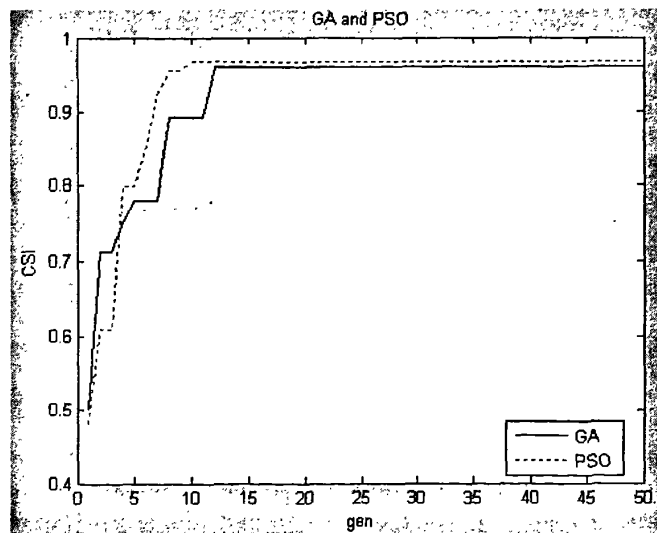


图 4-3 两种智能算法的对比

可见,用 400 个样本作训练时,PSO 算法的最优解出现在第 10 代,而 GA 算法的最优解出现在第 12 代。

情况二：600 个样本作为训练集。

遗传算法和粒子群算法的训练结果如图 4-4、4-5 所示。

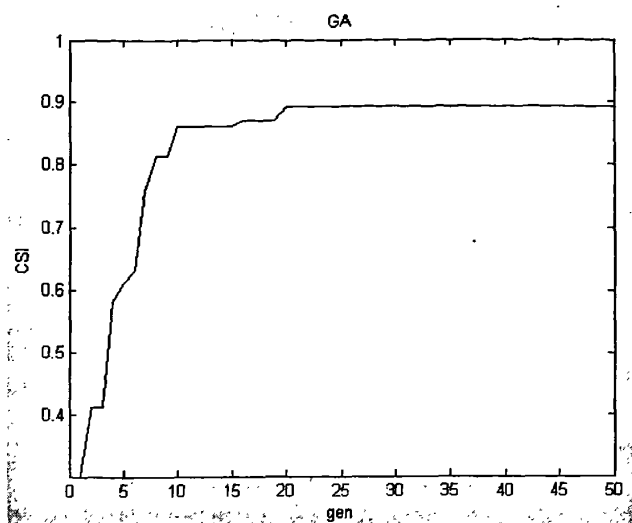


图 4-4 遗传算法的最优解随进化代数的变化情况

$(C, \sigma) = (91.0650, 1.7880)$, $CSI=0.9012$

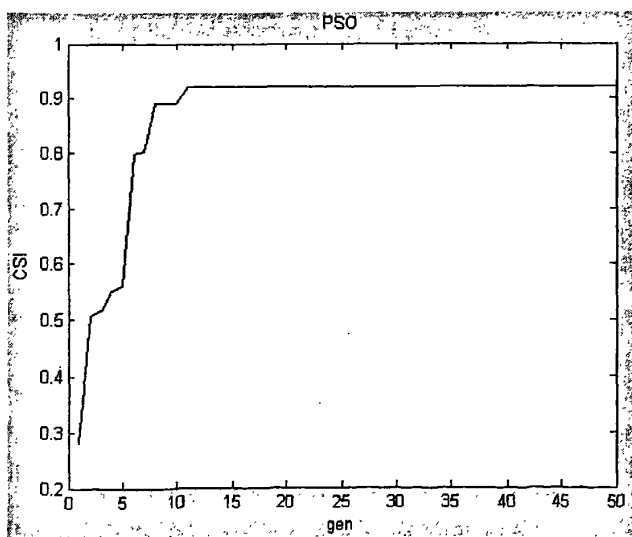


图 4-5 粒子群算法的最优解随进化代数的变化情况

$(C, \sigma) = (83.5197, 1.4938)$, $CSI=0.9304$

遗传算法和粒子群算法的对比如图 4-6 所示：

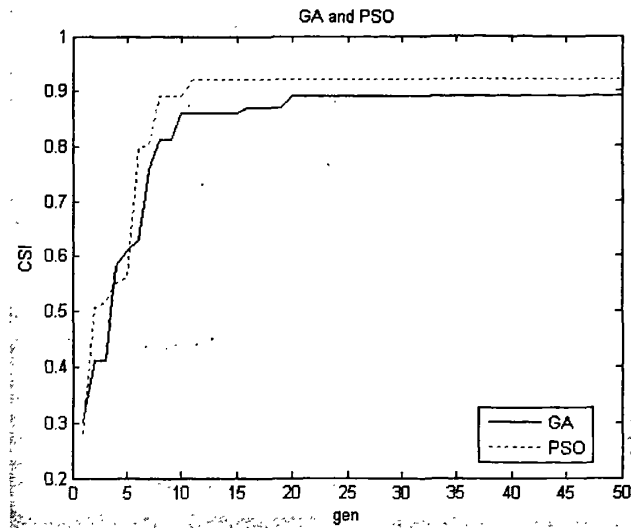


图 4-6 两种智能算法的对比

可见,用随机抽取的 600 个样本作训练时,从训练速度看,粒子群算法的最优解出现在第 11 代,遗传算法的最优解出现在第 20 代。

从上述训练结果看,粒子群算法寻找到的解好于遗传算法寻找到的最优解,即在训练过程中,利用粒子群算法得到的支持向量机参数对组合使得对沙尘暴预报的准确率高于遗传算法。

综上所述,无论用 400 个样本的训练集还是用 600 个样本的训练集作训练,粒子群优化算法在寻找最优解的效率及寻优结果上都好于遗传算法。

4.3.2 测试结果

训练过程的仿真只能说明算法一部分的性能,只有通过对测试集的测试才能说明算法所找到的参数的优劣程度。为了保证测试结果的可比性,本节利用随机抽取的同一组 1000 个样本作为测试集分别对以上四种训练情况下得到的训练结果进行测试,其测试结果如表 4-1 所示:

表 4-1 GA 算法和 PSO 算法的测试结果对比

训练方法	训练样本数	参数对(C, σ) 优化结果	预报准确率 CSI	测试次数
GA	400	(110.9376, 2.0274)	0.6140	5
	600	(91.0650, 1.7880)	0.6253	5
PSO	400	(119.9078, 1.7708)	0.6447	5
	600	(83.5197, 1.4938)	0.6675	5

测试结果表明：1) 利用 PSO 优化出的参数对组合进行沙尘暴预报，其预报准确率均高于 GA 算法预报准确率，可见，PSO 算法的寻优结果优于 GA 算法。2) 适当增加训练样本个数，可以提高预报准确率。3) 同一组参数对同一组样本的多次测试结果相同，说明了支持向量机分类器的稳定性。

文献[2]是基于手动调节方式选取支持向量机参数的测试结果如表 4-2 所示。

表 4-2 手动调节方法实现的 SVM 预测模型测试结果

样本集		SVM 预测模型	
训练样本数	测试样本数	(C, σ)	试报 CSI
600	1427	(16, 2^{26})	0.416
1000	1027	(3, 2^{20})	0.422
1627	400	(4, 2^{24})	0.582

通过对比表 4-1 与表 4-2 的对比可以看出，利用群智能算法得到的参数组合进行预报的准确率要远高于通过手动调节得到的预报结果。可见，将智能算法与基于 SVM 的沙尘暴预测模型相结合在很大程度上提高了效率和预报的准确率。

4.4 本章小结

本章首先详细叙述了遗传算法和粒子群算法在沙尘暴的预测模型——支持向量机参数优化中的应用，对两种不同算法的每一个具体的运行步骤和运行参数的取值都做了详细的阐述；然后对两种智能算法的训练过程的仿真结果和测试结果进行比较分析，得出了粒子群算法在此应用领域的性能优于遗传算法的结论。

第五章 改进的粒子群算法及仿真研究

粒子群算法是一种随机的、并行的优化算法,收敛速度较快,算法简单,容易理解,它不需要被优化函数具有可微、可导、连续等性质。但 PSO 算法也有一些严重的缺陷,一方面,容易陷入局部极值点,得不到全局最优解;另一方面,PSO 本身的参数设置如果不当,会导致寻优过程中粒子的多样性在很少的迭代中迅速消失,加上其寻优过程是对性能优良的点进行跟随的特性,使得算法容易进入“早熟收敛”。

5.1 PSO 算法的收敛性分析

PSO算法中,参数的惯性权值和学习因子对算法的寻优性能会产生很大的影响,下面分析各参数对算法收敛性的影响。对于公式(2-4), (2-5), 令 $c_1\text{rand}(1)=p_1$ 、 $c_2\text{rand}(2)=p_2$, 则在每一维上, 算法迭代的通式可以表示为:

$$v_i(t+1) = \omega v_i(t) + p_1(P_i - x_i(t)) + p_2(P_g - x_i(t)) \quad (5-1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (5-2)$$

为简化计算,假设粒子本身所找到的最优解的位置和整个种群目前找到的最优解的位置不变,故对算法的分析可以简化到一维进行。由式(5-1)、式(5-2)可以得到:

$$v_i(t+2) = \omega v_i(t+1) + p_1(P_i - x_i(t+1)) + p_2(P_g - x_i(t+1)) \quad (5-3)$$

$$x_i(t+2) = x_i(t+1) + v_i(t+2) \quad (5-4)$$

将式(5-2)、(5-3)代入式(5-4)中得到:

$$\begin{aligned} x_i(t+2) &= x_i(t+1) + \omega v_i(t+1) + p_1(P_i - x_i(t+1)) + p_2(P_g - x_i(t+1)) \\ &= x_i(t+1) + \omega(x_i(t+1) - x_i(t)) + p_1(P_i - x_i(t+1)) + p_2(P_g - x_i(t+1)) \\ &= (\omega - p_1 - p_2 + 1)x(t+1) - \omega x(t) + p_1 P_i + p_2 P_g \end{aligned} \quad (5-5)$$

$$\text{即: } x_i(t+2) + (-\omega + p_1 + p_2 - 1)x(t+1) + \omega x(t) = p_1 P_i + p_2 P_g \quad (5-6)$$

用特征方程法^[47,48]解以上二阶常系数非齐次差分方程, (5-6)式的特征方程为: $\lambda^2 + (-\omega + p_1 + p_2 - 1)\lambda + \omega = 0$, 此一元二次方程的解分三种情况:

情况一:

当 $\Delta = (-\omega + p_1 + p_2 - 1)^2 - 4\omega = 0$ 时, 方程有两个相等的实数根,

$$\lambda = \lambda_1 = \lambda_2 = -(-\omega + p_1 + p_2 - 1)/2$$

此时 $x_i(t) = (A_0 + A_1 t)\lambda'$, A_0, A_1 为待定系数, 可以由 $v_i(0)$ 和 $x_i(0)$ 来确定。

情况二: 当 $\Delta = (-\omega + p_1 + p_2 - 1)^2 - 4\omega > 0$ 时, 方程有两个不相等的实数根,

$$\lambda_{1,2} = (\omega - p_1 - p_2 + 1 \pm \sqrt{\Delta})/2$$

此时 $x_i(t) = A_0 + A_1 \lambda'_1 + A_2 \lambda'_2$, A_0, A_1, A_2 为待定系数, 可以由 $v_i(0)$ 和 $x_i(0)$ 来确定。

情况三: 当 $\Delta = (-\omega + p_1 + p_2 - 1)^2 - 4\omega < 0$ 时, 方程有两个不相等的虚数根,

$$\lambda_{1,2} = (\omega - p_1 - p_2 + 1 \pm i\sqrt{\Delta})/2$$

此时 $x_i(t) = A_0 + A_1 \lambda'_1 + A_2 \lambda'_2$, A_0, A_1, A_2 为待定系数, 可以由 $v_i(0)$ 和 $x_i(0)$ 来确定。

要使算法迭代收敛, 则当 $t \rightarrow \infty$ 时, $x(t)$ 趋向于有限值。以上三种情况下的 $x(t)$ 都收敛的条件是:

$$\|\lambda_1\| < 1, \|\lambda_2\| < 1$$

令 $p_1 + p_2 = p$, 则有:

当 $\Delta = 0$ 时, 收敛区域满足方程组为 $\begin{cases} (-\omega + p - 1)^2 - 4\omega = 0 \\ \|(-\omega + p - 1)/2\| < 1 \end{cases}$, 即为抛物线

$\omega^2 + p^2 - 2\omega p - 2\omega - 2p + 1 = 0$ 和 $0 \leq \omega < 1$ 共同围成的区域;

当 $\Delta > 0$ 时, 收敛区域满足方程组为 $\begin{cases} (-\omega + p_1 + p_2 - 1)^2 - 4\omega > 0 \\ \|(\omega - p_1 - p_2 + 1 \pm \sqrt{\Delta})/2\| < 1 \end{cases}$, 经过计算

收敛区域为: $\omega^2 + p^2 - 2\omega p - 2\omega - 2p + 1 > 0$, $p > 0$ 和 $2\omega - p + 2 > 0$ 共同围成的区域。

当 $\Delta < 0$ 时, 收敛区域满足方程组为 $\begin{cases} (-\omega + p_1 + p_2 - 1)^2 - 4\omega < 0 \\ \|(\omega - p_1 - p_2 + 1 \pm i\sqrt{\Delta})/2\| < 1 \end{cases}$, 经计算

收敛区域为: $\omega^2 + p^2 - 2\omega p - 2\omega - 2p + 1 < 0$ 和 $\omega < 1$ 所围成的区域。

综合以上三种情况, 由条件 $\omega < 1$, $p > 0$ 和 $2\omega - p + 2 > 0$ 共同决定的区域为收敛区域, 如图 5-1 所示:

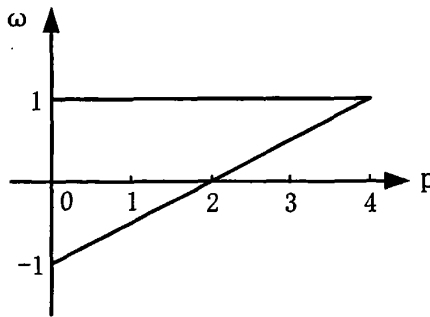


图 5-1 收敛区域

经过以上的收敛性分析可知, 要保证算法的收敛, PSO 中的参数之间需要满足一定的关系, 使得参数的设定有了一定的理论指导, 在收敛性分析的基础上,

本文对 PSO 参数的设定做了一些改进。

5.2 PSO 算法的改进

PSO 算法作为一种新出现的群智能算法,因其算法简单、容易实现等优点在优化领域得到了广泛应用,但基本 PSO 算法也存在容易陷入局部极小值点等缺点,为了提高算法的性能,目前已出现了多种对粒子群的改进算法,但多数是集中在对其参数惯性权重和学习因子的改进上。

5.2.1 惯性权值的改进

惯性权值 ω 控制粒子的全局搜索能力和局部搜索能力之间的平衡,较大的 ω 有利于跳出局部极小点,而较小的 ω 有利于算法收敛,本文引入Shi和Eberhart的研究成果^[49],采用了自适应调整 ω 的控制策略。算法的基本思想是:在运行的初期使用较大的惯性权值,使算法具有较强的全局搜索能力,后期则使用较小的惯性权值,以提高算法的局部搜索能力,即随着迭代的进行,线性的减少 ω 的值。惯性权值的计算公式如下:

$$\omega = (\omega_1 - \omega_2) \times \frac{\max(\text{iter}) - \text{iter}}{\max(\text{iter})} + \omega_2 \quad (5-7)$$

其中, $\omega_1=1.0$ 和 $\omega_2=0.6$ 是惯性权值的初始值和最终值, $\max(\text{iter})$ 代表迭代终止的最大代数。

对于惯性权重 ω 来说,不同的问题每一代所需的比例关系也不相同,这样,上面的线性递减关系只对某些问题有效,对于其它问题而言不一定最佳,所以以上方法也存在通用性差的弊端。

5.2.2 学习因子的改进

学习因子反映的是粒子“自我反省”和“向社会学习”的能力,基本粒子群算法中,参数的选取存在一定的盲目性,会对算法的性能和寻优效率产生一定的影响。以算法的收敛性分析作为指导,在对基本粒子群算法中惯性权值的选取方法进行改进的基础上,进一步对其学习因子进行改进。将 PSO 算法中的迭代公式简化为式(5-1)、(5-2)。

由收敛性分析可知,算法收敛满足的条件是:

$$\omega < 1, p > 0, 2\omega - p + 2 > 0$$

即

$$\omega < 1, p_1 + p_2 > 0, 2\omega - p_1 - p_2 + 2 > 0 \quad (5-8)$$

基本粒子群算法中, p_1, p_2 一般取 $(0, 2)$ 之间的随机数, 根据以上约束条件, 本文提出的改进方法为: 1) 按式(5-7)来设定惯性权值 ω , 显然满足 $\omega < 1$; 2) 选取 p_1 为 $(0, 2)$ 之间的随机数; 3) 选取 p_2 , 使其满足 $2\omega + 2 - p_1 > p_2$, 且 $p_2 > -p_1$ 。

5.3 改进粒子群算法的仿真结果

5.3.1 训练过程

由于粒子群算法是一种快速寻优的随机优化算法, 不需要对参数进行编码, 其收敛速度远远大于遗传算法、蚁群算法, 但容易陷入局部最优。本文即是针对此问题提出了对惯性权值和学习因子等参数取值的改进算法。为说明改进算法能避免早熟收敛的性能, 并便于与基本粒子群算法的性能作比较, 本节的仿真采用了与基本粒子群算法相同的训练集和测试集, 用含 400 个训练样本和 600 个训练样本的训练集对算法进行仿真。改进后的粒子群算法的仿真结果如下:

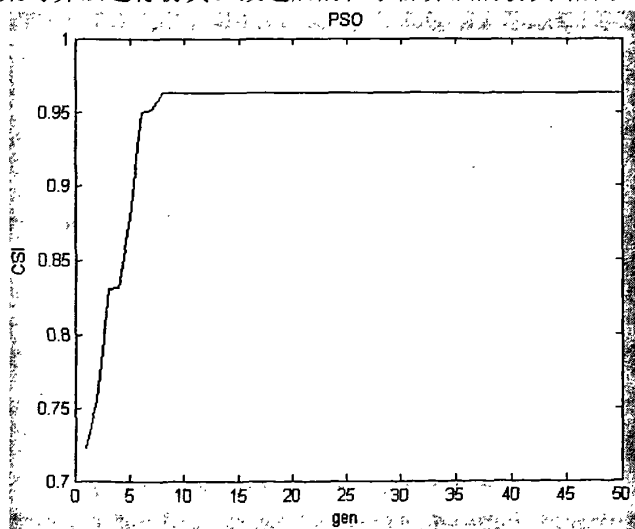


图 5-2 改进后 PSO 算法最优解随进化代数的变化情况
样本数=400, $(C, \sigma) = (127.1065, 1.3249)$, $CSI = 0.9625$

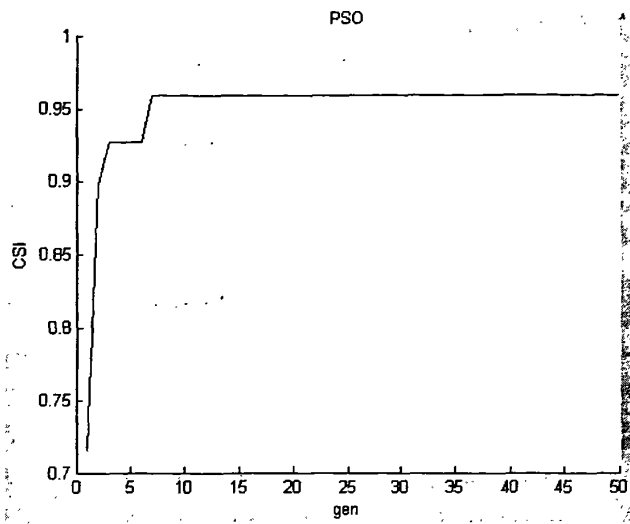


图 5-3 改进后的 PSO 算法最优解随进化代数的变化情况
样本数=600, $(C, \sigma)=(87.4727, 1.0423)$, $CSI=0.9590$

由以上仿真结果可以看出,改进后的粒子群算法针对不同的训练集,在快速性上都有了很大提高,尽管寻优过程不太相同,但都能在 10 代以内找到最优解,在不降低训练质量的同时大大地提高了寻优的效率。可见,改进算法的性能优于基本粒子群算法。

5.3.2 测试结果

表 5-1 改进 PSO 算法的测试结果

训练方法	训练样本数	参数对 (C, σ) 的训练结果	测试样本数	预报准确率 CSI	测试次数
改进 PSO	400	$(127.1065, 1.3249)$	1000	0.7654	5
	600	$(87.4727, 1.0423)$	1000	0.8406	5

测试结果如表 5-1, 由该表不难看出,改进后的粒子群算法所找到的最优参数对用于沙尘暴预测模型中,所得预报结果的准确率远大于基本粒子群算法,而且 5 次测试结果相同,表明支持向量机分类器具有较好的稳定性。

5.4 本章小结

本章首先从数学的角度对粒子群算法的收敛性做了详细的分析,得出了算法收敛与惯性权值、学习因子之间的关系,在此基础上提出了粒子群算法的改进算法,首先对惯性权值采取随迭代次数线性减小的改进方案,使得算法在迭代进行之初能在全局范围内搜索,在迭代后期算法收敛于局部解;然后再根据收敛性分析得出的约束关系来选择两个学习因子,这样就保证了算法的收敛性,训练和测试结果验证了改进的粒子群算法具有更快的寻优速度,同时改进后的粒子群优化算法比基本粒子群优化算法使得沙尘暴预测模型有更高的预测准确率。

第六章 总结与展望

6.1 本文总结

本文在沙尘暴预测应用背景下,针对两种典型的群智能算法——遗传算法和粒子群算法进行了深入研究,并提出了改进的粒子群算法。研究成果体现在以下几个方面:

1. 在学习分析支持向量机分类原理和沙尘暴样本特征的基础上,构建了基于 RBF 核支持向量机的沙尘暴预测模型。

2. 分析了支持向量机的参数对沙尘暴预测准确率的影响,将遗传算法和粒子群算法分别用于支持向量机的参数优化,通过仿真得出了在此应用领域粒子群算法的性能优于遗传算法的结论。

3. 对粒子群算法的收敛性进行了分析,得出了算法收敛与惯性权值 ω 、学习因子之间需要满足的关系,提出了改进的粒子群算法,一定程度上克服了粒子群算法容易陷入局部极小点的特性。对惯性权值 ω 的改进引入了Shi和Eberhart的研究成果^[49],实现了自适应调整 ω 的控制。另外,结合收敛性分析得出的结论,并在同时满足 ω 的选取策略以及 ω 和学习因子之间约束条件的情况下,选择学习因子。仿真结果验证了改进算法的有效性,表明改进算法比基本PSO算法具有更好的性能。

沙尘暴的预测已有多年的研究历史,基于支持向量机的沙尘暴预测模型是近年来提出的一种用于沙尘暴预报新模型。本文在分析了支持向量机参数确定方法的基础上,尝试了利用群智能优化算法优化支持向量机参数,仿真结果表明,群智能优化算法在基于支持向量机的沙尘暴预测模型的参数优化方面表现出了良好的性能,提高了沙尘暴预测的准确率,拓展了群智能算法的应用领域,推动了群智能算法的发展。

6.2 展望

本文的研究说明,以遗传算法、粒子群算法为代表的群智能算法在基于支持向量机的沙尘暴预测中具有良好的性能。随着对智能算法研究的发展和深入,群智能必将会在更多的应用领域内发挥作用。

下面将针对本文中存在的不足预测了群智能算法中仍需要进一步研究的问题。

1. 针对传统遗传操作容易使算法陷入早熟收敛的缺点, 构建新的遗传结构, 在算法中融入其他学科高效的算法和技术, 从整体上提高算法的可靠性; 对于大型的复杂问题可以利用遗传算法固有的并行处理性将其处理过程用并行计算机来实现, 提高搜索速度。

2. PSO算法是一种刚刚兴起的新算法, 其研究远没有尚未形成一套系统的分析方法, 有许多问题还需要进一步研究: PSO算法对最优点跟踪的寻优机制使得寻优过程容易陷入局部极值点, 寻找保持粒子多样性的方法是今后研究的一个方向; PSO算法中参数的选择很大程度上依赖于具体问题, 因此, 寻找一种有一定理论依据的参数设置的通用方法是PSO算法的研究中需要解决的另一个问题; 目前为止, PSO算法应用的最为成功的实例是神经网络训练^[50], 在其他领域的应用还停留在探索和进一步研究阶段, 扩大粒子群算法的研究领域也将大大的促进算法的发展。

3. 本文中改进的粒子群算法在使得算法的训练速度和寻优质量都有较大提高的同时增加了算法的复杂度。虽然从迭代次数上看改进后的粒子群算法在更少的代数内找到了最优解, 但每一代的运行时间却有所延长。对算法的改进的研究仍将是粒子群算法未来发展的一个方向。

参考文献

- [1]Vapnik V N. The Nature of Statical Learning Theory[M]. New York: Springer 2 Verlag, 1995
- [2]路志英, 张启孟, 赵智超, 基于 SVM 的沙尘暴预测模型的研究, 天津大学学报, 2006, 39 (9): 1110~1114
- [3]Reynolds C W Flocks.Herds and Schools:A Distributed Behavioral Model[J].Computer Graphics,1987, 21(4): 25~34
- [4] G.Syswerda, Uniform in genetic algorithms, ICGA'89, Morgan Kaufmann, 1989,2~9
- [5]L.Eshelman and D.Schaffer, Preventing premature convergence in genetic algorithms by preventing incest, ICGA'91, Morgan Kaufmann, 1991, 115~122
- [6]D.E.Goldberg,Genetic algorithm in search, optimization and machine learning,Addison-Wesley Reading, New York, 1989
- [7]S.A.Harp and T. Samad, Genetic synthesis of neural network architecture, In:L.Davis(editor), handbook of genetic algorithms, Van Nostrand Reinhold, New York,1991, 202~221
- [8]C.G.Langton, Studying artificial life with cellular automata, Physics 22D, 1986, 120~149
- [9]C.G.Langton,ed. Artificial life, Addison Wesley, 1989
- [10]KENNEDY J, EBERHART R. Particle swarm optimization[A]. Proc IEEE International Conference on Neural Networks[C]. Perth,1995, 1942~1948
- [11]Juang C F. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. IEEE Trans on Systems Man and Cybemetics(Part B -Cybemetics),2004, 34(2): 997~1006
- [12]Messerschmidt L, Engelbrecht A P. Learning to play fames Using a PSO-based competitive learning approach. IEEE Trans on Evolutionary Computation,2004, 8(3): 280~288
- [13]Kannan S,Slochanal S M R. Subbaraj P, et al. Application of Particle Swarm Optimization Technique and its Variants to Generation Expansion Planing Problem. Electric Power Systems Research,2004, 70(3): 406~413

- [14]Hiroyuki Mori, takuya Horiguchi. A Genetic algorithm based approach to economic load dispatching. In Proc.of the Second International Form on Applications of Neural Networks to Power Systems(ANNPS'93), Yokchama(Japan), 1993, 145~150
- [15]Yin P Y. A Discrete Particle Swarm Algorithm for Optimal Polygonal Approximation of Digital Curves. Journal of Visual Communication and Image Representation,2004, 15: 241~260
- [16]Parsopoulos K E, Papageorgiou E I, Groumpos P P, et al. Evolutionary Computation Techniques for Optimizing Fuzzy Cognitive Maps in Radiation Therapy Systems. Proceedings of the Genetic and Evolutionary Computation Conference, Springer,2004, 402~413
- [17]BERGH van den F. An Analysis of Particle Swarm Optimizers[D].Ph D thesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa, 2002
- [18]Shi Y, Eberhart R C. Parameter Selection in Particle Swarm Optimization. Evaluatory Programming V2(1998),Lecture Notes in Computer Science 1447, 591~600. Springer
- [19]Parsopoulos K E, Vrahatis M N. Modification of the particle swarm optimizer for locating all the global minima [J]. Artificial Neural Networks and Genetic Algorithms, Springer, 2001, 324~327
- [20]BRITS R, ENGELBRECH A P, Bergh van den F. Solving systems of unconstrained equations using particle swarm optimization [A]. IEEE Conference on Systems[C]. Man and Cybernetics, 2002, 6~12
- [21]BRITS R, ENGELBRECH A P, Bergh van den F. A niching particle swarm optimizer[A]. Proceedings of the Conference on Simulated Evolution and Learning, (Singapore), 2002
- [22]Shi Y. H. and Eberhart R. C. Fuzzy Adaptive Particle Swarm Optimization. Proceedings of the Congress on Evolutionary Computation, Seoul, Korea, 2001
- [23]Shi PJ.Using Selection to Improve Particle Swarm Optimization [A]. Proceedings of the 1999 Congress on Evolutionary Computation [C]. Piscataway, NJ:IEEE Press,1999, 84~89
- [24]Lovbjerg M, Rasmussen T K, Krink T. Hybrid particle swarm optimizer with breeding and subpopulations[A]. Proc. International Conference on Genetic and Evolutionary Computation[C]. SanFrancisco,2001, 469~476
- [25]Clerk M. The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization[C]. Proceeding of International Conference on Evolutionary Computation,1999, 51~57
- [26]N.Higashi, H.lbsa. Particle Swarm Optimization with Gaussian mutation[C].

- Proceedings of IEEE Swarm Intelligence Symposium, 2003, 72~79
- [27] Kennedy J, Eberhart R C. A Discrete binary Version of the Particle Swarm Algorithm[A]. Proceeding of the World Multiconference on Systems, Cybernetics and Informatics[C]. Piscataway, NJ: IEEE Press, 1997, 1404~1409
- [28] M. Lovbjerg, T. Krink. Extending Particle swarm optimization with self-organized critically[C]. Proceeding IEEE Conference on Evolutionary Computation, 2002, 1588~1593
- [29] Kennedy J. Small words and mega-minds: effects of neighborhood topology on particle swarm performance [A]. Proceedings of IEEE Congress on Evolutionary Computation[C]. Piscataway, NJ: IEEE Service Center, 1999, 1931~1938
- [30] Kennedy J, Eberhart R C. A Discrete binary Version of the Particle Swarm Algorithm[A]. Proceeding of the World Multiconference on Systems, Cybernetics and Informatics[C]. Piscataway, NJ: IEEE Press, 1997, 1404~1409
- [31] Goldberg D E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989
- [32] Eberhart R C, Kennedy J. A new optimizer using particles swarm theory. Pro Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995, 39~43
- [33] Shi Y H, Eberhart R C. A modified particle swarm optimizer. IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, May 4-9, 1998, 69~73
- [34] 李爱国, 覃征, 鲍复民, 等. 粒子群优化算法. 计算机工程与应用, 2002, 38(21): 1~3
- [35] 吕振肃, 侯志荣. 自适应变异的粒子群优化算法[J]. 电子学报, 2004, 33(3): 416~419
- [36] 曾建湖, 介婧, 崔志华. 微粒群算法[M]. 北京: 科学出版社, 2004
- [37] 刘淳安, 陈一虎. 基于带变异算子的粒子群优化算法, 计算机与自动化, 2005, 19(8): 38~40
- [38] 张荣沂. 一种新的集群优化算法——粒子群优化算法, 黑龙江工程学院学报, 2004, 18(4): 34~36
- [39] 高海兵, 周驰, 高亮. 广义粒子群优化模型, 计算机学报, 2005, 28(12): 1980~1987
- [40] 刘向东, 沙秋夫, 刘勇奎等. 基于粒子群优化算法的聚类分析, 计算机工程, 2006, 32(6): 202~217
- [41] 李国正, 王猛, 曾华军. 支持向量机导论, 北京: 电子工业出版社, 2005, 8~45

- [42]Zhang Xuegong. Introduction to Statistical Learning Theory and Support Vector Machines [J]. ACTA AUTOMATICA SINICA, 2000, 26(1):33~42(in Chinese).
- [43]Schölkopf B, Mika S. Input Space vs. Feature Space in Kernelbased Methods[J]. IEEE Trans on Neural Networks, 1999, 10(9):1000~1017
- [44]Vapnik V. Statistical learning theory [M]. New York: John Wiley&Sons, 1998
- [45]朱永生, 张优云. 支持向量机分类器中几个问题的研究[J]. 计算机工程与应用, 2003, 39(13):36-38
- [46]赵智超. 基于数据挖掘的沙尘暴智能系统得研究[D]. 天津: 天津大学电气与自动化工程学院, 2004
- [47]花栅, 计算机科学数学, 哈尔滨: 哈尔滨船舶工业学院出版社, 1994, 26~36.
- [48]卢开澄, 卢华明, 组合数学(第三版), 北京: 清华大学出版社, 2002, 145~150
- [49]Shi Y. H. and Eberhart R. C. Parameter selection in particle swarm optimization. 1998 Annual Conference on Evolutionary Programming, San Diego, March 1998
- [50]Kennedy J, Eberhart R C .Swarm intelligence [M]. San Francisco Morgan Kaufmann Publishers, 2001

发表论文和参加科研情况说明

发表的论文:

[1] 路志英, 李艳英, 杨乐, 《The Research on the Sand-dust Storm Forecasting Model Based on rough Sets》, JOURNAL OF TIANJIN UNIVERSITY, 在审.

参与的科研项目:

- [1] 国家级精品课程建设工作的网络课程制作
- [2] 基于沙尘暴预测的智能算法研究

致 谢

本论文的工作是在我的导师路志英副教授的悉心指导下完成的，路老师知识渊博，治学态度严谨，待生若子，不但在学业上给予了我无数次悉心的指导，而且在生活上给了我很多的帮助和鼓励，她所传授给我的科学的学习方法和做人的道理将使我受益终生。在此衷心感谢两年来恩师对我的关心和淳淳教诲。

感谢实验室的另一位孜孜不倦工作的老师王萍教授，她虽然不是我的导师，但却像导师一样关心着实验室的每一个人，让我感受到了家一般的温暖，另外王老师诲人不倦的忘我的工作精神深深的感动着我，为我以后走上工作岗位上了很好的一课。在此由衷的感谢王老师。

感谢岳士弘副教授对于我的科研工作和论文都提出了许多的宝贵意见，在此表示衷心的感谢。

自动化学院其他的各位领导和老师，他们教授了我知识，给我提供了好的研究条件，在此表示感谢，衷心希望在毕业后仍有机会向他们请教。

感谢包括代建辉、狄光敏同学在内的实验室的每一个人，在实验室工作及撰写论文期间，他们对我论文中的研究内容提出了一些宝贵的意见，为我的研究工作提供了一个良好的学习环境，给予了热情帮助，在此向他们表达我的感激之情。

感谢我的家人，他们的理解和支持使得我能安心的在学校完成我的学业。

最后感谢我的男朋友于春晓，论文的撰写期间是他的鼓励和支持、宽容和理解给我注入了力量。