分类号：TP39　　学校代码：10596
密 级：公开　　学　　号：1020190602

桂林理工大学
GUILIN UNIVERSITY OF TECHNOLOGY

硕 士 学 位 论 文

（全日制学术型硕士）

面向微生物 16S rRNA 序列的聚类与分类预测算法研究

研 究 生 姓 名：张佳
导　　师：崔建明 正高级实验师
学 科 专 业：软件工程
研 究 方 向：数据挖掘
所 在 单 位：信息科学与工程学院

二〇二二年四月

桂林理工大學
GUILIN UNIVERSITY OF TECHNOLOGY

Thesis for Master Degree

Research on Clustering and Classification Prediction Algorithm for Microbial 16S rRNA Sequences

Graduate Student：	Zhang Jia
Supervisor：	Senior Experimenter. Cui Jianming
Major：	Software Engineering
Direction of Study：	Data Mining
Affiliation：	College of Information Science and Engineering

April, 2022

研究生学位论文独创性声明

本人声明：所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含他人已经发表或撰写过的研究成果，也不包含为获得桂林理工大学或其它教育机构的学位或证书而使用过的材料。对论文的完成提供过帮助的有关人员已在论文中作了明确的说明并表示谢意。

学位论文作者（签名）：

2022 年 6 月 6 日

学位论文版权使用授权书

本学位论文作者完全了解桂林理工大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构及学校图书馆送交论文的印刷本和电子版本，允许论文被查阅和借阅。本人授权桂林理工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，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文，并通过网络向社会公众提供信息服务。（保密的学位论文在解密后适用本授权书）

本论文是否保密：是 否 如需保密，保密期限为： 年

学位论文作者（签名）： 导师（签名）：

2022 年6 月6 日 2022年6月6日

摘 要

微生物与人类的生活息息相关，测序技术的发展使得微生物宏基因组学跨入新的发展时期，基于高通量技术扩增的生物学实验产生了大量 16S rRNA（16S ribosomal RNA）序列信息，对产生的 16S rRNA 序列进行数据分析是生物信息学上一项严峻的挑战，其中一项主要的技术就是将 16S rRNA 序列进行聚类，从而分析环境中菌群物种丰度及多样性。目前，已经存在很多种不同的聚类算法，且均可实现有效聚类，因此，宏基因组学的研究者人员需要更多地考虑聚类算法的效率问题，其次，需要考虑如何更精准地通过已知 16S rRNA 序列的分类水平来推测未知序列的分类类别。本文的主要研究内容如下：

（1）针对 K-means 算法参数随机初始化的情况，本文考虑到网格聚类算法及密度聚类算法相结合带来的优势，结合 K-means++算法的思想策略，提出一种基于网格密度距离的 K-means 优化算法，并在聚类操作前对 16S rRNA 序列数据使用主成分分析法进行特征值提取，将维度降低，易于数据处理及可视化分析。优化后的算法在样本数量较大的数据集中，实现了初始聚类中心的稳定选取，同时减少了聚类迭代次数，提高了聚类稳定性。

（2）本文提出基于优化鸽群的 ELM 极限学习机的序列预测方法，主要针对数据库中没有的 16S rRNA 序列信息，通过机器学习构建模型来预测 16S rRNA 序列所属分类。神经网络模型通过学习大量 16S rRNA 序列数据的排列信息便可以做到高预测精度的分类预测。考虑到极限学习机具有较高的学习准确性和较低的运行时间，但是随机生成的输入层权重和偏置传播到隐含层解析出的输出矩阵若为非满列秩矩阵时，算法会出现计算问题。鸽群算法存在的问题是在地图和指针算子模型运算过程中，鸽群全部向位置较好的个体移动，容易陷入局部最优，对此引入遗传算法中交叉机制使鸽群巡航跳出局部最优。针对地标算子每次迭代都将鸽群数量减半，容易导致种群多样性降低，并且过早收敛，利用柯西变异来优化地标算子，提高种群多样性。将改进后的鸽群算法用于 ELM 网络模型的参数进行优化，提升极限学习机网络的性能，提高预测精度。

关键词： 16S rRNA；聚类；K-means；极限学习机

Abstract

Microorganisms are closely related to human life. The development of sequencing technology has brought microbial metagenomics into a new development period. Biological experiments based on high-throughput technology amplification have generated a large amount of 16S rRNA (16S ribosomal RNA) sequence information. Data analysis of 16S rRNA sequences is a serious challenge in bioinformatics. One of the main techniques is to cluster 16S rRNA sequences to analyze the abundance and diversity of bacterial species in the environment. At present, there are many different clustering algorithms, and all of them can achieve effective clustering. Therefore, researchers of metagenomics need to consider the efficiency of clustering algorithms. Second, they need to consider how to more accurately pass The taxonomic level of known 16S rRNA sequences can be used to infer the taxonomic class of unknown sequences. The main research contents of this thesis are as follows:

(1) In view of the random initialization of the parameters of the K-means algorithm, this thesis takes into account the advantages brought by the combination of the grid clustering algorithm and the density clustering algorithm, and combines the ideas and strategies of the K-means++ algorithm to propose a grid density-based algorithm. The K-means optimization algorithm of distance is used, and the 16S rRNA sequence data is extracted by principal component analysis method before the clustering operation, which reduces the dimension and facilitates data processing and visual analysis. The optimized algorithm realizes the stable selection of the initial cluster center in the data set with a large number of samples, reduces the number of clustering iterations, and improves the clustering stability.

(2) This thesis proposes a sequence prediction method based on the ELM extreme learning machine that optimizes the pigeon population. It mainly aims at the 16S rRNA sequence information that is not available in the database, and builds a model through machine learning to predict the classification of the 16S rRNA sequence. The neural network model can achieve classification prediction with high prediction accuracy by learning the arrangement information of a large amount of 16S rRNA sequence data. Considering that the extreme learning machine has high learning accuracy and low running time, but the randomly generated input layer weights and biases are propagated to the output matrix parsed by the hidden layer, if the output matrix is a non-full column rank matrix, the algorithm will A calculation problem occurred. The problem with the pigeon flock algorithm is that during the operation of the map and the pointer operator model, all the pigeon flocks move to the individuals with better

positions, which is easy to fall into the local optimum. For this, the crossover mechanism in the genetic algorithm is introduced to make the pigeon flock cruise out of the local optimum. excellent. For each iteration of the landmark operator, the number of pigeon flocks is halved, which will easily lead to the reduction of population diversity and premature convergence. The Cauchy mutation is used to optimize the landmark operator and improve the population diversity. The improved pigeon colony algorithm is used to optimize the parameters of the ELM network model to improve the performance of the extreme learning machine network and improve the prediction accuracy.

Keywords: 16S rRNA; Clustering; K-means; Extreme Learning Machine

目录

第 1 章 绪论

1.1 研究背景及意义

微生物种群的体型微小，但是却与人类的生活密切相关，在地球上各个生态系统中均有它们的存在。微生物不仅包括无法肉眼用观测的细菌、病毒、真菌、一系列小型的原生生物、部分藻类等，还包括一些可以被直接看到的菇类、灵芝等真菌生物群体[1]。微生物有些是没有益处的，可能使我们日常食用的产品组织结构发生不良变化，导致变质腐烂，人们食用后引起身体不适。一般来说，事物都具有双面性，微生物也存在对人们生产活动有益的种类，比如酒类的发酵，面食的发酵等。微生物在生物学方面上的特性涵盖了物种、遗传背景等多种信息，研究人员们可以从这些信息中更全面地了解微生物的多样性作用，同时为发展解决人类目前面临的环境生态问题提供更多途径。

宏基因组的定义是环境中生存着的所有微生物遗传物质的集合体[2]，宏基因组的出现为研究环境中微生物的群落结构提供了全新高效的技术平台，使人类逐渐深入认识微生物这一个陌生的世界，了解微生物种群的多样性，同时揭示了物种进化的动力。宏基因组学主要研究探讨微生物的多样性对生态环境的关系。实验人员从土壤或水质等样品中获得微生物群落组成以及微生物种群之间的相对丰度绝对丰度，这些研究对微生物资源的利用有着极为重要的理论和现实意义，比如环境治理改善、垃圾降解等。

测序技术推动宏基因组学的研究发展，由于高通量测序技术[3]（Next Generation Sequencing, NGS）的经济性和测序通量高等优势，被越来越多的用来解决环境微生物群落问题中的分析。目前，学者们对微生物群落的描述知识基本上都来自于高通量测序技术产生的 16S rRNA 标记基因。

基于高通量测序的 16S rRNA 测序技术是一种经济有效的技术，已成功广泛的被用于研究复杂微生物群落或环境样品的分析，它克服了传统分离培养的缺点，可以直接从生态环境中提取微生物的基因序列[4]。生态环境不止于土壤、海洋水质，还包括人体、动物等体内的有机生态环境。16S rRNA 序列是细菌基因 rRNA 编码上所对应的 DNA 序列，由于所有细菌的基因组中都含有，同时包含高变区和保守区[5]，所以 16S rRNA 成为深入分析菌群组成成分的强大工具。

通过高通量测序技术得到的 16S rRNA 序列，下游分析的第一步就是序列聚类成可操作分类单元（Operational Taxonomic Units，OTU）[6]。依靠 16S rRNA 序列片段之间的相似程度，对测序片段进行聚类操作，然后通过分析聚类操作的结果，进行判断聚

类簇中的序列是否是同一个物种，或者在是否在同一个分类水平上。

当下科学技术处在蓬勃发展的阶段，先进的技术、算法的出现使人们应接不暇，随着高通量测序技术的飞速发展，有关 16S rRNA 序列数据处理分析的方法也层出不穷，尽管已经存在很多种不同的算法，但是，相同的算法针对不同的数据集，或者不同的聚类参数，也可能会得到很大差异性的输出结果。在这样的情况下，就更加要求研究者们了解多种聚类算法的原理机制和性能特点，尤其是运行效率。在分析实验得到的序列数据时才能选择合适的算法，进一步选择使用先进的神经网络预测技术对其他数据中序列所属分类级别进行精准预测，特别是未被收录在比对数据库中的 16S rRNA 基因序列。

1.2 相关技术国内外研究现状

鉴于各种基因组测序项目的展开，聚集了大量生物数据。利用这些数据源进行知识挖掘分析成为计算机科学家和生物学家的一个重要且具有挑战性的课题。序列分析是生物信息学中诸多研究方向之一。 研究人员们随之需要面临的问题就是通过测序技术得到的序列数据的分析处理，16S rRNA 聚类是研究分析微生物菌群的组成及其分布的关键一步。聚类作为一种研究工具，在数据科学领域中占据着举足轻重的地位。聚类的工作方式一般是无监督学习[7]，其原理是基于数据集中不同样本之间的相似度给予样本标签，因而聚类可以在缺乏先验知识的前提下实现数据的分类。集群是数据对象的集合，相较于其他数据挖掘方法，聚类算法可以定义为将物理对象或者抽象对象根据一定的相似性度量方法归为集群的过程。聚类分析包括基于划分、基于层次、基于网格、基于密度以及基于模型的聚类方法等[8]。本文主要针对基于划分的聚类算法中的 K-means 算法进行研究。

1.2.1 K-means 算法应用研究现状

K-means 算法隶属于基于划分的聚类算法，具有简单易懂的数学逻辑、轻松容易的实现方法、快速敏捷的收敛速率等优点[9]。对于大量的高维数值数据，它提供了一种将相似样本分归到同一聚类中的有效手段。因此在诸如数据挖掘、信息检索、自然语言处理、模式识别、计算机视觉等领域都占据重要地位[10]。

由于 K-means 聚类算法的诸多优点，在生物信息领域被广泛使用，并且取得较好的效果。叶骁[11]使用 K-means 聚类算法有效的对肿瘤基因变异特征进行聚类，与其他主流检测工具相比，无监督的 K-means 算法识别更为精准。Angell I L[12]等人提出一种新颖的六聚体频率结合 K-means 聚类一种分析策略，基于六聚体频率的方法结合 K-

means 聚类，实现了对纳米孔测序数据从头识别和量化细菌种类，进行物种鉴定和验证，并通过实验使用六聚体 K-means 方法，确定了两种与阴道分娩相关的新的低丰度物种。Antoine G B 等人[13]对于一些高度特异性的表达谱，使用 K-means 算法与成分数据转换结合使用的方法，实现了识别更小且功能上更可解释的簇。王侠林等人[14]面向成年雌性北平顶猴 PMA 和 PMB 阴道菌群数据，分别使用 K-means 算法和 PCA 方法进行聚类，结果表明 K-means 算法的聚类效果更佳精确。K-means 算法原理简单易懂，计算速度较快，具有可伸缩性，对于大规模数据集适用性更强，因而基于 K-means 算法的改进优化以及采用率都较高。

聚类是机器学习和数据挖掘领域长期存在的问题，因此引发了大量的研究。传统的聚类方法，例如 K-means 和高斯混合模型 (GMM)[15]，完全依赖于原始数据表示。随着数据类型的多样化以及数据量的大幅度增长，传统的 K-means 算法表现出一定的局限性。K-means 聚类方法基于聚类数量固定的假设对数据集进行划分，该方法的主要问题是，如果要选择较少的聚类数量，则添加不同项目的概率较高进入同一组；另一方面，如果选择的集群数量多，那么在不同组中添加相似项目的机会就更高。为此，颇多学者关于传统 K-means 算法的缺欠之处作出了改善举措。曾俊[16]把原始数据分类，设定一个与密度大小成正比的密度参数 ϑ，通过密度参数优化 k 个样本数据的聚类中心点选取，进行聚类收敛，通过实验证明收敛速度明显加快。黄松等人[17]提出改进的遗传 K-means 聚类算法，采取使用并行计算的方式同时进行 K-means 算法聚类，以此减轻 k 值选择和初始聚类中心点的确定对聚类结果产生的影响，借助改进遗传算法的遗传算子来提高聚类算法的效率，根据类内平均距离距和不同类之间的距离设计适应度函数，可有效保证聚类结果的准确度，值得考虑的一个问题是，设置 k 值范围过于宽泛时，并行方式要求较高配置的计算机硬件。Shi H 等人[18]使用排序邻域法对数据进行预处理，依靠遗传算法的特点对数据样本进行降维，从而完成优化对初始聚类中心的拣选，提升 K-means 算法的精确度，取得了较好的分类效果。

Hossain M Z 等人[19]提出的方法动态地执行数据聚类。所提出的方法最初将阈值计算为 K-means 的质心，并基于该值形成集群的数量。在 K-means 的每次迭代中，如果两点之间的欧几里得距离小于或等于阈值，则这两个数据点将在同一组中。否则，所提出的方法将创建一个具有不同数据点的新集群。结果表明，所提出的方法优于原始的 K-means 方法。Yang M S 等人[20]考虑到大多数多视图 K-means 聚类算法在聚类过程中都不能减少特征。一般来说，如果在聚类过程中存在不相关的特征分量，则聚类算法必须花费更多的计算时间，甚至会产生不正确的聚类结果，尤其是对于多视图数据。因此，多视图 k 均值聚类算法的特征减少模式变得很重要。多视图数据集中也存在较高的特征维度，因此有必要考虑降低其维度用于聚类算法。对此，提出了一种新型的多视图K-means，称为特征减少多视图K-means（FRMVK）。主要做法是构建了多视图

K-means 算法自动计算个体特征权重的学习机制，可以减少每个视图中的一些不相关的特征分量。并且首次提出了一种新的多视图 K-means 目标函数，用于构建多视图聚类中特征权重的学习机制，并考虑使用小权重消除不相关特征的模式来减少特征。

Zhang G 等人[21]为提高 K-means 算法的准确性和稳定性，解决确定最合适的聚类数目 k 和最佳初始聚类中心的问题，提出了一种基于密度聚类 Canopy 的改进 K-means 算法。首先计算数据集中各样本的密度以及类内样本平均距离和不同类之间的距离，选择密度最大的采样点作为第一个簇中心，从数据集中去除密度簇。定义样本密度的乘积，簇内样本间平均距离的倒数，簇间距离作为权重乘积，其他初始种子由剩余数据集中的最大权重乘积确定，直到数据集中的数据为空。经过实验对比证明了改进后的算法的聚类精度和较好的抗噪性。Bai L 等人[22]考虑到聚类是无监督的，现有的大多数集成方法都试图获得与基聚类最一致的聚类结果。假如在具有非线性可分聚类的数据集上，如果基础聚类是由一些线性聚类生成的，这些方法通常无法将它们整合以获得良好的非线性聚类。因此，选择 K-means 作为基础聚类器，从基聚类中提取局部可信标签、不同基聚类的产生、聚类关系的构建以及每个对象的最终分配，提出多个 K-means 的集成聚类器算法，表示基于局部假设的聚类。实验表明所提出的集成聚类器不仅继承了 K-means 的可扩展性，而且克服了它只能找到线性可分聚类的局限性。

Fard M M 等人[23]提出了一种基于目标函数的连续重新参数化的 K-means 聚类的新方法，通过将 K-means 聚类损失视为可微函数极限的 K-means 算法和学习表征联合聚类。真正联合了简单的随机梯度下降更新、表示和 K-means 聚类损失，除了可以在所有方法中使用的预训练之外，这种方法还可以依赖确定性退火方案来进行参数初始化。通过确保使用相同的架构、初始化和小批量，在几个数据集上进行实验证实了 Deep K-means 在所有实验数据集上的良好行为，具有可扩展性。Wang S 等人[24]提出了一种新的基于 K-means 的聚类算法来处理不完整的数据，它将聚类和归类统一到一个目标函数中，把插补和聚类步骤集成到一个过程中，这两个过程相互引导，以实现更好的聚类。数据矩阵的缺失的特征交替插补，以便更好地进行聚类任务并揭示每个聚类中的内部结构。

本文基于 K-means 算法，主要对其过程中的参数初始化进行改进，结合网格的聚类算法和密度的聚类算法的特点，以及 K-means++算法思想策略，优化原始 K-means 算法初始聚类中心选取的随机性，使用 16S rRNA 序列数据进行特征值提取后应用优化后的 K-means 算法聚类，较为稳定的控制了初始中心的选定，一定程度上提高了算法效率。

1.2.2 极限学习机研究现状

数据驱动科学技术发展的环境下，宏基因组学也自然的将机器深度学习应用于捕

捉序列数据中的关联规律，进一步推测未知的生物学假设。深度学习从大型数据集中有效挖掘分析其中信息，为自然语言处理、图像处理等诸多领域均做出重要贡献，因而，研究人员首要考虑这项技术来对宏基因组学中的数据进行建模。 Zhao Z 等人[25]提出了一种用于微生物 DNA 序列数据的集成深度学习模型，模型利用卷积神经网络、递归神经网络和注意力机制来预测分类学分类和样本相关属性，并应用于短 DNA 读数和 16S 核糖体 RNA 标记基因的完整序列，以识别微生物群落样本的异质性。目前，机器学习中的数据输入要求为数字形式的值或者数字特征矩阵，但实验下机得到的数据仍是字符或者字符串形式的文件，所以需要编码字符格式数据为数值或数值特征矩阵。编码序列数据的方法通常有顺序编码基因序列、独热（one-hot）编码基因序列和把基因序列作为一种独特的独立语言，也就是 k-mer 计数法。

近年来，机器学习算法中的极限学习机（Extreme Learning Machine，ELM）由于其较好的灵活性，并可以直接应用于具有同质模型的不同学习任务，为回归、分类、聚类等提供了统一的框架，被广泛使用。ELM 不需要参数的调整，只需一次输入权重和偏置即可。Wang D 等人[26]基于蛋白质序列数据，使用两种主要的神经网络分类器，BP（Back Propagation，BP）神经网络和 ELM 网络进行序列分类的性能比较。研究表明，与传统的 BP 神经网络分类器相比，ELM 网络分类器需要的训练时间要少得多，分类精度略好于 BP，并且 ELM 没有需要调整的参数，可以很容易地实现。同时，ELM 中可以使用许多非线性激活函数和核函数。它的核函数可以是任何非线性有界可积函数，几乎在任何地方都是连续的。Rasheed Z 等人[27]提出基于序列组成的 TAC-ELM 的极限学习机进行宏基因组分析。TAC-ELM 使用极限学习机的框架来快速准确地学习神经网络模型的权重，输入特征由 GC 含量和寡核苷酸组成。在两个宏基因组基准上进行实验评估，结果表明了 ELM 网络框架的优势，它在准确性和实现复杂性方面优于其他较为先进的神经网络分类器。每个算法都有其优势和局限性，极限学习机也不例外，为了进一步提高 ELM 的性能或满足某些特定的应用要求，研究人员在过去几年中研发了各种基于 ELM 一定程度的优化改进。Zheng Y 等人[28]基于内核极限学习机，提出基于混合熵的内核极限学习机（KELM）的方法。KELM 是基于内核的 ELM，它将基本 ELM 扩展到了内核学习。根据核方法的思想，ELM 中的显式特征映射可以替换为由核定义的隐式映射，从而不需要手动调整隐藏节点的数量，也不需要在 KELM 中随机分配输入权重和偏置。此外， KELM 为 ELM 提供了与几种经典学习方法的结合点，包括径向基函数网络[29]、最小二乘支持向量机（SVM）[30]和近端支持向量机[31]，从而在一定程度上丰富了 ELM 理论。由于 ELM 和 KELM 依赖于最小均方误差准则，不适用于复杂的噪声环境，特别是在某些数据受到离群点干扰的情况下，因此提出将最大混合 correntropy 准则作为优化准则，从而提高 KELM 的鲁棒性。

唐延强等人[32]提出了改进粒子群优化 ELM 算法，改进的粒子群算法（IPSO）算

法的惯性权重和学习因子随着迭代次数自适应调整，实现两个参数的变化，在初始阶段具有较大的搜索范围和较快的搜索速度，在后期具有较强而稳定的收敛能力。另外，粒子群算法很容易困于局部搜索中。因此提出了一种粒子停滞扰动策略，将陷入局部最优的粒子重定向到全局最优飞行。通过改进 PSO 自适应调整全局和局部寻优能力优化 ELM，使预测结果更加准确，并在保持快速收敛的同时，提高算法的稳定性。Liu Z F 等人[33]提出基于改进鸡群优化器的极限学习机预测模型。首先针对传统的鸡群优化算法（ CSO ）在处理更复杂的问题时全局搜索和局部搜索能力较差。研究改进了 CSO 优化器，增强了 CSO 优化器的全局和局部搜索能力。在 CSO 优化器中，公鸡在整个种群占据主导地位，其觅食能力在种群中也处于领先地位。若种群中的主导公鸡陷入局部最优就会导致鸡群中所有个体同样陷入局部最优，为了解决这一问题，引入余弦惯性权重以增强公鸡的局部搜索能力。在小鸡粒子位置更新方程中引入最优粒子学习部分，以扩大小鸡粒子的搜索范围。在迭代后期，鸡群的搜索范围逐渐变窄，引入柯西变异算子，在迭代后期增强种群的多样性。将改进后的鸡群优化器用于极限学习机模型的参数优化。

Chen Y[34]考虑到经典的 ELM 算法由于随机初始化的隐含层神经元的参数往往表现出较差的稳定性，并且由于网络结构简单，不能很好地表示复杂的目标函数。由于网络结构不够复杂，残差中还有一些有用的信息没有被捕捉到，通过学习残差中的信息来修正预测值，使算法具有更大的性能，更好的拟合精度和泛化能力。由此提出了一种应用于回归的多层结构的深度残差补偿极限学习机模型（DRC-ELM）。第一层是基本的 ELM 层，这有助于通过学习样本的特征来获得目标函数的近似值。其他层是残差补偿层，其中通过在输入层和上层输出之间构建特征映射，将学习到的残差逐层校正为上一层获得的预测值，通过补偿拟合误差引起的残差进行回归。更详细的做法是利用残差神经网络的思想，加深网络结构，逐层补偿预测值，建立特征图，将上一层输出的预测值混合样本的原始特征作为输入，学习上一层输出的预测值和真实值之间的残差，预测的残差用来修正上一层的预测值，作为本层的输出。通过迭代对预测误差进行拟合和补偿，提高网络的预测精度，直到精度小于预定值或达到预设的补偿层数。

针对未知序列所属分类水平的预测，本文考虑使用极限学习机的网络模型，提出基于遗传算法交叉变异机制的优化鸽群的极限学习机神经网络学习算法对 16S rRNA 基因序列进行建模，训练学习序列数据排列的特征规律，预测其所属分类水平。

1.3 课题的组织结构

为了更好的对高通量测序技术得到的 16S rRNA 序列进行聚类，并实现序列分类水平上的预测，本文利用测序技术得到的基因序列，首先进行数据预处理，将预处理后

得到的干净序列再使用改进后的算法聚类，观察聚类结果，分析算法特点，最后利用主流神经网络模型，优化模型参数选择后，对数据展开建模和分类预测。本文主要框架结构如下：

第一章，主要对课题的研究背景和研究意义，以及该课题涉及到的技术知识的目前研究国内外现状进行简单阐述。针对研究现状存在的局限性，技术发展的快速性，以及不同算法存在的必要性，引出本课题研究内容，章末最后介绍了本文的章节内容结构。

第二章，主要介绍了基于网格的聚类算法和基于密度的聚类算法的原理机制和算法特点，阐述了目前最常用的两种针对高维数据进行特征值提取的算法，并分析了两种算法的区别，然后给出了三种主流的常用来组合使用的群智能优化算法，最后介绍了常用于自然语言处理的两种神经网络模型的原理。

第三章，着力于解决 K-means 算法选取初始聚类中心的随机不确定性，提出基于网格和密度的优化方法，并通过距离控制来确定合适且稳定的初始集群中心的选取，通过一系列相关实验对比表明改进优化后的算法一定程度的提高了准确性和迭代稳定性，并且拥有较少的迭代次数。

第四章，针对 ELM 极限学习机由于随机的权重阈值可能会出现运行过程的计算问题，提出基于优化鸽群的 ELM 极限学习机算法。初始化参数后，使用鸽群优化算法，首先用遗传算法中交叉机制优化的指针算子提高鸽群搜索全局能力，寻找全局最优，其次转为地标算子，并使用柯西分布的变异机制，提高鸽群多样性，避免过早收敛，深度寻找局部最优。将鸽群算法优化后得到的参数输入 ELM 学习机网络中对训练集进行特征值学习，最后对测试集进行预测分类。经过仿真实验对比，以及多项误差指标表明本文提出的改进 ELM 模型算法的有效性。

第五章，主要总结全研究文内容，并进一步分析阐明本文存在的不足，以及今后工作的研究目标和重点。

1.4 本章小结

本章首先叙述了本选题的研究背景和意义，接着针对该选题中使用的聚类算法 K-means 的应用研究以及前反馈单隐含层网络模型极限学习机研究的国内外的研究现状进行介绍，最后阐述了本文选题的主要研究内容和文章结构框架的安排。

第 2 章 相关技术理论知识

2.1 特征值提取

2.1.1 主成分分析法

统计学的发展使复杂的分析方法及多元统计方法应用于诸多领域，分析并解决各种各样的数学模型统计问题。主成分分析法[35]（Principal Component Analysis，PCA）是一种数学工具，其目的是使用少量的因素来表示数据集中存在的变化，是最广泛使用的工具来探索样本之间的相似性和隐藏模式。

主成分分析法作为常见的数据分析方式之一，无需考虑数据样本的类别输出，是一种无监督的技术。数据分析的过程中，有时很难找到属性之间的所有关系，PCA 允许将包含在最初相关数据中的大量信息转换为一组新的正交分量，从而可以发现隐藏的关系、增强数据可视化、异常值检测以及新定义的维度内的分类，即找到一个子空间，其基向量对应于原始空间中的最大方差方向。常用于提取数据的主要特征分量以及高维数据的降维[36]。

PCA 的目标是将 p 维的数据集 X 转换为较小维 L （$L < p$）的新样本集 Y，其中 Y 是 X 的主成分，数学符号表达如公式（2.1）。定义 X 为含有向量数量为 n 的集合，表示为$X = (x_1, x_2, \dots, x_n)$，集合中元素$x_i$为数据集中的一个样本。

$$Y = PC(X) \qquad \text{公式（2.1）}$$

首先计算数据集 X 的平均值，计算表达式为公式（2.2）。数值的分散程度，数学上使用方差来表述。一个变量的方差可以看做是每个元素与变量平均值的差的平方和的均值，数学表达式为公式（2.3）。

$$\overline{X} = \frac{\sum_{i=1}^{n} x_i}{n} \qquad \text{公式（2.2）}$$

$$S^2 = \frac{\sum_{i=1}^{n} \left(xi - \overline{X} \right)^2}{(n-1)} \qquad \text{公式（2.3）}$$

在一维空间中，数据间的分散程度可以使用数学概念方差来表示。对于多维空间的高维数据，协方差可以用来当作约束条件，协方差是用来表示两个变量之间的相关性的概念。相关性是指两个变量间的信息表示存在着相同的部分，二者并非完全独立。当两个变量之间不具有线性相关性的时候，可能更多地表示出数据原始信息。协方差的计算方法为公式（2.4）。

$$\operatorname{cov}\left(x_i, x_j\right)=\frac{1}{n-1}\sum_{i=1}^{n}\left(x_i-\overline{X}\right)\left(x_j-\overline{X}\right) \qquad \text{公式（2.4）}$$

提取数据的主要特征分量可以将一组 n 维向量维度降为 k ，其目标是确定 k 个单位的正交基，使得原始数据经过这组正交基的变换后，各变量两两间的协方差为 0，而变量间的方差尽可能大。求出协方差即可得到协方差矩阵，使用公式（2.5）表示。

$$X^{n\times n}=\left(x_{i,j}, x_{i,j}=\operatorname{cov}\left(Dim_i, Dim_j\right)\right) \qquad \text{公式（2.5）}$$

$X^{n\times n}$是 n 行 n 列的数据矩阵，Dim_i是第 i 个维数。主成分分析法的关键是协方差矩阵的特征值及对应的特征向量。特征向量用来确定新的特征空间的方向，特征值则决定其大小。假定一个$n\times n$的矩阵A，那么存在有一个非零向量x，$x\in\mathbb{R}^n$，被称为A的特征向量，如果Ax是x的标量倍数，用公式（2.6）表示。

$$Ax=\lambda x \qquad \text{公式（2.6）}$$

对于一些标量的 λ 。标量 λ 为矩阵 A 的特征值，由于特征向量是与矩阵 A 的特征值相所对应的并且满足该方程的非零向量，因此非零向量x被称为特征值 λ 所对应的特征向量。

$$\left(\lambda I-A\right)x=0 \qquad \text{公式（2.7）}$$

将集合 E 定义为满足公式（2.7）的所有向量x作为对应的特征空间，使用公式（2.8）来表示。

$$E=\left\{x:\left(A-\lambda I\right)x\right\}=0 \qquad \text{公式（2.8）}$$

在协方差矩阵中找到特征空间，接着将特征向量根据其特征值从大到小对进行排序，从而消除影响程度较低的成分，剩下提供原始数据的良好近似的主成分。

PCA 的特点有如下几点：

1．解决高维度问题：PCA 算法通过提取含有特征的信息，舍去部分不重要的信息，增大了数据样本的采样密度，降低了维数，从而缓解维度过高的问题。

2．降噪：数据分析过程中经常会被噪声数据所影响，排序末端的特征向量为噪声数据的可能性最大，因此舍弃最小的特征值指向的特征向量可以一定程度上缓解噪声影响。

3．过拟合：PCA 尽可能多的保留原始信息中的主要信息，舍弃一些可能没有价值的信息，但是舍弃的无用信息也可能是表示数据的重要信息，舍弃的信息仅没有表现在训练集上，因此，PCA 可能出现对数据过拟合的情况。当使用主成分分析法对数据进行特征值提取时，必须同时对训练集和测试集执行同样操作，如果将测试集零均值化，则该均值为训练集中数据均值，而不能是测试集数据的中心向量。

4．特征独立：经过主成分分析处理后的数据特征保持相互独立，并将原始数据的维度压缩降低。

2.1.2 线性判别分析法

1936 年，R. Fischer 首次提出监督学习的技术，线性判别分析（Linear Discriminant Analysis，LDA）。给定一些与描述数据相关的独立特征，LDA 创建这些特征的线性组合，从而产生所需类别之间的最大平均差异[37]。与主成分分析法相似，同样可以通过用求解特征值及其对应的特征向量，从而寻找一个投影超平面，即确定一个较低维数的子空间。这个超平面能够针对给定的独立特征进行降维、分类及解释。相对于原始数据样本维度，原始问题的数据点是可分离的。可分离性是依据平均值和方差的统计度量。LDA 的优点之一是可以通过求解一个广义特征值系统来得到解，这允许快速和大规模地处理数据样本。此外，线性判别分析可以通过内核技巧扩展到非线性判别分析。

定义数据集$N=\{(x_1,y_1),(x_2,y_2),\dots,(x_m,y_m)\}$，其中任意的数据样本$x_i$表示维度为 n 的向量。定义第 t 类数据的集合为 X_t，第 t 类数据的数量为 N_t，第 t 类数据的均值为 μ_t，第 t 类数据的协方差矩阵为 Σ_t，其中 $t\in[1,k]$。假设将 n 维数据集投影到 d 维的低维子空间中，变换的基向量为$(w_1,w_2,\dots,w_d)$，则基向量可以构成一个 $n\times d$ 的矩阵 W，则线性判别分析的优化目标为公式（2.9）。

$$J=\frac{W^T S_b W}{W^T S_w W} \qquad \text{公式（2.9）}$$

上述公式（2.9）中，S_b 表示类间散度矩阵，S_w 表示类内散度矩阵，数学表达分别

为公式（2.10），公式（2.11）所示。

$$S_b = \sum_{j=1}^{k} N_j \left(\mu_j - \mu\right)\left(\mu_j - \mu\right)^T \qquad \text{公式（2.10）}$$

$$S_w = \sum_{j=1}^{k} \sum_{x \in X_j} \left(x - \mu_j\right)\left(x - \mu_j\right)^T \qquad \text{公式（2.11）}$$

由于类间散度矩阵 S_b 中每个$(\mu_j - \mu)$的秩为 1，且矩阵的秩一定不会大于各个相加矩阵的秩的累加和，则协方差矩阵累加后秩的最大值为 k，但是当有了μ_1到 μ_{k-1}，就可以线性表示最后的μ_k，因此，类间散度矩阵 S_b 的秩最大值为 $k-1$，即线性判别分析降维的最大维度为 $k-1$。

线性判别分析对高维数据进行特征值提取，实现降维的流程步骤如下：

1．分别计算各类别中数据的均值向量和数据集中全部数据对象的均值向量。

2．计算类间散度矩阵 S_b 和类内散度矩阵 S_w。

3．计算矩阵 $S_w^{-1}S_b$ 的特征值及对应的特征向量。

4．选择 d 个较大的特征值及其特征向量，得到对应的投影矩阵，投影矩阵的每一列表示特征向量。

5．投影矩阵转置与原数据向量相乘，得到新的样本，实现对对数据集 N 进行降维，得到维度为 d 的降维数据集。

线性判别分析法 LDA 与主成分分析法 PCA 同样都是对高维数据集进行特征提取从而降低数据维度，并且二者均对数据集做了服从高斯正态分布的假设，使用了矩阵特征值分解的思想。不同的是，线性判别分析法降维最大只能降到比类别数少一维的维度，主成分分析法没有维度限制，可以降至二维并保证原有数据信息的特征。

线性判别分析法的优点在于监督学习，可以依据类别的先验知识来实现降维，LDA 搜索的是最能区分类的向量，而不是最能描述数据的向量。当数据样本区分类别信息平均值的作用大于方差的作用时，LDA 的表现比 PCA 更为优异。线性判别分析法存在的两个问题，一方面是在存在与问题维度相关的大量数据的情况下，对线性决策边界的限制可能过于严格。另一方面当存在许多相关特征时，线性判别分析可能会使用过多的参数并过度拟合数据，从而以较大的方差估计参数[38]。

2.2 聚类算法

2.2.1 基于密度的聚类算法

基于密度的聚类算法通过将数据视为基础密度函数的表示来工作，其中具有更多点的区域（即较密集的区域）是基础函数更可能产生结果的区域[39]。这些方法尝试根据这些局部的局部最大密度将点聚类为组，把周围的点归入高密度区域。

基于密度的聚类无需将集群中的样本数量作为算法的输入参数，无需假定数据集可能含有的潜在密度簇，也无需考虑数据集中可能存在的类内方差，因此基于密度的聚类属于非参数的算法，通常聚类被认为是密度 ρ(·) 的高密度区域。基于密度的聚类不一定是具有低成对聚类内相异性的点组，因此，不一定具有凸形但可以在数据空间中任意塑造。直观地说，数据空间中，基于密度聚类而得到的集群是分布在密度相对较高的不间断区域上的数据对象，通过低密度对象的连续区域与其他基于密度的集群分开[40]。基于密度的聚类可以想象成点的资产，这些点是通过某个密度水平的数据的概率密度函数“切入”估计而产生的。每个切入都会引起分离，对特征空间中概率密度高于截断值的连通区域进行评分。每个这样的区域对应一个包含所有落入该区域的数据点的集群。如果选择的级别太低，不同的集群将合并为一个集群；如果密度水平选择得太高，密度较低的簇将丢失。

基于密度的聚类形成由稀疏区域分隔的密集聚集对象的聚类，它的优点是可以发现任意形状的簇并轻松滤除噪声对象。DBSCAN、OPTICS 和 DENCLUE 是被广泛使用的基于密度的聚类算法。

DBSCAN 提出了两个对象的密度-连通性关系，并将聚类定义为密度连通对象的最大集合[41]。基于两个对象之间的密度连通关系，DBSCAN 给定的对象集 N，首先，将所有数据的状态设置为未确定的对象。对于每个未确定的对象 p，DBSCAN 计算邻居$L \in (p)$来确定 p 是核心对象还是噪声对象。DBSCAN 的时间复杂度为$O(n^2)$, 其中 n 是数据集 N 中的数据个数。如果是空间的预先构造包含所有对象的索引结构，这样对每个对象 p 可以在$O(\log n)$时间内计算邻居对象。因此，DBSCAN时间复杂度的值变成$O(n \log n)$。DBSCAN 的缺点是难以发现参数以获得最优的聚类结果，发现这些参数需要很多时间。OPTICS 是 DBSCAN 的扩展，解决了参数选择问题。OPTICS 同时对各种参数进行聚类，并创建有序的聚类结果，这使得能够轻松定位最佳聚类结果。DENCLUE[42]为每个对象定义了一个影响函数，对于一个唯一的对象 p，它定义一个密度函数为所有对象的影响函数值之和。将密度函数值为局部最大值的物体定义为密度吸引子。密度吸引子用于创建集群。DENCLUE 的执行速度比 DBSCAN 快，但是 DENCLUE，聚类结果的好坏受参数选择的影响很大。

下面介绍一种求数据集 N 中数据样本的近邻密度[43]的方法。定义数据集 N 中数据样本 f 到数据样本 g 的距离为 $d(f,g)$，给出任意正整数 k，定义 $Range\_k(f)$ 为数据样本 f 的 k 距离。那么 f 的近邻邻居由公式（2.12）得到。

$$Near_{Range\_k} = \{\{h\} \subseteq N \mid d(f,h) \le Range\_k(f)\} \quad \text{公式（2.12）}$$

数据样本 f 到数据样本 g 近邻距离定义为公式（2.13）。

$$Dist(f,g) = max\{d(f,g), Range\_k(f)\} \quad \text{公式（2.13）}$$

样本 f 的近邻密度数学表示为公式（2.14）。

$$\varepsilon(f) = \frac{\sum_{g \in Near_{Range\_k}} Dist(f,g)}{\left| Near_{Range\_k}(f) \right|} \quad \text{公式（2.14）}$$

上述公式（2.14）说明，如果数据样本点近邻密度值小，说明其近邻距离小，区域密度高。本文结合了基于网格的算法，更加关注与各个网格的密度，而不需要逐一计算每个样本点的近邻密度。

2.2.2 基于网格的聚类算法

基于网格的聚类算法执行通常与数据分析对象的数量无关，使用网格结构，由矩形块划分值空间作为执行对象。其工作思想是将原始数据空间划分为确定数量的独立网格单元的结构，对网格单元整体执行操作，这样，便将大量的数据样本点转换为少量的网格样本进行数据处理。因此算法处理时间与网格点数相关，减少了处理时间，提高了运行效率[44]。在挖掘大型数据集中的信息时，基于网格的聚类将数据空间分成一定数量的网格结构的单元，以网格结构中的单元为数据处理对象，聚类为集群。基于网格的聚类算法的有效性受到预定义网格的大小、单元格的边界、以及面对数据空间中形状和密度的局部变化时网格单元的密度阈值的限制。

此外，基于网格的聚类算法并不关心数据输入序列的分布情况，它与输入样本的数量大小成线性关系，当样本数量、样本维度增加时同样具有良好的伸缩性，可以有效的用于大型高维数据集间的聚类分析[45]。所以基于网格的聚类算法的优势在于其处理数据的较高速率，并且算法的运行时间并不与数据样本的数量密切相关，仅受制于

量化空间中各个维度上的网格单元。STING 算法、CLIQUE 算法是两种典型的基于网格的聚类算法。

该算法的缺点表现在如果网格结构的划分始终固定不变，那么就遮盖了数据信息的流动性，也就无法很好的处理流数据相关的内容。还有就是对密度阈值选择的高依赖性，如果阈值设置过高，也许会遗失部分低密度的聚类簇；如果阈值设置太低，又可能会分离本应该聚集的类，影响聚类结果的准确性。

因此，基于网格的聚类算法往往被用来联合其他算法综合使用，尤其与基于密度的聚类算法结合使用率更高。它们在空间信息处理领域被广泛运用。加之目前新进对大规模数据集处理算法的需求以及高伸缩性聚类算法的开发与改进，基于网格和密度的聚类算法在空间数据挖掘方面炙手可热。

2.3 群智能优化算法

近几十年来，群智能优化算法以其结构简单、求解效率高等优点受到众多学者的青睐。包括粒子群优化算法[46]（PSO）、蚁群优化算法（ACO）、人工鱼群算法[47]（AFSA）、人工蜂群算法[48]（ABC）、萤火虫算法[49]（FA）、和蝙蝠算法[50]（BA）、果蝇优化算法[51]（FOA）等。群智能优化算法在解决不同的优化问题中越来越受欢迎，并已成功在多项研究中用于优化特征选择。

群智能的最佳解决方案出现在群体本身中，但问题的解决方案不是事先知道的，而是在程序运行时自组织进行群体改变。自组织在其中起着重要作用适应性，最佳解决方案在不断变化的环境中做出较好的反应，修改了其自身行为，并自主地适应环境。因此，群智能必须满足五个原则，包括邻近原则、质量原则、多样化响应原则、稳定性原则和适应性原则[52]。

群智能算法的基本思想类似，均有如下五个步骤：初始化种群规模及一些必要的行为参数；定义迭代循环停止条件；定义适应度函数并对个体适应度进行评估；更新和移动可能的解决方案；返回全局最佳解决方案。其流程图表示如图 2.1 所示。

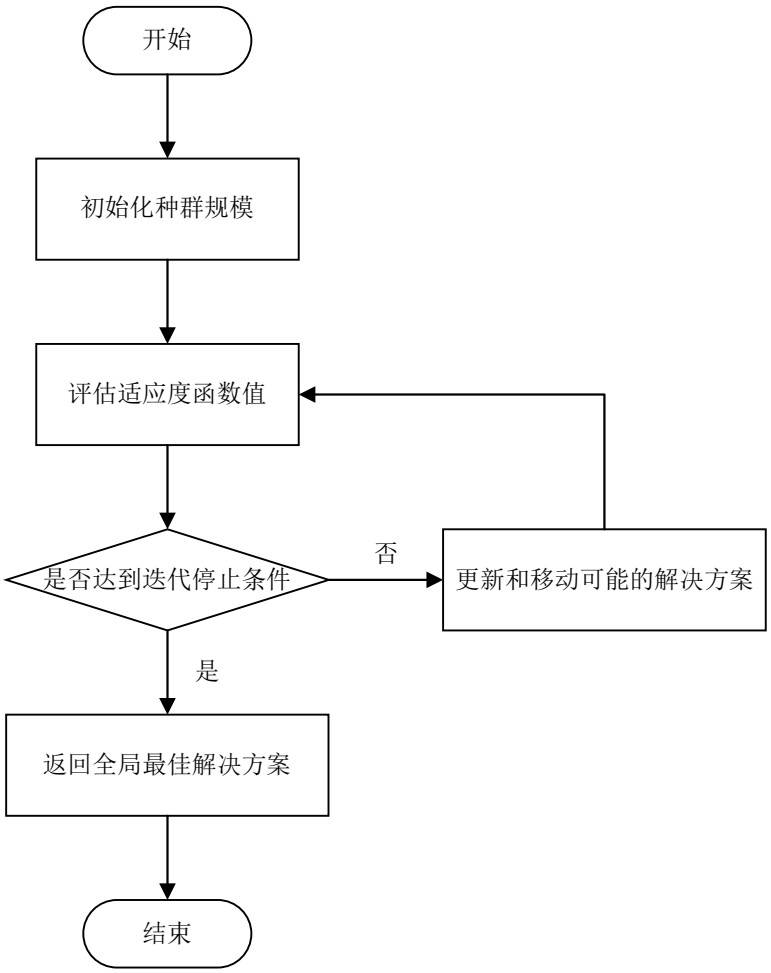

图 2.1 群智能算法流程图

2.3.1 遗传算法

遗传算法（Genetic Algorithm，GA）是一种受进化过程驱动的元启发式算法，也是概率解决方案，用于优化在生物学上以遗传评估过程为模型的问题，并专注于作为一种有效的算法来寻找许多类型问题的全局最优解[53]。GA 被用于不同的人工智能应用中，如面向对象系统、机器人技术和未来的新兴技术。

遗传算法是一种进化类算法，搜索方式受自然进化启发。该算法经常被用于创建高质量的解决方案，它会根据适应度不断找到更好的解决方案，非常适用于复杂的优化问题。遗传算法随机一定数量的个体构成初始种群，种群中个体分别代表不同的染色体，逐渐进化出一系列候选解，每个作为解决方案的染色体被编码转换为二进制的字符串。之后对染色体进行解码，并通过适应度函数来评估每个染色体个体。适应度函数用于评估群体的表现，进行选择并应用遗传参数，如交叉和变异。

标准遗传算法从随机生成的可能解决方案开始，即个体。根据适应度函数计算出的个体适应度的值，一般选择部分适应度较优的个体作为父代。对被选择的父代群体

通过交叉算子运算，产生新的种群，然后在新的群体中插入随机的基因来进行变异算子操作，就产生了一个新的群体可能具有的解决方案，获得对新生成的种群的适应度。重复用新一代替换上一代的迭代，直到满足停止标准。

遗传算法是效果显著的全局优化算法，具有多种优点，比如较高的鲁棒性、全局搜索的快速性等，但是算法收敛速率较慢，编解码过程复杂，参数选择往往依赖经验，反馈信息没有得到很好的利用[54]。

2.3.2 蚁群算法

蚁群算法是一类经典的群体智能算法，其灵感来自于蚂蚁使用信息素作为一种交流媒介，对信息素轨迹的铺设和跟踪行为[55]。该算法为了在不影响解质量的情况下进一步提高收敛速度，适用于组合优化问题。ACO 原理是来自蚂蚁组成的群体觅食行为，每只蚂蚁都会分泌一种信息物质，并且释放在路过的路径上供其他蚂蚁辨识。当其余蚂蚁进行路径选择时就会被这些信息物质所吸引，再通过识别每条路径上信息物质的浓度来选择最优行径方向。所以就会形成一种正反馈，信息物质越多的路径，会吸引越多的蚂蚁，然后留下更多信息物质，逐渐这条路径上的信息物质浓度将会更高，随着信息物质的挥发，会有更过的蚂蚁跟从。这种反馈机制就会帮助蚁群在面临多条路径的时候，通过识别信息物质的变化，最终找到一条最短最优路径。简化这种行为过程就是：离食物更短的路径上，经过的蚂蚁数量就会更多，蚂蚁释放的信息素的浓度就会变高，使得其余蚂蚁趋向这条路径，逐渐产生正反馈机制，就会找到最佳路径。

ACO 是常用于确定最优路径的仿生算法，具有以下几个优点[56]：适合与其他算法组合使用，适用于分布式并行计算，包括智能搜索，具有良好的全局优化和与其他群体智能算法相比具有很强的鲁棒性。此外，蚁群优化算法是具有代表性的不完全算法之一，不仅具有速度快、精度高的优点，而且可以快速找到准最优解。蚁群优化算法在解决简单问题时与其他算法相比可能没有显着优势，但平均而言，它在应用于相对复杂的问题时具有较高的效率， 并且在解决大规模集成优化问题上具有较低的成本。

2.3.3 麻雀算法

麻雀算法（Sparrow Search Algorithm，SSA） 的仿生学原理是把麻雀在觅食过程中的行为抽象为具有侦察和警告机制的发现者-跟随者模型[57]。麻雀是具有强烈记忆力的群居鸟类，通常，发现者适应能力强，搜索范围广，积极寻找食物来源，引导麻雀群体进行搜索和觅食，跟随者跟随发现者进行觅食。此外，证据表明，鸟类通常灵活地使用行为策略，并扮演不同的角色。也可以说，为了找到食物，每只麻雀会在发现者和跟随者两种间进行转换。在觅食过程中，个体监控其他个体的行为，一些麻雀负责观察周围环境，而其余的则寻找食物并注视观察环境的麻雀，如果观察环境的麻雀

发出警告信号，整个麻雀群体将立即进行反捕食行为，逃离危险，飞到另一个安全区域觅食。捕食者通常优先攻击边缘位置的麻雀，因此边缘位置的麻雀需要持续向种群中心移动调整位置，觅食区最危险边缘的麻雀，也有可能飞到别处。

麻雀算法是一种新颖的组织良好的元启发式算法[58]。该算法无需受制于目标函数的可微性、可导性和连续性，具有较强全局搜索能力、较好的算法稳定性以及较快的收敛速率等优点。但麻雀搜索算法高度依赖群体中的某个角色，缺乏学习能力，在高维复杂问题上仍然容易陷入局部最优。

2.4 神经网络模型

高通量测序技术使研究人员能够确定在不同生态环境中共存的微生物群落的集体基因组，不同群落内不同的物种丰度、长度和复杂性，再加上新物种的发现，使得对短 DNA 序列读数的分类分配问题极具挑战性。目前有许多研究方法可用于分类任务，例如决策树、统计技术和神经网络模型等。对于使用不同的预测分类模型存在很多研究，比如 BP 神经网络、卷积神经网络[59]、循环神经网络[60]、极限学习机等。

2.4.1 BP 神经网络

BP 神经网络通常是多层结构，输入层、隐含层和输出层组成基础的三部分。数据输入变量、输出变量的数目决定了输入层、输出层的神经元个数，而隐含层一般通过试错法来确定，层数可以是一层，也可以有多层[61]。一个隐含层层数为二的神经网络结构如图 2.2 所示 BP 神经网络具有较好的非线性映射能力、泛化能力和容错能力等优势，所以在人工神经网络中，拥有较高的使用率，但这并不表示 BP 网络没有缺点，在训练过程中，算法可能会出现陷入局部最优，算法收敛速度较慢，模型添加新的数据样本可能会影响到原学习的样本等情况。

BP 网络是一种根据误差进行反向传播调整的多层前馈神经网络，使用监督学习的机制，它可以在不确定输入与输出之间映射关系的情况下，推导学习并存储大量数据的输入和输出间映射关系的方程表达。该算法包括正向传播和反向传播两种过程。在正向传播部分，将实际输出值和预期输出值进行比较，如果输出结果符合理想范围，则学习结束。如果输出结果不理想，则可以通过均方误差或梯度下降的方法，反向传播进行网络参数的修正，以获得理想结果。

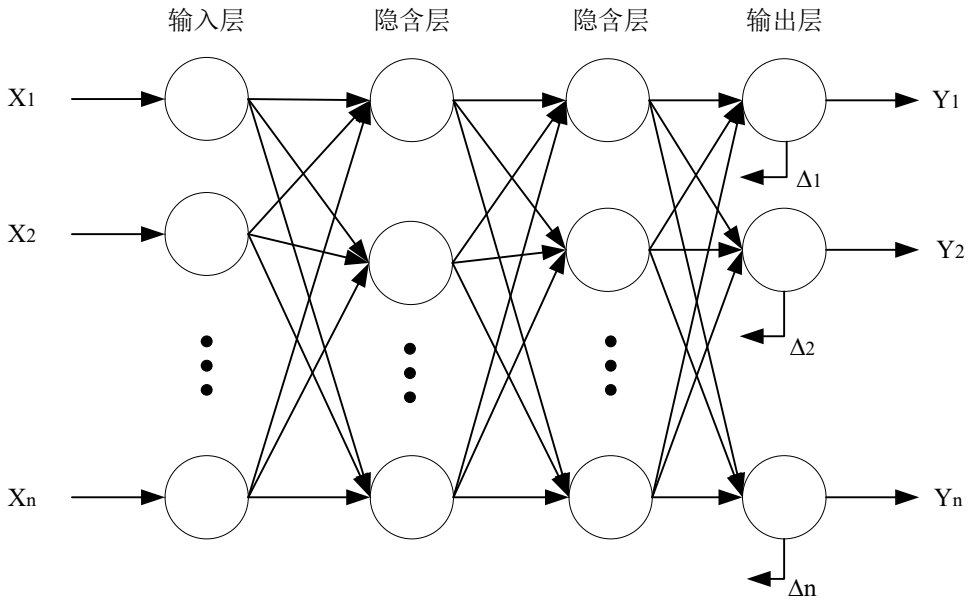

图 2.2 包含两个隐含层的神经网络图

BP 神经网络的学习过程[62]可以详细描述为：（1）数据信息的正向传播：数据输入从输入层经过隐含层传播到输出层。在数据信息的正向传播过程中，神经网络的权重和偏差保持不变，各层神经元的状态仅会影响到传播过程的下一层神经元的状态。正向传播结束后，网络计算结果从输出层输出，与期望输出结果进行比较，若偏差较大，则将偏差作为误差信息进入反向传播过程修正网络参数。（2）误差信息的反向传播：误差信息从输出层逆向经过隐含层传播到输入层。反向传播过程中，神经网络的权值通过误差信息的反馈进行修正，经过不断地权重和偏置连续修改调整，使得神经网络的实际输出尽可能地接近真实值。

定义简单的三层网络的输入神经元为 X_i，隐含层神经元为 Z_l，输出层神经元为 Y_j。输入层神经元与隐含层神经元间的权重为 w_{il}，隐含层神经元与输出层神经元间的权重为 v_{lj}。当输出神经元的期望结果为 t_j 时，则正向传播过程，该神经网络模型的隐含层神经元的输出计算如公式（2.15），输出层神经元的输出和输出误差数学计算分别如公式（2.16），公式（2.17）所示。

$$Z_l = f\left(\sum_i w_{il} X_i - \theta_l\right) \qquad \text{公式（2.15）}$$

$$Y_j = f\left(\sum_l v_{lj} Z_l - \theta_j\right) \qquad \text{公式（2.16）}$$

$$E=\frac{1}{2}\sum_{j}\left(t_j-Y_j\right)^2$$
$$=\frac{1}{2}\sum_{j}\left(t_j-f\left(\sum_{l}v_{lj}f\left(\sum_{i}w_{il}X_i-\theta_l\right)-\theta_j\right)\right)^2 \quad \text{公式（2.17）}$$

反向传播过程采用梯度下降法对各层的权值进行调节，权值的学习算法首先利用误差函数求导推出输出节点，计算方法如公式（2.18）所示。通过误差函数推导出隐含层节点的偏差，计算方法为公式（2.19）。

$$\frac{\partial E}{\partial v_{lj}}=\sum_{k=1}^{n}\frac{\partial E}{\partial y_k}\cdot\frac{\partial y_k}{\partial v_{lj}}=\frac{\partial E}{\partial y_j}\cdot\frac{\partial y_j}{\partial v_{lj}} \quad \text{公式（2.18）}$$

$$\frac{\partial E}{\partial w_{il}}=\sum_{j}\sum_{l}\frac{\partial E}{\partial y_j}\cdot\frac{\partial y_j}{\partial z_l}\cdot\frac{\partial z_l}{\partial w_{il}} \quad \text{公式（2.19）}$$

由于误差函数与权重Δv_{lj}和Δw_{il}的调整修正成比例关系，且沿梯度下降，因此隐含层和输出层的权重修正可以表示公式（2.20）。

$$\Delta v_{lj}=-\eta\frac{\partial E}{\partial v_{lj}} \quad \text{公式（2.20）}$$

在上述公式中，η表示学习速率。输入层和隐含层之间的权重修正表示如公式（2.21）所示，其中η'同样表示学习速率。在对权重修正的同时也需要对阈值变量 θ 进行调整，所应用的理论与修正权重时所用的理论相同。

$$\Delta w_{il}=-\eta'\frac{\partial E}{\partial w_{il}} \quad \text{公式（2.21）}$$

2.4.2 ELM 极限学习机

极限学习机是黄广斌教授所提出的，是一种简易有效的前馈神经网络（SLFN）学习算法[63]，且隐含层只有一层，其网络模型结构图如图 2.3 所示。传统的前馈神经网络需要调整结构中的全部参数，位于不同层的权重阈值和偏置参数之间存在依赖关系。单隐含层前馈网络的隐含层节点不需要与神经元相似，可以随机生成，并且可以保证

此类 SLFN 的通用逼近能力。ELM 不需要参数的调整，只需一次输入权重和偏置即可。

在过去的研究中，前馈神经网络学习通常采用基于梯度下降的方法，但是，基于梯度下降的学习方法具有明显地不足，主要表现在学习步骤不恰当导致运行过程非常缓慢，还有易于收敛到局部极小值。ELM 与具有反向传播过程的学习算法实现不同，它的方案基于加性神经元。对于基于加性神经元的 SLFN [64]，可以随机确定输入权重和隐含层神经元的偏置，通过分析确定输出权重。输入权重指的是连接输入层和隐含层神经元节点之间的权重，输出权重是指连接隐含层和输出层神经元节点间的权重。前馈神经网络可以被认为是一个简单的线性系统，随机生成输入权重和偏置，利用广义逆运算来解析获得 SLFN 的输出权重矩阵。

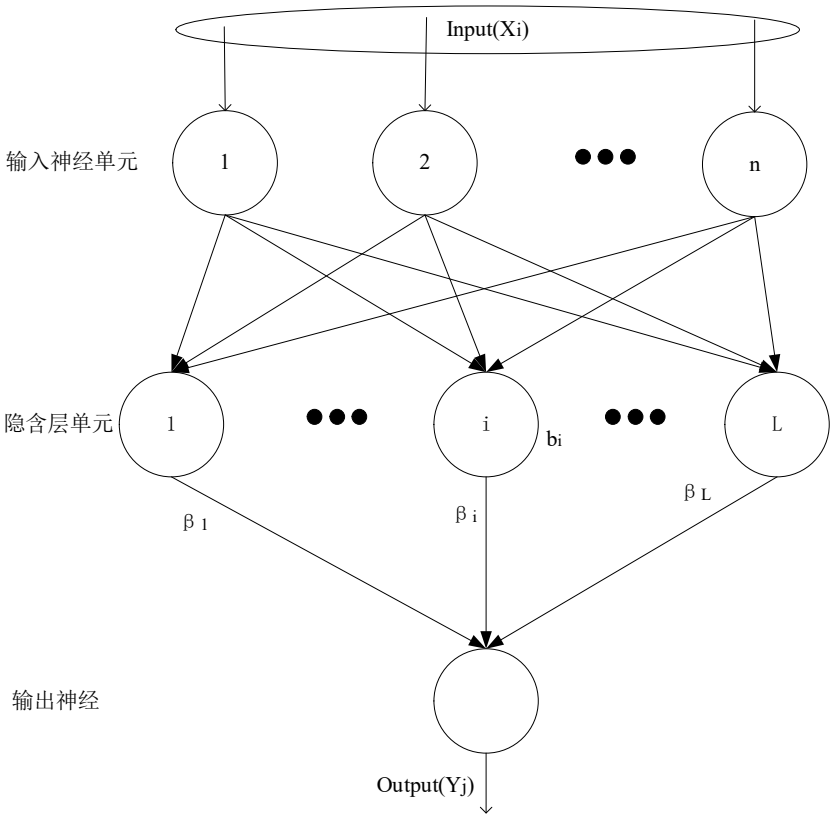

图 2.3 单隐含层前馈神经网络模型结构图

相比较于传统神经网络算法，ELM 的一个关键特性就是保证了学习准确性，降低了运行时间[65]，并且 ELM 往往具有良好的泛化性，易于实现，可以很好的避免学习率、陷入局部最小值等问题。此外，ELM 为回归、分类、聚类等提供了统一的框架。因此，ELM 非常灵活，并且可以直接应用于具有同质模型的不同学习任务。

极限学习机的理论原理简单概括是，定义 N 个任意不同的训练样本（X_i，Y_j），$X_i = [X_{i1}, X_{i2}, \dots, X_{in}]^T \in R^n$, $Y_j = [Y_{j1}, Y_{j2}, \dots, Y_{jm}]^T \in R^m$。设置隐含层到输出层的

非线性映射激活函数为$g(x)$,则一个具有 L 个隐含层单元的单隐含层神经网络可以表示为公式（2.22）。

$$\sum_{i=1}^{L}\beta_i g\left(W_i\cdot X_j+b_i\right), j=1,2,...,n \qquad \text{公式（2.22）}$$

其中，$W_i=[w_{i1},w_{i2},\dots,w_{in}]^T$代表输入权重，$\beta_i$代表隐含层第 i 个单元的输出权重，b_i为输入层到隐含层第 i 个单元的偏置。

神经网络学习的目的在于使得输出的结果误差尽可能的小，则有公式（2.23）这样的表示。

$$\exists\beta_i,W_i,b_i,\sum_{i=1}^{N}\left\|\beta_i g\left(W_i\cdot X_j+b_i\right)-Y_j\right\|=0 \qquad \text{公式（2.23）}$$

上述公式（2.23）使用矩阵形式表示可以写成如下公式（2.24）的形式。

$$M\beta=Y \qquad \text{公式（2.24）}$$

公式 2.24 中，M 为隐含层单元的输出矩阵，Y 是样本的期望输出矩阵，M、Y 的矩阵表达式为公式（2.25），公式（2.26）所示。

$$M=\begin{bmatrix} g\left(W_1\cdot X_n+b_1\right) & \dots & g\left(W_L\cdot X_1+b_L\right)\\ \vdots & \dots & \vdots \\ g\left(W_1\cdot X_n+b_1\right) & \dots & g\left(W_L\cdot X_n+b_L\right)\end{bmatrix}_{n\times L} \qquad \text{公式（2.25）}$$

$$\beta=\begin{bmatrix}\beta_1^T\\ \vdots \\ \beta_L^T\end{bmatrix}_{L\times m},\ \mathrm{Y}=\begin{bmatrix}Y_1^T\\ \vdots \\ Y_L^T\end{bmatrix}_{n\times m} \qquad \text{公式（2.26）}$$

ELM 输入权重和隐含神经元参数可以随机确定而不需要调整。对于输入保持不变的权重和偏置，训练 SLFN 相当于寻找关于线性系统 $M\beta=Y$ 的最小二乘解 $\hat{\beta}$。则该线性系统的唯一最小范数最小二乘解计算方法为公式（2.27）。其中，$M^{\dagger}$ 表示隐含层的输出矩阵 M 的广义逆矩阵。

$$\hat{\beta}=M^{\dagger}Y \qquad \text{公式（2.27）}$$

2.5 本章小结

本章主要介绍了本文中涉及到的相关技术知识，首先介绍了用于高维数据集的特征值提取和降维的算法理论，然后介绍了基于网格和基于密度的聚类算法原理，接着，阐述了常用来组合进行参数优化的群智能算法，包括遗传算法、蚁群算法、麻雀搜索算法，最后列出具有反向调节机制的 BP 神经网络模型和前馈神经网络 ELM 的结构及其原理知识。

第 3 章 基于优化的 K-means 算法的序列聚类

K-means 算法隶属于基于划分的聚类算法，具有简单易懂的数学逻辑、轻松容易的实现方法、快速敏捷的收敛速率等优点。对于大量的高维数值数据，它提供了一种将相似样本分归到同一聚类中的有效手段。因此各个领域的使用率都极为广泛。但对于大型数据集，K-means 聚类算法的随机初始参数的弊端就显露出来，本章主要针对 K-means 算法初始聚类中心的随机性，优化其聚类过程。一般情况下，聚类中心最终落于密度较高的区域中，因此，考虑优化 K-means 算法的初始过程，将样本点放入网格内，将网格作为数据处理对象，计算各个网格密度，然后在密度较大且距离相对较远的网格中选取 k 个数据样本作为初始聚类中心，以此来优化初始聚类中心的选定，增加聚类过程的稳定性。

3.1 K-means 聚类算法

K-means 算法包含于硬聚类类型算法，是原型的目标函数聚类方法的代表性范例之一[66]。其工作原理是把样本集作为输入，运用该算法对样本实行聚类过程，将拥有相似特征的数据样本聚成一类，同时，类别内的样本点尽量紧密集中在一起，而类别间的距离差异尽可能的明显，聚类结果则越理想。K-means 算法一般以欧几里得距离作为相似性度量达成对某一初始聚类中心进行最优划分。在处理大量数据时，该算法具有较高的可扩展性。

定义一个 D 维的欧几里得空间中的数据集$X=\{x_1,x_2,\dots,x_n\}$，集群中心$Z=\{z_1,z_2,\dots,z_k\}$，$y=[y_{ic}]_{n\times k}$用来指示数据点 x_i 是否属于第 c 个集群，其中，$y_{ic}\in\{0,1\}$，$c\in[1,k]$。K-means 的目标函数表示为公式（3.1），算法通过最小化 K-means 目标函数的必要条件进行迭代分别更新聚类中心 z_c 和集群中成员 y_{ic}，数学方程表达如公式（3.2），公式（3.3）所示。

$$f(y,Z)=\sum_{i=1}^{n}\sum_{c=1}^{k}y_{ic}\left\|x_i-z_c\right\|^2 \qquad \text{公式（3.1）}$$

$$z_c=\frac{\sum_{i=1}^{n}y_{ic}x_i}{\sum_{i=1}^{n}y_{ic}} \qquad \text{公式（3.2）}$$

$$y_{ic}=\begin{cases}1, \left\|x_i-z_c\right\|^2=\min\limits_{1\le c\le k}\left\|x_i-z_c\right\|^2\\ 0\end{cases} \qquad \text{公式（3.3）}$$

传统 K-means 聚类算法进行聚类是通过迭代计算对聚类中心点进行更好确立的过程。其基本步骤为：算法初始任意择取 k 个数据样本作为初始聚类中心，计算剩余数据样本与中心点间的距离，按照距离度量将样本点归到 k 个集群中，重新计算集群中新的中心点，通过聚类中心持续变动，依次计算数据样本与新集群中心之间的距离，将上述过程迭代循环，直至产生的聚类中心稳定不变，聚类结束。

合理的确定聚类数目 k 值和 k 个初始聚类中心点将对于聚类效果的好坏造成重要影响，获取最佳的聚类数目 k 和初始聚类中心是 K-means 算法的核心。k 值选择方法有核方法、Elbow 方法、差距统计、轮廓系数法、Canopy 方法等[67]。

Elbow 方法的基本思想是用每个簇中的样本点与聚类中心的距离的平方来给出一系列的 k 值，把误差平方和 (SSE) 用作性能指标。迭代 k 值并计算 SSE，SSE 值越小则表示各集群更收敛，聚类效果更好，当集群的数量设置为接近真实集群的数量时，SSE 显示出快速下降。当集群的数量超过真实集群的数量时，SSE 会继续下降，但会很快变慢。用于确定具有未知分类数的数据集的聚类数。

差距统计算法的基本思想是引入参考测量值，参考测量值可以通过蒙特卡罗抽样方法获得，计算每个集群中两次测量值之间的欧几里德距离的平方和，比较构建的参考零均值分布的聚类结果，以确定数据集中的最佳聚类数计算公式如公式（3.4）所示。$E_n^*(\log(W_k))$指的是$\log(W_k)$的期望，一般由蒙特卡罗随机生成。在样本多次定位的矩形区域中随机生成与原始样本数一样多的随机样本W_k，得到多个$\log(W_k)$，为了求它们的平均值，首先计算一个近似值，也就是上面所说的期望，P 是采样数，s(k)是加入的标准，最后就可以计算差距统计值$G_n(k)$，最大的差距统计值对应的 k 就是最佳选择。

$$G_n(k)=E_n^*\left(\log\left(W_k\right)\right)-\log W_k E_n^*\left(\log\left(W_k\right)\right)=\sqrt{\frac{1+P}{P}}s\left(k\right) \quad \text{公式（3.4）}$$

Canopy 算法简略地将原始数据分为多个含有相交部分的子数据集，每个子集都充当一个集群，通常使用低成本的相似性度量来加速集群，因此，对聚类算法进行初始化，Canopy 是一个较好的选择。构成子集需要设置距离阈值 T1 和 T2，并且规定 T1>T2，T1 和 T2 的设置可以根据用户的需要或者使用交叉验证获得。实现原理是将原始数据集 N 为按照一定的规则排序，在数据集 N 中任意选取数据向量 x，使用粗略距离计算方法计算数据向量 x 和数据集 N 中其他数据向量样本之间的距离 d。若数据向量计算得到的 d 小于 T1，将该数据向量映射到一个冠层中，d 小于 T2 的数据向量将不会被作为候选中心向量的选择。重复上述步骤，候选列表中无可选中心向量时停止操作。

对于小型数据集，现有的大部分 k 值选择方法均可适用，但是针对大型数据集，Canopy 方法就显现出了其优越性 。

3.2 基于网格密度距离的 K-means 优化算法

K-means++的初始聚类中心优化办法是基于经典 K-means 聚类算法，随机择选出一个数据点作为第一个聚类中心，对于后面*k-1*个聚类中心，均选择数据集中与已选中心点距离较大的点。本文基于 K-means 算法、基于网格的聚类算法、基于密度的聚类算法的特点，综合 K-means++的策略，提出一种基于网格密度距离的 K-means 优化算法（GDD-Kmeans），首先将数据样本放入网格中，计算每个网格单元中数据样本点的数量，即网格单元的密度，根据密度将网格降序排序，选取*k*个密度较大的网格且*k*个网格的距离较远，K-means 聚类的初始集群中心就在这*k*个网格中选择，这样便解决了随机任意初始聚类中心选取造成的影响。该算法执行步骤流程图如图 3.1 所示。

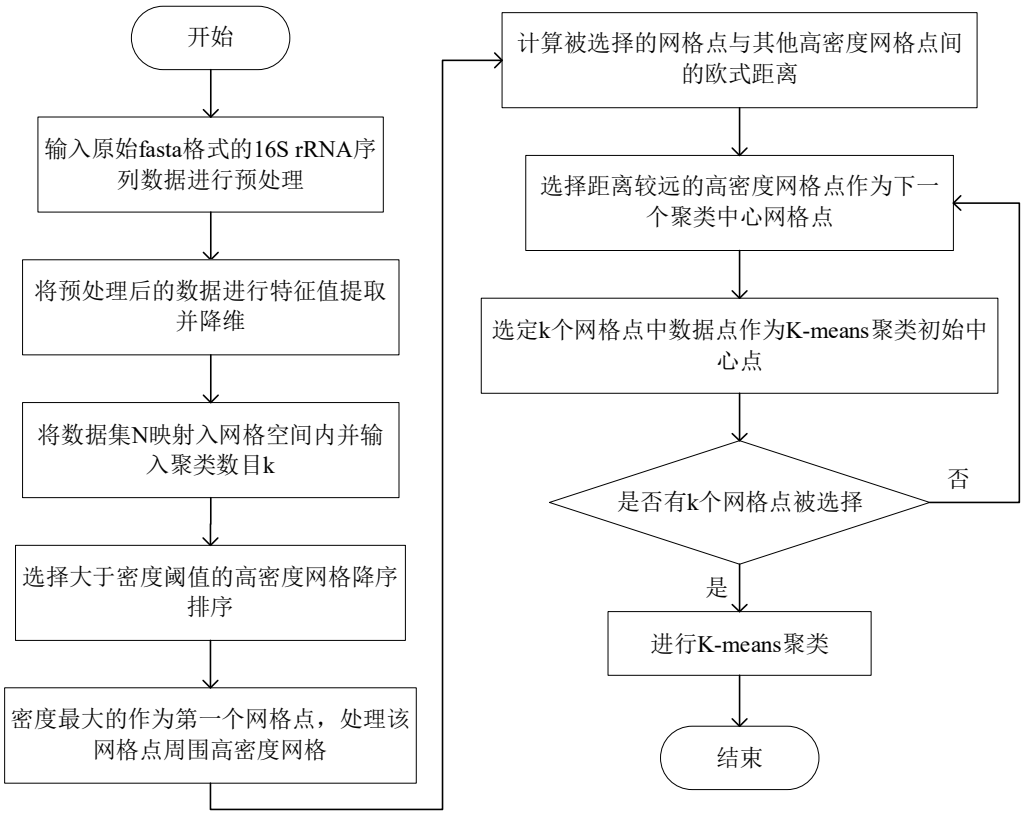

图 3.1 优化 K-means 算法流程图

GDD-Kmeans 算法实现详细步骤如下：

（1）网格化数据集

GDD-Kmeans 算法的第一步利用网格聚类算法的作用对象与数据样本数量无关的优点，将数据点放到网格空间中，把每个网格单元作为数据处理对象。

定义 n 个数据样本存在于数据集 N 中，其中各样本均包含 m 个属性，即数据集 N 的维度为 m，由公式（3.5）表示。数据集 N 由聚类中心 $z=\{z_1,z_2,\ldots,z_k\}$ 被聚类算法划分为 k 个类别簇。

$$N=\{a_1,a_2,...,a_n\},a_i=\{a_{i1},a_{i2},...,a_{im}\},i=1,2,...,n \quad \text{公式（3.5）}$$

合适的网格点数对后面以网格为单位进行数据分类有着重要作用，因此将各维度数据范围及聚类数目考虑入网格点的划分中。每个维度的样本点的数据范围 *Range* 计算如公式（3.6）。

$$Range_i=a_{i\max}-a_{i\min},i=1,2,...,m \quad \text{公式（3.6）}$$

当被聚类数据样本数量稀少时，我们可将网格点数设置为类别数 k。对于数据集中且数据量繁多的样本，仅将网格点数设为类别数，网格划分就很笼统，经实验验证，设置每个维度的网格点数 *GridNum* 为聚类数目的整数倍，根据数据样本点的分布情况适当调整。同时考虑到随着聚类数目的增大，网格点数也越来越多，造成计算量增大，耗费时间，而网格聚类算法的优势也不能得到很好的表现。为此，设置网格点数上限为 $\sqrt{n/2}$，可以相对有效的避免网格点数对聚类集群准确性的影响。由于不同维度的网格内样本数据范围不同，于是，不同维度的网格网格步长 *Step* 也会有所不同，计算表达式为公式（3.7）。

$$Step_i=\frac{Range_i}{GridNum},i=1,2,...,m \quad \text{公式（3.7）}$$

将网格点数及网格步长确定下来，数据集 N 中的 n 个样本点就转换为网格单元，将网格单元作为处理对象，从而大大减少了操作对象的数量。

（2）计算网格密度

聚类算法最终的聚类中心一般都落在密集数据点中，把数据样本映射到网格空间中，选取高密度网格中的数据点作为 K-means 聚类算法的初始聚类中心，以此实现较为稳定的 K-means 算法初始聚类中心的确定。

网格密度指的是网格空间网格内的数据点的总数。遍历网格内所有数据点得到所有网格的密度，定义密度阈值 *Threshold*，网格中样本点数量大于等于该阈值则该网格为高密度网格。密度阈值以样本数量除以网格点数得到平均密度为标准，计算表达式为公式（3.8）。

$$Threhold = \frac{n}{GridNum^2}$$ 公式（3.8）

（3）选择初始中心

把根据阈值选取的高密度网格进行降序排序，放入集合 B 中，作为备选聚类中心网格点。同时定义空集合 P，用于存储被选择网格点，首先将集合 B 中密度最大的网格点放入集合 P 中。

为了排除高密度网格点周围网格的影响，若分布在该网格周围的点较为密集时，将该网格周围一层网格从集合 B 中删除，否则向外扩展两层网格进行删除，数据集数量稀少时不需要此步骤操作。

对于集合 B 中剩余网格点，结合 K-means++思想，依次计算集合 B 中排行前 $2 \cdot k$ 个网格点，计算其与集合 P 中网格点的欧式距离，将距离最大的网格点放入集合 P 中。重复以上步骤直至集合 P 中含有 k 个网格点。最后将这 k 个网格中的中心点 $z = \{z_1, z_2, \ldots, z_k\}$，即为所求的初始聚类中心。

3.3 实验结果与分析

3.3.1 实验环境和数据集

实验所用计算机环境如下：Intel® Core i5-8250U，CPU 1.60Ghz， Windows 10 x64，编程软件是 64 位的 PyCharm R2019b。实验所用数据集是从分子生物学公共数据库 NCBI 信息库（https://www.ncbi.nlm.nih.gov/）中下载的红树林底层沉积物的环境样本数据集。数据集为两个包含不同样本数量的数据集，数据集 1 含有 254 条 16S rRNA 基因序列，数据集 2 含有 2212 条 16S rRNA 基因序列。

3.3.2 数据预处理

采集样品经过一系列生物学实验操作，实验下机后需要进行数据过滤，将过滤后剩于的高质量的干净序列再用于后期数据挖掘分析。数据挖掘前要对原数据进行预处理，数据预处理原理主要是将原始序列进行拆分，拆分后通过序列之间的可重叠关系将序列进行拼接，数据预处理环节结束。将拼接后的序列用于聚类成 OTU，并与数据库中已知序列信息进行比对，挖掘物种信息。根据聚类得到的 OTU 结果以及比对注释的物种信息可以对环境样品中的微生物进行多类型的分析，例如物种复杂度、组间物种差异以及关联分析与模型预测等。

从环境样品实验到结果分析的过程流程图如图 3.2 所示，本章主要完成该流程中数

据处理的步骤。

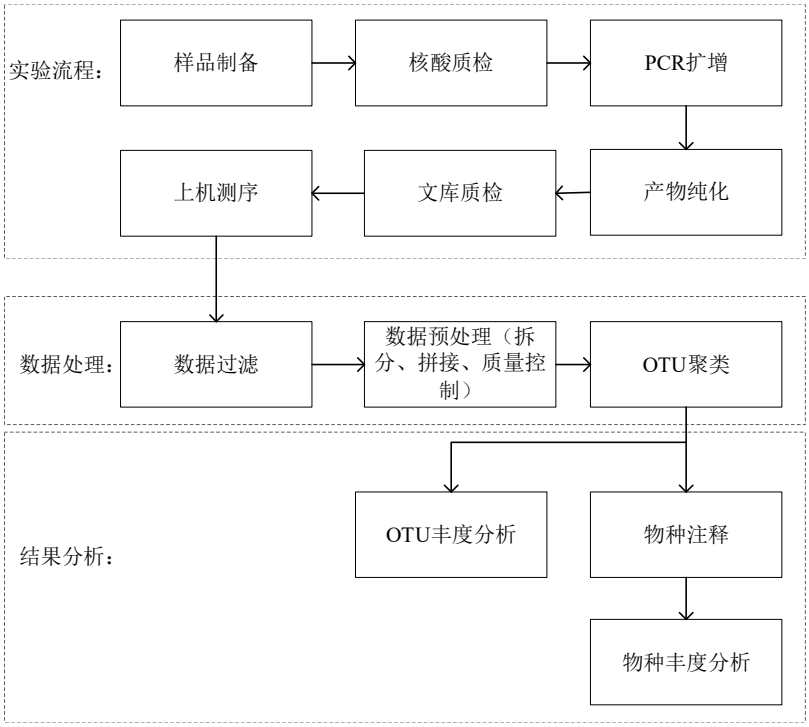

图 3.2 16S rRNA 处理流程图

数据集原数据未经预处理前是包含很多字符和序列的杂乱数据，如图3.3所示。序列数据预处理主要包括数据拆分、数据拼接、质量控制三个步骤。数据拆分将原数据的序列、字符拆分开来，保留序列信息，下一步通过序列间的重叠区将序列进行拼接，得到只包含 AGCT 四种脱氧核糖核苷酸的序列数据，如图 3.4 所示。最后将拼接好的序列进行质量控制，得到用于挖掘信息的可用 16S rRNA 序列数据。

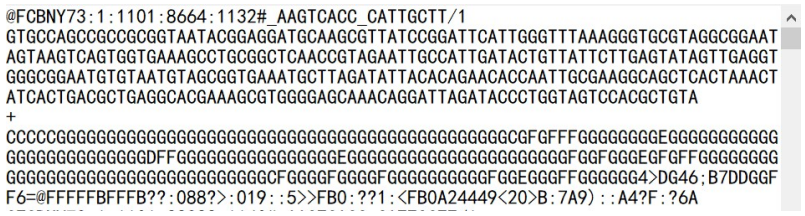

图 3.3 序列原数据形式图

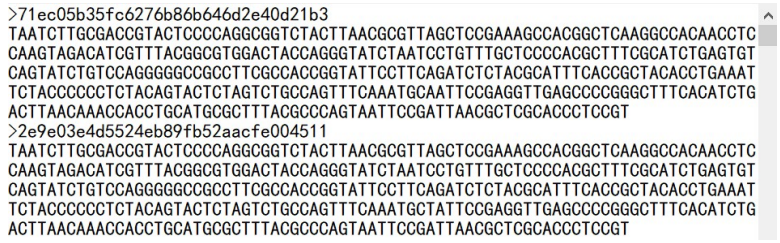

图 3.4 预处理后数据形式图

数据预处理后得到仅包含 A、G、C、T 的序列字符串，每条序列长度均不低于370bp，对于如此高的维度的数据，选择使用 PCA 主成分分析法对 16S rRNA 序列进行特征值提取、数据降维。特征值提取、数据降维的主要步骤如下，将降维后的数据用于 K-means 聚类算法分析。

（1）对序列数据进行顺序编码，将编码后的数据按列组成 n 行 m 列的矩阵 X。

（2）将矩阵 X 按行进行零均值化操作，即每行中各个元素依次减去该行的平均值。

（3）计算协方差矩阵并求出该矩阵的特征值及特征值相对应的特征向量。

（4）按照特征值从大到小将其对应的特征向量依次排列为矩阵形式，则该矩阵的前 k 行组成子矩阵 P。

（5）矩阵 P 与矩阵 X 的乘积即为所求特征值提取后维度降为 k 的易于处理的数据。

3.3.3 算法评价指标

本文使用可视化误差平方和(Sum of Square due to Error，SSE)，通过观察不同的 k 值对误差平方和的影响，以此确定较为合适的 k 值。SSE 评价指标表示当前的迭代所确定聚类簇的中心点到其所在簇中所有点的位置的距离总和，数学表达式为公式（3.9）所示。

$$SSE=\sum_{i=1}^{n}\sum_{j=1}^{k}\left\|x_i-z_j\right\|_2^2 \qquad \text{公式（3.9）}$$

上式中，z_j 为第 j 个聚类集群的中心点。作为评估当前聚类效果的指标 SSE，使其值在迭代结束尽可能最小，当 SSE 的值急速降低出现肘型曲线时，此时拐点的 k 值为较优选择。

3.3.4 结果分析

本章使用两个样本数量不同的数据集进行实验，误差平方和随 k 值大小的变化曲线图如图 3.5 所示。由图 3.5(a)可知，对于数据集 1，SSE 的值在集群数量为 3 时明显减小即集群数量 k 值的较优选择为 3。由图 3.5(b)可知，在数据集 2 中 SSE 曲线图的肘处，k 值为 3，此时不需更多次的迭代，k 值较优选择为 3。

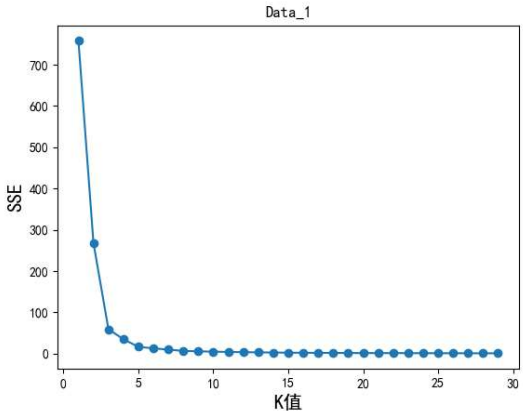

图 3.5(a) 数据集 1 SSE 曲线图

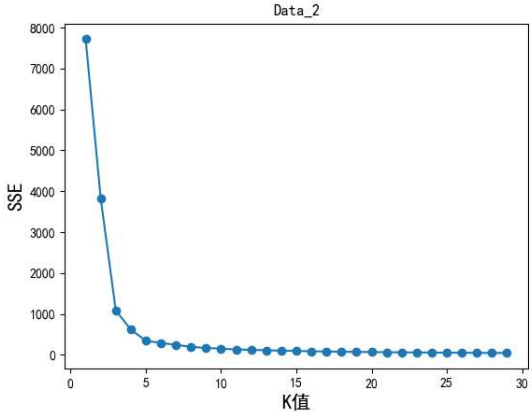

图 3.5(b) 数据集 2 SSE 曲线图

通过 SSE 变化曲线确定最优 k 值后，在两个不同数量的数据集上分类应用原始 K-means 算法和改进后的 GDD-Kmeans 算法，将初始聚类中心和聚类结束的中心点进行比较，如图 3.6、图 3.7 所示。

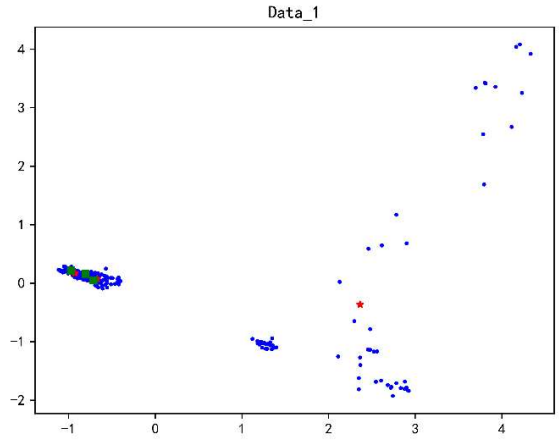

图 3.6(a) K-means 聚类中心对比

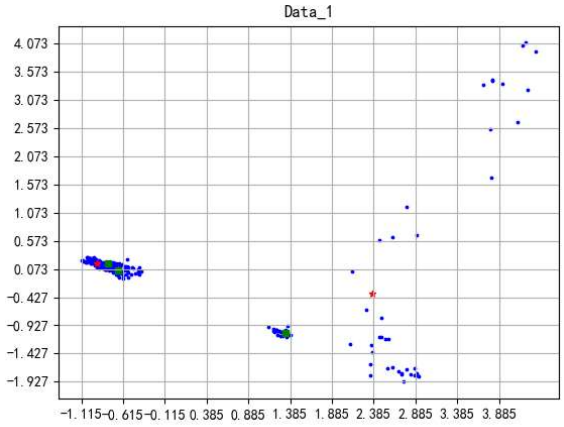

图 3.6(b) GDD-Kmeans 聚类中心对比

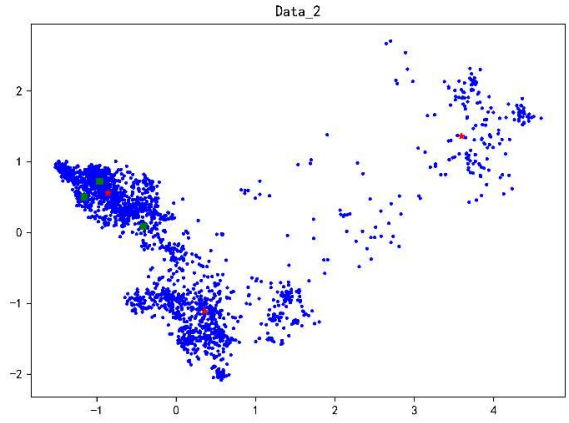

图 3.7(a) K-means 聚类中心对比

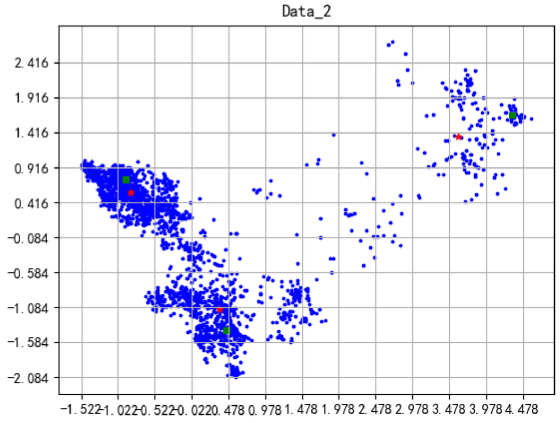

图 3.7(b) GDD-Kmeans 聚类中心对比

图 3.6(a)和(b)分别是传统 K-means 和 GDD-Kmeans 算法在同一个数据集上初始聚类中心与最终聚类中心的比较。绿色方块点为算法选择的初始聚类中心，红色星星点为聚类结束后最终的中心点。相对于传统 K-means 算法任意选定初始聚类中心，GDD-Kmeans 算法选择的初始聚类中心更接近最终结果的聚类中心。图 3.7(a)和(b)是在数据量较大的数据集上的中心点的对比，显然，GDD-Kmeans 算法的初始聚类中心与最终聚类中心距离更近，因此，算法运行迭代更快，有效证明了对初始聚类中心选取的优化。

为进一步证明本章中改进 K-means 聚类算法优化初始中心选择后比传统算法随机确定初始聚类中心的表现更优，在 16S rRNA 序列的两个不同数量的数据集上聚类实验，随机记录 10 次实验的算法迭代次数，可视化后如图 3.8 所示。

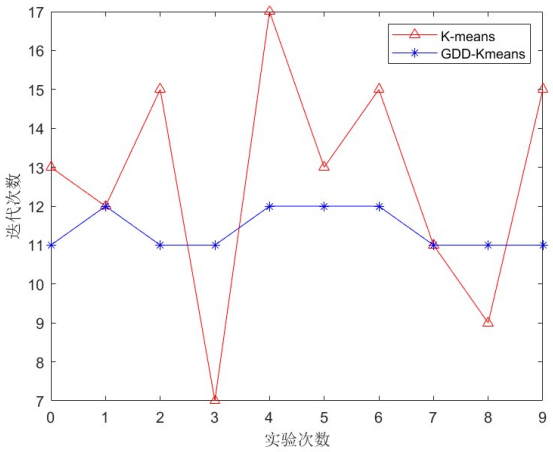

图 3.8(a)数据集 1 迭代次数对比

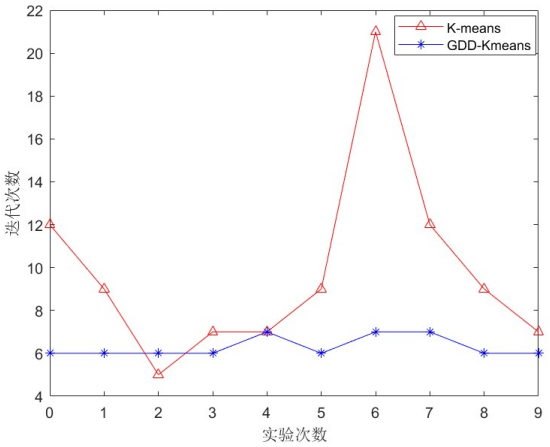

图 3.8(b)数据集 2 迭代次数对比

图 3.8(a)是在 16Sr RNA 序列数据样本数量较少的 254 条序列的数据集上聚类的迭代次数，传统 K-means 算法随机抉择初始中心可能会出现较少的迭代而完成聚类，但存在极不稳定的问题，针对更大型的数据集可能会因达到迭代次数而终止运行影响聚类结果。图 3.8(b)是在数量较多的 2212 条基因序列的数据集上的聚类，可以看出 GDD-Kmeans 算法在迭代次数较少的同时保持着稳定的运行迭代次数，说明优化后的算法针对大型数据集的聚类表现更为优异。对于数据集 1 和数据集 2 来说，随着实验次数的增加，传统 K-means 算法的迭代次数变化波动大，而 GDD-Kmeans 算法的迭代次数较为平缓、稳定，从而可以看出 GDD-Kmeans 算法更适用于数量较多的数据集的聚类。随着数据集中样本的增加，GDD-Kmeans 算法聚类效果更为突出，因此，GDD-Kmeans 算法更适用于数据集较多的数据聚类。

聚类实验完成后将聚类结果中代表性16S rRNA序列与SILVA数据库（https://www.arb-silva.de/）132版本进行比对，从而确定该环境样本中的菌群类别。数据集1中的16S rRNA序列对应的菌种均为古细菌，分类依次是Archaea->Euryarchaeota->Thermoplasma

ta->Marine Benthic Group D and DHVEG-1->uncultured archaeon；Archaea-> Euryarchaeota->Thermoplasmata->SG8-5；Archaea->Nanoarchaeaeota->Nanohaloarchaeia->Deep Sea Euryarchaeotic Group(DSEG)->uncultured archaeon。数据集2中菌种为细菌，其分类依次为： Bacteria->Actinobacteria->Acidimicrobiia->Microtrichales->Ilumatobacteraceae->Ilumatobacter； Bacteria->Bacteroidetes->Bacteroidia->Bacteroidales->SB-5； Bacteria->Bacteroidetes->Bacteroidia->Chitinophagales->Saprospiraceae。

该环境样本中菌群种类没有准确的分类性评价，因而不能明确聚类结果的准确与否。为了体现优化后 K-means 算法的聚类准确性，本章在公开的 UCI 数据（http://archive.ics.uci.edu/）中下载三个有明确分类标准的数据集上再次进行实验，数据集属性表如表 3.1 所示。

表 3.1 数据集属性表

数据集	属性	数据样本数量
Iris	4	150
UserKnowledgeModel	5	258
xclara	2	3000

在三个不同数据集上分别使用原始 K-means 算法和优化改进后的 GDD-Kmeans 算法分别进行聚类，并与数据样本的类别进行对照，计算误差率，误差率是记录多次实验的值并取平均值，证明了优化后算法聚类误差率有一定的降低，如表 3.2 所示。

表 3.2 算法误差率对比

数据集	K-means	GDD-Kmeans
Iris	0.2827	0.1893
UserKnowledgeModel	0.5209	0.4965
xclara	0.0080	0.0074

由表 3.2 可以看出 GDD-Kmeans 算法在三个不同数据集上均在一定程度上降低了传统 K-means 算法的误差率，经过计算得出，在三种数据集上的误差率分别降低了 0.0934、0.0244、0.0006。并且在数据样本分散的情况下，两种算法误差率都较高，表现都不佳。面对不同的数据集大小，优化后的 GDD-Kmeans 算法基本上都可以得到比传统 K-means 算法更好的聚类准确度。

3.4 本章小结

本章提出了基于网格密度距离优化的 K-means 算法，首先简单介绍了原始 K-means 算法的原理及步骤流程，然后详细介绍了本章提出的结合网格密度和 K-means++的思想策略优化后的算法步骤，并给出流程图，最后将传统 K-means 算法和优化后的 GDD-Kmeans 算法用于经过主成分分析法处理后的两个不同样本数量的 16S rRNA 序列的数据集上进行聚类实验，以 SSE 的值衡量聚类效果，同时比较两种算法运行的迭代次数，分析了算法优化的有效性。由于 K-means 算法聚类后需要进行的步骤是聚类簇中的物种丰度分析，该环境样本中菌群没有明确的准确度分类标准，本章选择三个有明确分类标准的数据集再次进行聚类算法的比较，实验证明改进后的 K-means 算法在精度上也有一定的提高。

第 4 章 基于优化鸽群的 ELM 的序列分类预测

从第三章可以看出，经过主成分分析进行特征值提取后的 16S rRNA 序列数据，利用优化后 K-means 算法对数据聚类，较好的控制了算法的稳定性。将聚类好的 16S rRNA 数据中代表序列与数据库比对即可得到所属分类，但是针对数据库中没有的序列，研究推断该菌落所属分类水平是生信技术的重要一步。神经网络可以在学习大量数据特征信息后对同类型的数据进行精准预测。基因序列的特征在于其核苷酸的排列结构，使用神经网络模型进行机器学习，通过机器学习来识别序列数据中的排列信息规律，在学习大量序列数据特征后，由于神经网络模型的泛化能力，可以实现准确预测序列的分类水平或者物种丰度等其他有用信息。本章提出一种基于优化鸽群的极限学习机分类预测算法（PIO-ELM），以四种误差指标来评价算法预测分类的效果。

4.1 优化鸽群算法

鸽群算法（Pigeon-inspired Optimization PIO），是一种仿生群优化算法，受鸽子的归巢行为启发仿生[68]。PIO 的特点是算法原理简单，算法参数无需大量调整，实现方法较容易，并且在计算方式和鲁棒性等方面具有显著优势。

影响鸽子归巢的主要原因大致可以分为三个原因，一是太阳，太阳的高度一定程度的影响着鸽子的巡航能力；二是地球磁场，磁场频率和的形状影响着鸽子的巡航能力；三是地形场景。鸽群算法原理[69]：鸽子飞行旅程中会用到各种不同的巡航工具。首先会通过地磁场掌握大体方向，而后利用地形场景修正当前方向，直至到达准确目的地。鸽子的磁场感知能力是通过鼻子经过三叉神经反馈给鸽脑．并在脑中构建地图，鸽子把太阳高度当作自己导航指南针。据此，PIO 算法基于地磁场和太阳构建出地图和指针算子模型，基于地标构建出地标算子模型[70]。简单来说就是仿生鸽群算法包括地图和指针算子、地标算子两个基础部分。当鸽子远离目的地时，使用地磁场来确定飞行方向，即地图和指南针算子；当鸽子接近目的地时，使用该区域的地标进行搜索，对路线进行评价并修正，即地标算子。

（1）地图和指针算子

在飞行的早期，鸽子会感受到磁场，主要是根据它们的磁感，在大脑中对地理进行映射，以对飞行方向进行调节。鸽群中每个个体 i 都被看做是可行的解决方案，进行目标函数优化，每个个体都有一个位置和相应的速度。

在鸽群飞行的早期阶段，鸽子感知磁场，主要根据它们的磁感，映射大脑构建地图并不断调整方向。 针对函数优化问题，每个可行的解决方案作为一个单独的个体 i，

都有一个位置和相应的速度，使用$X_i = [X_{i1}, X_i2, \ldots, X_{iD}]$表示第 i 只鸽子的位置，$V_i = [V_{i1}, V_{i2}, \ldots, V_{iD}]$表示第 i 只鸽子的速度，D 为维度。

在 D 维解空间中，鸽群个体的位置和速度每迭代一次就会更新一次。第 i 个个体的速度取决于其上一代的飞行速度和位置和此刻鸽群个体最佳位置。第 i 只鸽子的速度和位置迭代计算方法如下公式（4.1），公式（4.2）。

$$V_i(t) = V_i(t-1) * e^{-Rt} + rand * \left(X_g - X_i(t-1)\right) \qquad \text{公式（4.1）}$$

$$X_i(t) = X_i(t-1) + V_i(t) \qquad \text{公式（4.2）}$$

上式中， R 代表地图因子，t 为迭代代数，$rand$ 是 0 到 1 范围内的随机数。第 i 只鸽子的位置取决于前一次迭代的位置和此刻速度两个因素。地图用来确保鸽群整体的飞行，通过比较可以找到鸽子的全局最佳位置 X_g。个体根据公式（4.1）向处在最佳位置的个体来进行飞行方向的调整，根据公式（4.2）调整位置，达到迭代次数后停止，转入地表算子进行下一步运算。

（2）地标算子

基于使用地标进行导航的鸽子构建地标模型。通过地标导航比在地图上导航更接近目的地。 如果鸽子不熟悉其当前地理位置或地标，将由熟悉地标的相近鸽子引导巡航。当鸽子发现熟悉的地理标志时，将自行抵至目的地。在地标算子模型中，每次迭代都会造成鸽群数量减少一半，鸽群数量使用 Np 表示，数学表达为公式（4.3）。当前鸽群即为适应度较优的一半，此时将鸽群中心位置 X_c 作为参考方向，中心的计算方法为公式（4.4），并假设鸽子以直线距离飞至目的地，鸽子根据公式（4.5）更新个体位置。

$$Np(t) = \frac{Np(t-1)}{2} \qquad \text{公式（4.3）}$$

$$X_c(t) = \frac{\sum X_i(t) * F\left(X_i(t)\right)}{Np * \sum F\left(X_i(t)\right)} \qquad \text{公式（4.4）}$$

$$X_i(t) = X_i(t-1) + rand * \left(X_c(t) - X_i(t-1)\right) \qquad \text{公式（4.5）}$$

其中，函数 $F(\cdot)$是评价个体质量，也就是解的质量，分两种情况，对于最小优化

问题函数 F 表达式为公式（4.6），对于最大优化问题，函数 F 表达式为公式（4.7）。

$$F\left(X_i(t)\right)=\frac{1}{fitness_{\min}\left(X_i(t)\right)+\varepsilon} \qquad \text{公式（4.6）}$$

$$F\left(X_i(t)\right)=fitness_{\max}\left(X_i(t)\right) \qquad \text{公式（4.7）}$$

此时，将部分优势个体的位置中心看成整体的参考方向，此时，鸽群个体速度惯性消失，使鸽群收敛速度加快。地标算子在迭代循环达到限制值时停止运算。

鸽群算法在地图和指针算子模型中，鸽群全都向处在最佳位置的个体来进行飞行方向的调整，鸽子很容易汇集到同一个区域，陷入局部最优；在地标算子模型运算过程中，每次迭代鸽群数量均进行减半，这样可能会导致种群多样性较差的问题。针对这一问题，选择遗传算法[71]进行过程优化。遗传算法随机生成群体，群体中每个个体作为可能解决方案，根据计算出的个体适应度值，选择其中一些父代个体。对这些父代使用交叉算子，生成新一代并使用变异算子，产生新的可能解决方案的种群。通过这种方式使鸽群跳出局部最优，扩大全局搜索的能力。

在地图和指针算子过程中，为了提升全局搜索能力，应用遗传算法的选择交叉机制，在鸽群每次更新位置和对应速度后，随机选择适应度较高的部分鸽子两两配对进行交叉操作，代替父代鸽子，子代的生成方式如公式（4.8），公式（4.9）所示。

$$X_i^{'}=rand*X_i^t+\left(1-rand\right)*X_j^t \qquad \text{公式（4.8）}$$

$$X_j^{'}=\left(1-rand\right)*X_i^t+rand*X_j^t \qquad \text{公式（4.9）}$$

上述公式中，X_i'和X_j'分别表示鸽群中随机选择的父代个体X_i^t和X_j^t交叉后生成的新个体。达到最大迭代次数后转入地标算子，进行局部深度寻优。在地标算子模型运算过程中，每次迭代鸽群数量都减半，导致出现种群多样性较差的情况，也可能会过早收敛，针对这一问题，计算并比较当前全局最优的适应度的值，将适应度较优的个体以一定的强度进行柯西变异[72]操作，表示为公式（4.10）。与高斯分布的密度函数相比，柯西密度函数图像在原点处有较低的峰值，且在两端有较长的分布，因此，柯西变异可以在搜索邻域空间时耗时较短，跳出局部最优概率较高。

$$X_g^*=X_g+\lambda*Cauchy(c) \qquad \text{公式（4.10）}$$

上述公式（4.10）中 X_g 为突变前鸽子，λ是控制柯西变异算子的变化强度，*Cauchy(c)*是柯西分布的随机变量。比较交叉变异之前的最优个体与交叉变异之后的个体的适应度，将最优位置更新为适应度更好的，直到达到迭代停止条件。

4.2 基于优化鸽群的 ELM 算法

极限学习机 ELM 不仅适用于回归拟合，而且也有较好的分类效果，在诸多领域均应用良好，但是 ELM 是前反馈神经网络，不能根据反向传播来调整权重阈值，可能会出现陷入局部最优的问题，导致效果差强人意，并且存在一种隐含层输出矩阵是非满列秩矩阵或者输出矩阵式病态矩阵的情况，此时，会造成极限学习机网络模型计算出现问题，因此，从此种角度考虑，结合鸽群优化算法，优化 ELM 极限学习机权重和偏置参数，避免非满列秩矩阵或病态矩阵的出现，保证 ELM 网络模型的正常运行。改进鸽群优化后的极限学习机算法执行步骤如下：

1．顺序编码 16S rRNA 序列数据，划分数据集为训练集和测试集。

2．初始化鸽群规模参数，地图和指针算子及地标算子最大迭代次数，鸽群中鸽子个体代表极限学习机网络模型中的权重和偏差。

3．将训练数据集输入极限学习机，适应度函数设置为均方根误差，计算每只鸽子个体的适应度。

4．更新鸽子个体的位置和对应速度，根据公式（4.1）向处在最佳位置的个体 X_g 来进行速度和方向的调整，根据公式（4.2）调整个体位置。

5．在鸽群中选择部分鸽子使用公式（4.8）、公式（4.9），进行交叉配对生成新个体，代替父代鸽子融入新种群，计算鸽群中新的适应度，更新全局最优 X_g 和对应的适应度 gbest，判断是否达到迭代次数阈值，若未达到设置条件转入步骤 4，否则进入步骤 6。

6．进入地标算子模型，根据公式（4.3）更新鸽群数量 Np，根据公式（4.4）计算中心位置 X_c，最后由公式（4.5）更新当前鸽群中个体位置。

7．根据当前鸽群适应度排序，对适应度最佳的鸽子根据公式（4.10）进行柯西分布变异算子操作，重新计算鸽群中个体适应度，更新全局最优 X_g 为适应度更优的个体，对应适应度为 gbest， 判断是否达到迭代次数阈值，若不满足转入步骤 6，否则输出最优解，进入步骤 8。

8．将步骤 7 中输出最优参数值，即极限学习机模型的权重，传入极限学习机模型对数据集进行训练并对 16S rRNA 序列测试集数据预测分类输出结果。

基于优化鸽群算法的极限学习机模型步骤流程图如图 4.1 所示。

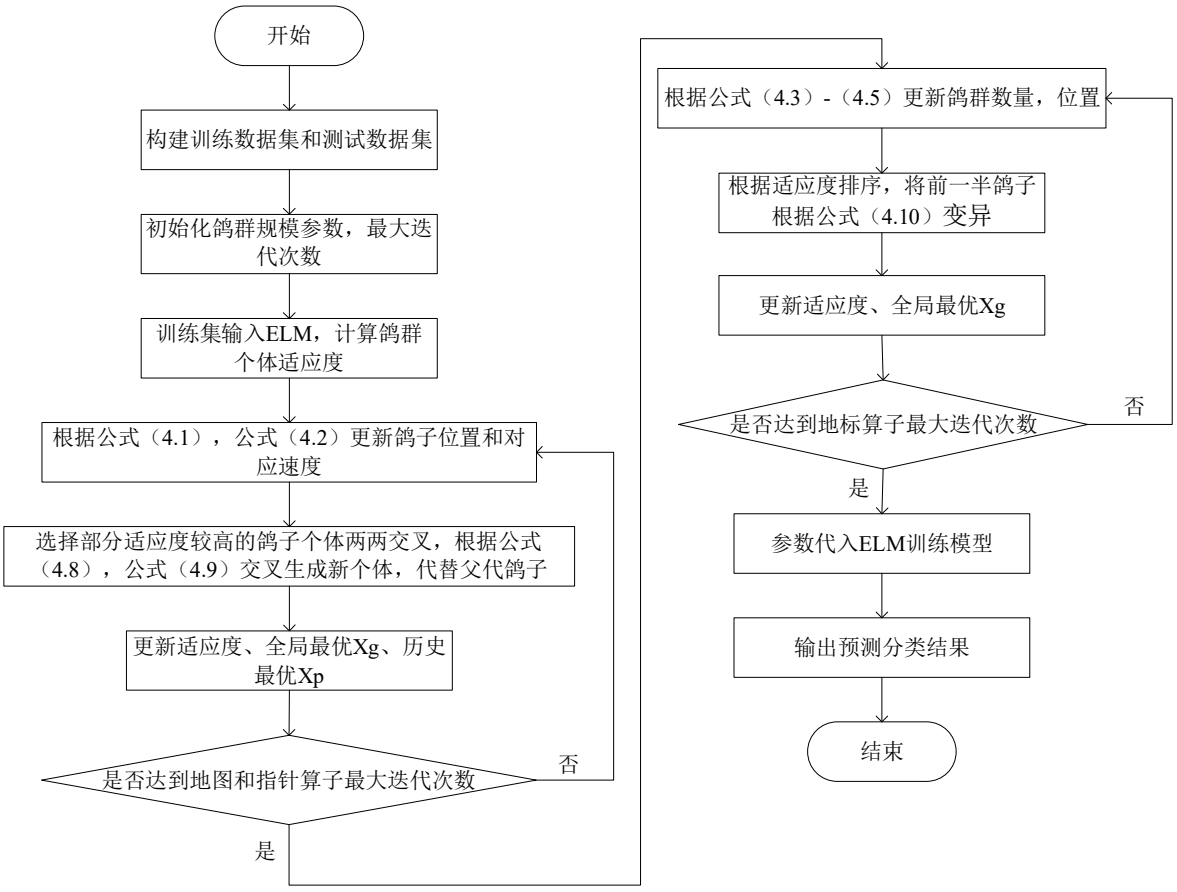

图 4.1 基于鸽群算法的 ELM 优化算法

4.3 实验结果和分析

4.3.1 实验环境及数据集

实验所用计算机环境如下：Intel® Core i5-8250U，CPU 1.60Ghz， Windows 10 x64，编程软件是 64 位的 PyCharm R2019b，matlab R2019b。本章实验数据集使用第三章聚类后与 SILVA132 版本比对后的序列数据，从中分离出两种数据集用于预测分类算法训练。数据集 1 为 321 条包含两种分类的 16S rRNA 基因序列数据集，其中 80%作为训练集，20%作为测试集；数据集 2 含有三种不同分类的 639 条 16S rRNA 序列，其中 520 条序列作为训练集，139 条序列作为算法测试数据集。

4.3.2 算法评价指标

算法评价指标有多种，在不同程度上有不同的衡量效果，因此，多方位考量指标评价算法优异性更加全面。本章同时选用四种指标进行比较。

均方误差（Mean Squared Error，MSE），是用来度量预测结果和真实结果之间偏

差的指标，数学表达式为公式（4.11），其中，y_i为真实值，$\hat{y}_i$为模型预测值，n 表示数据样本数量。

$$MSE = \frac{\sum_{i}^{n}\left(y_i - \hat{y}_i\right)^2}{n} \qquad \text{公式（4.11）}$$

均方根误差（Root Mean Squard Error，RMSE），与 MSE 实质性相同，但是 RMSE 相当于与数据在同一数量级，感知数据效果更好，数学表示形式见公式（4.12）。

$$RMSE = \sqrt{\frac{\sum_{i}^{n}\left(y_i - \hat{y}_i\right)^2}{n}} \qquad \text{公式（4.12）}$$

平均绝对误差（Mean Absolute Erro，MAE），数学表达为公式（4.13）。

$$MAE = \frac{\sum_{i}^{n}\left|\left(y_i - \hat{y}_i\right)\right|}{n} \qquad \text{公式（4.13）}$$

平均绝对百分比误差（Mean Absolute Percentage Error，MAPE），归一化处理了所有数据样本点的误差值，所以这种评价指标更具鲁棒性，如公式（4.14）所示。

$$MAPE = \frac{100\%}{n} \cdot \sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \qquad \text{公式（4.14）}$$

4.3.3 结果分析

对于 BP 神经网络的隐含层数目，通常可以根据公式（4.15）来选择合适大小，但是此经验公式并不适用于极限学习机，式中 n 代表输入层神经元节点数量，m 为输出层神经元节点数量，b 为正整数，一般情况下范围设置在 1 到 10 之间。

$$hn = \sqrt{(n+m)} + b \qquad \text{公式（4.15）}$$

由于极限学习机没有参照公式选择合适的隐含层节点数目，同时隐含层单元节点

的数量也是极限学习机网络模型的关键，本章以均方误差 MSE 作为评价指标，来确定极限学习机隐含层节点的最佳数目。

对于数据集 1，设置极限学习机迭代次数为 10 至 20，根据训练仿真的均方误差的值可以看出随着隐含层节点的增加，均方误差开始慢慢降低，而后又慢慢增加，出现起伏波动，比较均方误差的值，确定的隐含层节点数目为 17，如表 4.1 所示。

表 4.1 隐含层节点数量与 MSE

隐含层节点数目	MSE
10	0.164980
11	0.115870
12	0.084276
13	0.070874
14	0.180380
15	0.110830
16	0.133100
17	0.052603
18	0.334000
19	0.081621
20	0.064756

通过对训练集仿真确定最佳隐含层节点后，利用优化后的鸽群算法寻找极限学习机的最优权值阈值，图 4.2 是原始鸽群算法和基于遗传算法中交叉变异机制优化后鸽群算法的寻优曲线图，记录的是随着迭代次数的增加，适应度函数值的变化曲线，从图中可以看出优化后的鸽群算法均方误差有所降低，即适应度函数有所提升，证明了基于交叉变异机制优化的有效性。

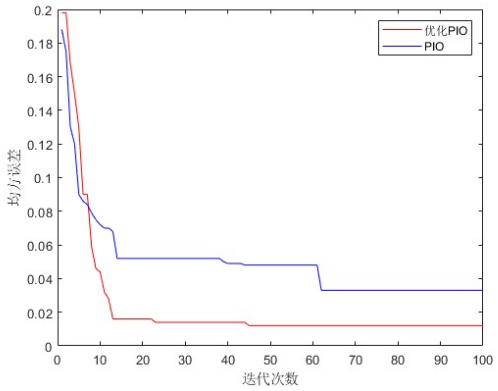

图 4.2 PIO 与优化 PIO 均方误差对比

将原始极限学习机网络模型与基于优化鸽群改进后的极限学习机模型的隐含层节点均设置为 17，BP 神经网络模型隐含层单元节点根据经验公式（4.15）设置。把构建好的训练集和测试集分别输入三种模型，得到模型的预测分类值，并与真实值对比，如图 4.3(a)和图 4.3(b)所示。

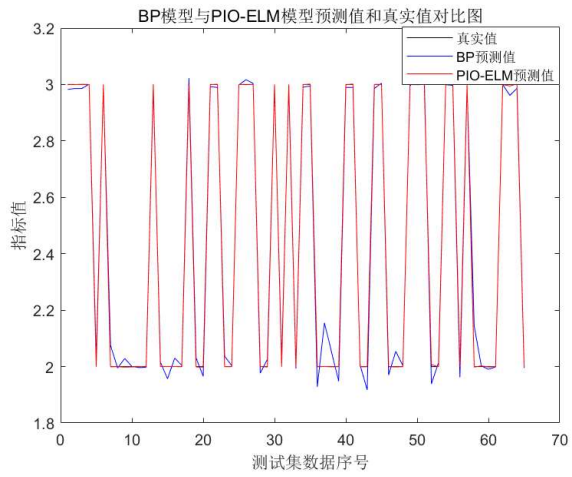

图 4.3(a) BP 模型与 PIO-ELM 模型结果对比

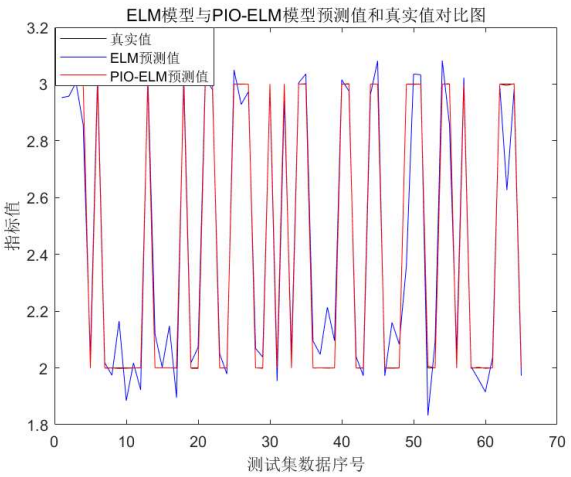

图 4.3(b)ELM 模型与 PIO-ELM 模型结果对比

从图 4.3 中，可以看出与 BP 神经网络模型、原始 ELM 模型预测值均有多个波动峰值出现，相比之下，本章提出的基于优化鸽群改进的极限学习机预测分类算法与真实值的拟合程度更高，说明 PIO-ELM 的预测精度较高。

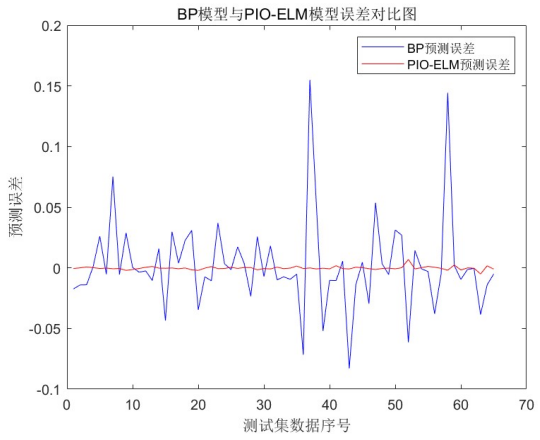

图 4.4(a)BP 模型与 PIO-ELM 模型误差对比

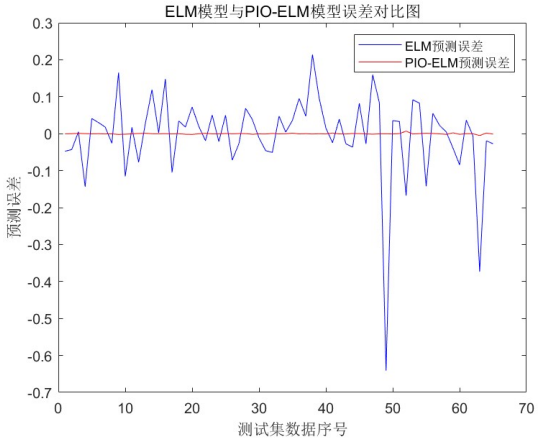

图 4.4(b)ELM 模型与 PIO-ELM 模型误差对比

为了更详细地对比模型预测精度，将三种模型预测误差对比可视化，如图 4.4 所示。从图中可以清晰明了地看出本章提出地 PIO-ELM 网络模型在数据集 1 上的预测误差较小，仅出现微弱的波动情况。BP 神经网络预测模型误差波动在-0.1 到 0.2 之间，原始 ELM 网络模型预测误差范围在-0.7 到 0.3 之间，两者相比，具有反向传播调整权

重参数的 BP 神经网络表现出了其优势。

针对数据集 1 的仿真实验，计算 BP 神经网络模型、原始 ELM 网络模型和 PIO-ELM 网络模型在 MAE、MSE、RMSE、MAPE 四种误差评价指标上的表现，如表 4.2 所示。

表 4.2 模型预测误差对比

模型	MAE	MSE	RMSE	MAPE
BP	0.016113	0.0099478	0.099739	1.5734%
ELM	0.078047	0.0106270	0.103090	3.2935 %
PIO-ELM	4.62211e-03	6.6434e-07	8.1507e-03	0.1886 %

根据表 4.2 中数据计算得到，PIO-ELM 模型相较于 BP 神经网络模型平均绝对误差、均方误差、均方根误差、平均绝对百分比误差分别降低了 0.0115、0.00995、0.0916、1.3848%。与原始 ELM 网络模型相比，四种指标分别降低了 0.0734、0.0106、0.00949、3.1049%。

对于只有两个分类的序列数据，PIO-ELM 网络模型取得了较好的预测拟合效果，在此基础上，使用含有更多分类的数据集 2，同样设置 80%的数据为训练集，剩余的 20%数据为测试集。首先确定隐含层节点数目，设置迭代范围为 20 到 30，同样比较训练集的均方误差，结果如表 4.3 所示。

表 4.3 隐含层节点数量与 MSE

隐含层节点数目	MSE
20	0.26899
21	0.20326
22	0.20208
23	0.16796
24	0.17026
25	0.18172
26	0.18466
27	0.22349
28	0.19458
29	0.15880
30	0.19167

比较表 4.3 中 MSE 值的大小可知，隐含层节点数目为 29 时，均方误差最小，因

此，在数据集 2 上的 ELM 极限学习机的隐含层单元节点设置为 29。

仿真实验构建训练数据集与测试数据集，分别使用 BP 神经网络算法、ELM 算法和基于改进 PIO 的 ELM 算法分别学习多分类数据集 2 的训练集数据特征，并对测试集进行预测分类，三种模型的实验对比结果如图 4.5(a)，图 4.5(b)所示。

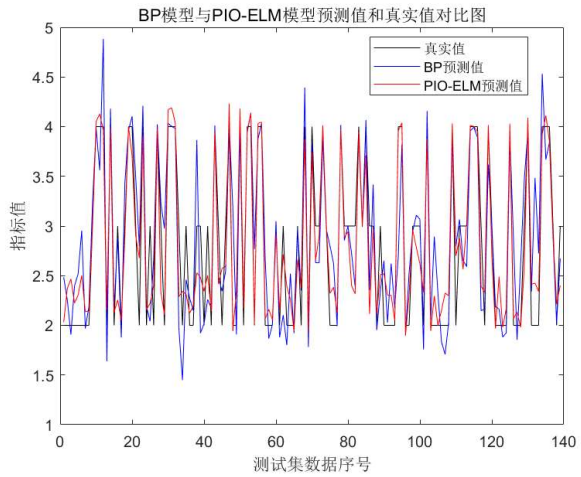

图 4.5(a) BP 模型与 PIO-ELM 模型结果对比

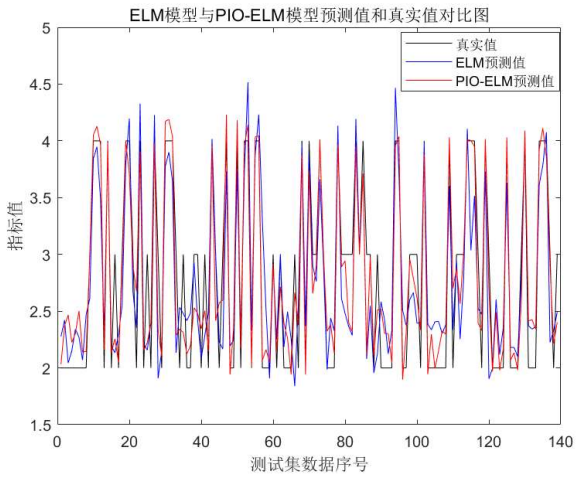

图 4.5(b) ELM 模型与 PIO-ELM 模型结果对比

图 4.5(a)(b)分别给出了网络模型训练完成后，在含有 139 条序列的测试数据集上 BP 神经网络模型与 PIO-ELM 网络模型预测结果与真实值的对比，以及原始 ELM 网络模型与优化后的 PIO-ELM 模型预测结果与真实值的对比，通过图中折线的峰值可以看出，与 BP 网络模型、ELM 网络模型相比，本章提出的 PIO-ELM 模型的预测精度都更接近目标值，峰值波动均低于 BP 神经网络模型和原始的极限学习机网络模型，说明了模型优化的有效性。

与二分类问题的数据集 1 的预测结果相比，各模型的拟合精度都降低很多，PIO-ELM 模型误差也有较为明显的提升。为了进一步证明本章提出算法的预测性能，选取基于蚁群算法的极限学习机（ACO-ELM）和基于麻雀算法的极限学习机（SSA-ELM）进行对比，结果如图 4.6(a)和图 4.6(b)所示。由指标值的坐标可以看出经过仿生算法优化后的 ELM 模型均提高了一定的精度，PIO-ELM 模型表现更为稳定。根据预测值与真实值的对比，只能模糊看出各模型预测分类效果，下面将预测值与真实值的误差可视化来更清晰的观察算法效果，如图 4.7 所示。

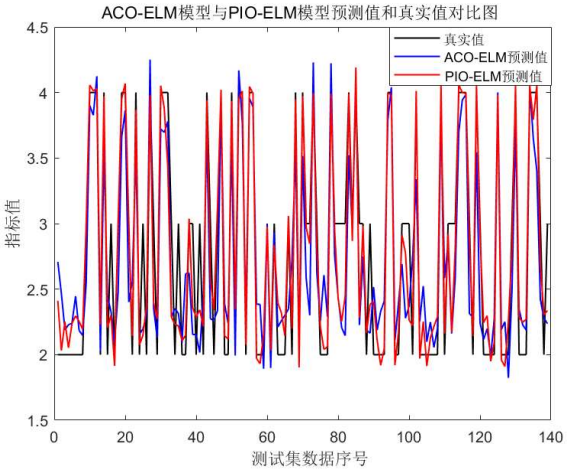

图 4.6(a) AG-ELM 与 PIO-ELM 结果对比

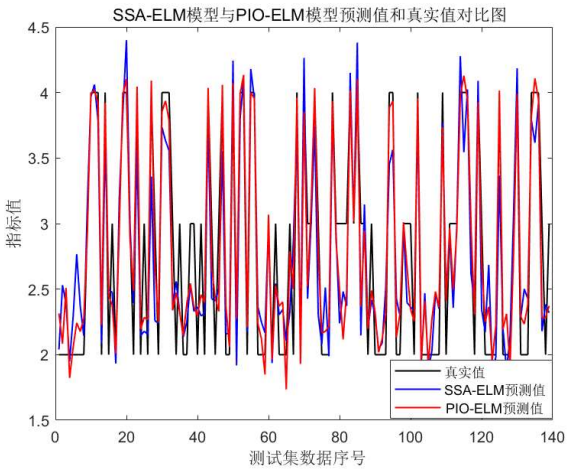

图 4.6(b) SSA-ELM 与 PIO-ELM 结果对比

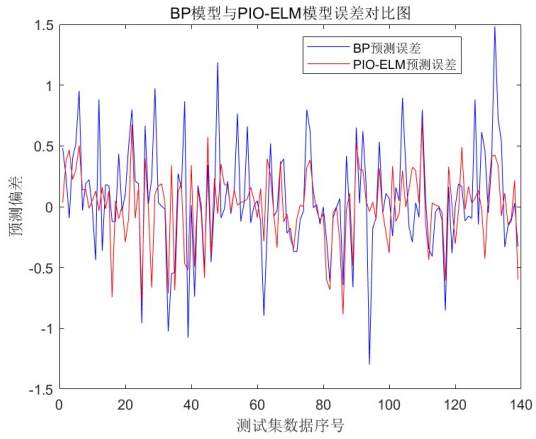

图 4.7(a) BP 模型与 PIO-ELM 模型误差对比

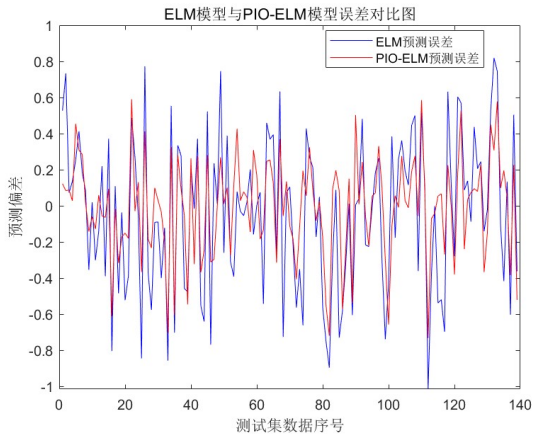

图 4.7(b) ELM 模型与 PIO-ELM 模型误差对比

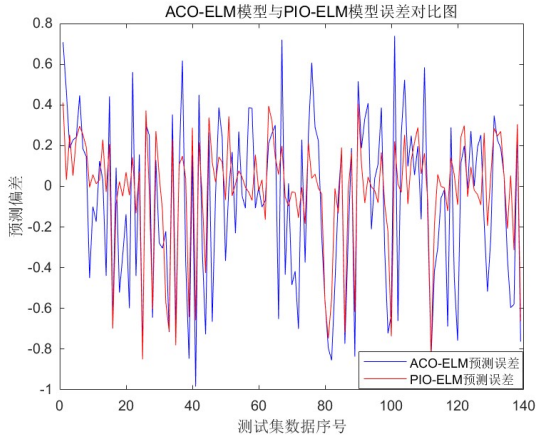

图 4.7(c) ACO-ELM 与 PIO-ELM 模型误差对比

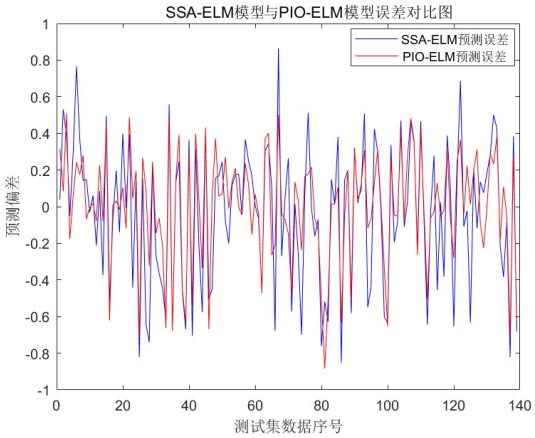

图 4.7(d) SSA-ELM 与 PIO-ELM 模型误差对比

通过图 4.7(a)给出的 BP 模型与 PIO-ELM 模型误差对比，PIO-ELM 模型预测误差明显低于 BP 模型，误差波动较为平稳，BP 模型的误差在-1.5 到 1.5 之间，而 PIO-ELM 模型的误差大致在-1 到 1 之间。由图 4.7(b)原始 ELM 网络模型与鸽群算法优化后的

ELM网络模型的误差对比，可以看出ELM模型将结果误差值缩减到-1到1之间，PIO-ELM模型误差在-0.8到0.6之间，具有更小的误差和预测稳定性。

根据图4.7(c)，图4.7(d)的预测偏差指标看出，经过仿生优化算法改进后的ELM模型，预测误差均得到了降低，PIO-ELM 模型曲线误差更小，更稳定。实验结果还分别统计了 BP 神经网络模型、ELM 模型、ACO-ELM 模型、SSA-ELM 模型与本章提出的 PIO-ELM 模型在平均绝对误差、均方误差、均方根误差、平均绝对百分比误差四种评价指标上的值，统计结果如表 4.4 所示。

表 4.4 模型预测误差对比

模型	MAE	MSE	RMSE	MAPE
BP	0.39655	0.207510	0.45553	13.2720%
ELM	0.33716	0.174060	0.41721	13.2265 %
ACO-ELM	0.22939	0.086330	0.29382	9.3681%
SSA-ELM	0.23811	0.095838	0.30958	9.7092%
PIO-ELM	0.19477	0.080468	0.28367	7.7497 %

从表4.4中的数据可以看出两种原始的神经网络模型BP和ELM的表现不相上下，但是BP神经网络需要设置的参数有很多，并且需要不断反向传播去调节参数的值，因此训练时间大大高于 ELM 网络模型，此时 ELM 网络只需一次性设置模型权重阈值和偏差并且算法运行时间大大减少的优势就体现了出来。将蚁群算法 ACO、麻雀算法 SSA 应用于 ELM 模型参数寻优也得到了一定的性能优化，本章提出的 PIO-ELM 模型在四种评价指标上表现更优，经过计算可以得出，PIO-ELM 模型预测比 BP 网络模型平均绝对误差、均方误差、均方根误差、平均绝对百分比误差分别降低了 0.19734、0.133289、0.18309、5.1081%；比 ELM 模型预测结果平均绝对误差、均方误差、均方根误差、平均绝对百分比误差分别降低了 0.138、0.099、0.145、5.063%；比 ACO-ELM 模型的平均绝对误差、均方误差、均方根误差、平均绝对百分比误差分别降低了0.00872、0.005862、0.01015、1.6184%；比SSA-ELM模型在平均绝对误差、均方误差、均方根误差、平均绝对百分比误差上分别降低了0.04334、0.01537、0.02591、1.9595%。这些数据表明了基于优化鸽群的极限学习机算法具有较好的预测准确性。

通过对二分类问题的数据集 1 和多分类问题的数据集 2 分别进行训练学习输入的序列数据与输出的类别之间的映射关系，多种神经网络模型在二分类的数据集上均表现出较好的预测结果。当数据集中的分类类别增大后，神经网络模型的预测精度都有一定的降低，但是相比与BP神经网络模型、原始极限学习机模型、基于蚁群算法的极限学习机和基于麻雀算法的极限学习机，本章提出的基于改进鸽群算法优化后的极限

学习机预测 16S rRNA 序列的分类有较好预测精度。

4.4 本章小结

本章提出了基于优化鸽群的极限学习机优化算法，通过鸽群仿生算法优化极限学习机的模型参数，指针算子提高搜索全局性，地标算子提高局部搜索的能力。由于鸽群算法也存在其局限性，因此，本章对地图和指南针算子使用遗传算法中的交叉机制，提高鸽群算法寻优的全局性，避免陷入局部最优；对地标算子使用遗传算法中的变异机制，增加种群多样性，同时避免出现过早收敛。为了比较本章提出的神经网络预测模型与其他神经网络预测模型的性能，使用两种具有不同分类类别的 16S rRNA 序列数据集，分别采用 BP 神经网络的预测模型、ELM 网络预测模型、基于蚁群算法的 ELM 网络模型、基于麻雀算法的ELM网络模型和本章提出的基于改进鸽群算法的PIO-ELM网络预测模型分别进行 16S rRNA 序列训练数据集特征学习并对测试集预测输出结果。实验结果表明在二分类问题上，神经网络模型均有较好表现，在多种分类类别的数据集上，虽然五种模型的预测结果与真实值都有一定差距，但相比其他四种网络模型，本章提出的 PIO-ELM 模型预测出的 16S rRNA 序列分类各项评价参数指标上表现更优，拟合效果更好，预测精度更高。

第 5 章 总结与展望

5.1 总结

微生物在调节地球生态环境中有着不可磨灭的作用，比如分泌激素降解垃圾。随着人类社会活动的不断拓展，微生物群落在人类生物化学方面同样发挥着关键作用，很大程度上影响着生物于非生物环境间的关系。随着技术的发展，实现了从环境中直接扩增和测序 16S rRNA 基因，而不再限制于实验室中培养分离。面对大量的 16S rRNA序列，下游的数据处理步骤变的十分重要，序列数据中包含物种多样性，物种丰度等重要信息，对推进人类合理利用微生物，与微生物和谐共存具有非常重要的意义。

16S rRNA 序列进行数据分析是生物信息学上一项严峻的挑战，有关 16S rRNA 序列聚类得到 OTU 的方法层出不穷，不同的聚类分析方法得到的信息侧重不同，因此，有必要发展研究不同的聚类方法，供宏基因组学的研究人员们选择。目前，研究者们更多的考虑聚类算法的运行时间以及效率。聚类完成后的分析信息用于神经网络的模型训练，根据大量学习，预测结果将会逐渐趋向于真实结果，对推测未知序列的分类类别意义重大。对此，本文主要的工作如下：

（1）本文基于网格的聚类算法和基于密度的聚类算法的特点，结合 K-means++算法思想，提出一种基于网格密度距离的 K-means 优化算法。旨在解决 K-means 算法随机选取初始聚类中心，从而导致聚类不稳定，出现偏差的问题。针对数据集高维度的问题，首先使用主成分分析法对原始 16S rRNA 数据进心特征值提取，降低数据维度，再利用 K-means 算法聚类。对于大型数据集，随机的初始聚类中心使得算法迭代产生大量时间成本，有时可能会出现超出迭代次数从而影响聚类结果的情况。对此，本文基于网格，把数据样本放在网格中，用网格代替数据作为处理单元，使得数据集在逻辑结构上缩小，然后在密集的网格中选取距离较远的 k 个网格，初始聚类中心确定在这 k 个网格中，以此来优化初始聚类中心的选定。经过实验证明了优化后的 K-means 聚类算法具有更少的迭代次数和迭代稳定性，一定程度上提高了算法效率。

（2）本文提出基于优化鸽群算法的 ELM 极限学习机的序列预测方法。考虑到极限学习机采用的是单隐含层，并且不会进行反向传播，在训练阶段不会更新或优化网络中的权重，导致在不同训练阶段的性能不稳定，所以选取鸽群算法对 ELM 网络学习中的参数进行优化。鸽群算法存在的问题是容易陷入局部最优，对此，本文引入遗传算法中的交叉变异机制分别对鸽群算法中的地图和指南针算子模型及地表算子模型进行优化，提高鸽群多样性以及鸽群搜索全局能力。极限学习机网络模型使用优化后的

鸽群算法对模型参数寻优，以此提升对 16S rRNA 序列的特征值学习能力，提高预测分类精度。通过实验，将 PIO-ELM 网络模型与 BP 神经网络模型、原始极限学习机网络模型、基于蚁群算法的极限学习机模型、基于麻雀算法的极限学习机模型在 16S rRNA 序列数据集上进行训练学习并预测分类，结果表明，基于优化鸽群的极限学习机算法预测结果更接近真实值，并且在平均绝对误差、均方误差、均方根误差、平均绝对百分比误差四项指标上都有更好的表现。

5.2 展望

随着宏基因组学和测序技术的进一步发展，今后的基因序列数据信息挖掘将愈加炙热，通过在分子生物学公共数据库中搜索比对大量的 16S rRNA 数据，再把单个序列分配其所属的分类水平增加大量的时间成本。所以，将大量 16S rRNA 序列数据通过聚类算法分类为集群，再选取每个集群簇中的代表序列与公共数据库进行比对，确定所属分类水平，节省了大量处理序列数据的时间成本。本文通过优化 K-means 算法，提高了聚类算法运行的稳定性，并基于优化鸽群算法提高了 ELM 极限学习机的预测精度，但是仍存在许多有待改进优化的问题，今后将把研究重点放在以下几点上：

1．K-means 算法必须要进行初始化参数选择，没有先验知识来确定集群的数量，使得 K-means 的性能得到掣肘，针对这一问题，加强研究工作。

2．增加聚类算法精度，考虑如何利用序列特征使聚类精确到 16S rRNA 的分类水平上，更利于后续物种丰度，物种多样性的分析。

3．训练 PIO-ELM 网络模型使用的数据量较少，泛化能力较弱，一定程度上影响了预测的结果，并且机器学习采用顺序编码，特征值提取精准度不够，也一定程度影响了分类预测的精度。

参考文献

[1] 朱真, 朱嗣博, 张铁军, 等. 宏基因组学与人类健康关系研究进展[J]. 中国公共卫生, 2019, 35(01): 122-124.

[2] 孙欣, 高莹, 杨云锋. 环境微生物的宏基因组学研究新进展[J]. 生物多样性, 2013(4): 393-400.

[3] Behjati S, Tarpey P S. What is next generation sequencing?[J]. Archives of Disease in Childhood-Education and Practice, 2013, 98(6): 236-238.

[4] Johnson J S, Spakowicz D J, Hong B Y, et al. Evaluation of 16S rRNA gene sequencing for species and strain-level microbiome analysis[J]. Nature Communications, 2019, 10(1): 1-11.

[5] Janda J M, Abbott S L. 16S rRNA gene sequencing for bacterial identification in the diagnostic laboratory: pluses, perils, and pitfalls[J]. Journal of Clinical Microbiology, 2007, 45(9): 2761-2764.

[6] 邓飞龙. 微生物 16S rRNA 基因序列分类单元(OTUs)聚类算法的设计与实现[D]. 成都:四川农业大学, 2016.

[7] Sinaga K P, Yang M. Unsupervised K-means Clustering Algorithm[J]. IEEE Access, 2020, 8: 80716-80727.

[8] 黄晓辉, 王成, 熊李艳,等. 一种集成簇内和簇间距离的加权 K-means 聚类方法[J]. 计算机学报, 2019, 42(12): 2836-2848.

[9] 杨俊闯, 赵超. K-means 聚类算法研究综述[J]. 计算机工程与应用, 2019,55(23): 7-14+63.

[10] Alhawarat M, Hegazi M. Revisiting K-means and Topic Modeling, a Comparison Study to Cluster Arabic Documents [J]. IEEE Access, 2018, 6: 42740-42749.

[11] 叶骁. K-means 聚类算法在肿瘤基因变异识别中的应用[J]. 计算机应用与软件, 2019, 36(03): 287-290+333.

[12] Angell I L, Nilsen M, Carlsen K C L, et al. De novo species identification using 16S rRNA gene nanopore sequencing[J]. PeerJ, 2020, 8: e10029.

[13] Antoine G B, Cathy M R, Andrea R. Clustering transformed compositional data using K-means, with applications in gene expression and bicycle sharing system data[J]. Journal of Applied Statistics, 2019, 46(1): 47-65.

[14] 王侠林, 贺建峰. 基于 K-means 聚类的微生物群落结构研究[J]. 软件导刊, 2018, 17(01): 146-148+151.

[15] Zong B, Song Q, Min M R, et al. Deep autoencoding gaussian mixture model for unsupervised anomaly detection[C]. International Conference on Learning Representations. 2018.

[16] 曾俊. 基于划分的数据挖掘 K-means 聚类算法分析[J]. 现代电子技术, 2020, (03): 14-17.

[17] 黄松, 邱建林. 改进的遗传 K-means 算法及其应用[J]. 计算机工程与设计, 2020, 41(6): 1617-1623.

[18] Shi H, Xu M. A Data Classification Method Using Genetic Algorithm and K-means Algorithm with Optimizing Initial Cluster Center[C]. IEEE International Conference on Computer and Communication Engineering Technology (CCET), 2018, 224-228.

[19] Hossain M Z, Akhtar M N, Ahmad R B, et al. A dynamic K-means clustering for data mining[J]. Indonesian Journal of Electrical Engineering and Computer Science, 2019, 13(2): 521-526.

[20] Yang M S, Sinaga K P. A feature-reduction multi-view K-means clustering algorithm[J]. IEEE Access, 2019, 7: 114472-114486.

[21] Zhang G, Zhang C, Zhang H. Improved K-means algorithm based on density Canopy[J]. Knowledge-based Systems,2018,145: 289-297.

[22] Bai L, Liang J, Cao F. A multiple K-means clustering ensemble algorithm to find nonlinearly separable clusters[J]. Information Fusion, 2020, 61: 36-47.

[23] Fard M M, Thonet T, Gaussier E. Deep K-means: Jointly clustering with K-means and learning representations[J]. Pattern Recognition Letters, 2020, 138: 185-192.

[24] Wang S, Li M, Hu N, et al. K-means clustering with incomplete data[J]. IEEE Access, 2019, 7: 69162-69171.

[25] Zhao Z, Woloszynek S, Agbavor F, et al. Learning, visualizing and exploring 16S rRNA structure using an attention-based deep neural network[J]. PLoS Computational Biology, 2021, 17(9): e1009345.

[26] Wang D, Huang G B. Protein sequence classification using extreme learning machine[C]. Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005. IEEE, 2005, 3: 1406-1411.

[27] Rasheed Z, Rangwala H. Metagenomic taxonomic classification using extreme learning machines[J]. Journal of Bioinformatics and Computational Biology, 2012, 10(05): 1250015.

[28] Zheng Y, Chen B, Wang S, et al. Mixture correntropy-based kernel extreme learning machines[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 33(2): 811-825.

[29] 许敏, 胡丽丹. 径向基函数神经网络快速算法及其应用[J]. 统计与决策, 2021, 37(16): 52-56.

[30] 顾燕萍, 赵文杰, 吴占松. 最小二乘支持向量机的算法研究[J]. 清华大学学报: 自然科学版, 2010(7): 1063-1066,1071.

[31] Li Y, Tian X, Song M, et al. Multi-task proximal support vector machine[J]. Pattern Recognition, 2015, 48(10): 3249-3257.

[32] 唐延强, 李成海, 宋亚飞. 基于改进粒子群优化和极限学习机的网络安全态势预测[J]. 计算机应用, 2021, 41(03): 768-773.

[33] Liu Z F, Li L L, Tseng M L, et al. Prediction short-term photovoltaic power using improved chicken swarm optimizer-extreme learning machine model[J]. Journal of Cleaner Production, 2020, 248: 119272.

[34] Chen Y, Xie X, Zhang T, et al. A deep residual compensation extreme learning machine and applications[J]. Journal of Forecasting, 2020, 39(6): 986-999.

[35] Zhu C, Idemudia C U, Feng W. Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques[J]. Informatics in Medicine Unlocked, 2019, 17: 100179.

[36] Granato D, Santos J S, Escher G B, et al. Use of principal component analysis (PCA) and hierarchical cluster analysis (HCA) for multivariate association between bioactive compounds and functional properties in foods: A critical perspective[J]. Trends in Food Science & Technology, 2018, 72: 83-90.

[37] Xanthopoulos P, Pardalos P M, Trafalis T B. Linear discriminant analysis[M]. Robust data mining. Springer, New York, NY, 2013: 27-33.

[38] Tharwat A, Gaber T, Ibrahim A, et al. Linear discriminant analysis: A detailed tutorial[J]. AI Communications, 2017, 30(2): 169-190.

[39] Brown D, Japa A, Shi Y. A Fast Density-Grid Based Clustering Method[J]. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), 2019, 0048-0054.

[40] Kriegel H P, Kröger P, Sander J, et al. Density‐based clustering[J]. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2011, 1(3): 231-240.

[41] Khan K, Rehman S U, Aziz K, et al. DBSCAN: Past, present and future[C]. The fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014). IEEE, 2014: 232-238.

[42] Idrissi A, Rehioui H, Laghrissi A, et al. An improvement of DENCLUE algorithm for the data clustering[C]. 2015 5th International Conference on Information & Communication Technology and Accessibility (ICTA). IEEE, 2015: 1-6.

[43] 罗军锋, 洪丹丹. 一种基于密度和距离的 K-means 聚类算法[J]. 软件工程, 2020, 23(10): 23-25+4.

[44] Suman, Rani, Pinki. A Survey on STING and CLIQUE Grid Based Clustering Methods[J]. International Journal of Advanced Research in Computer Science, 2017, 8(5): 1510-1512.

[45] 蔡馥励. 基于网格的聚类算法研究[D]. 哈尔滨:哈尔滨工程大学, 2017.

[46] Cai W, Yang J, Yu Y, et al. PSO-ELM: A hybrid learning model for short-term traffic flow forecasting[J]. IEEE Access, 2020, 8: 6505-6514.

[47] Gao S, Wen Y. An improved artificial fish swarm algorithm and its application[C]. ACIS 17th International Conference on Computer and Information Science (ICIS). IEEE, 2018: 649-652.

[48] Hakli H, Kiran M S. An improved artificial bee colony algorithm for balancing local and global search behaviors in continuous optimization[J]. International Journal of Machine Learning and Cybernetics, 2020, 11(9): 2051-2076.

[49] Kumar V, Kumar D. A systematic review on firefly algorithm: past, present, and future[J]. Archives of Computational Methods in Engineering, 2021, 28(4): 3269-3291.

[50] Jayabarathi T, Raghunathan T, Gandomi A H. The bat algorithm, variants and some practical engineering applications: a review[J]. Nature-inspired Algorithms and Applied Optimization, 2018: 313-330.

[51] Wang J, Di Y, Rui X. Research and application of machine learning method based on swarm intelligence optimization[J]. Journal of Computational Methods in Sciences and Engineering, 2019, 19(S1): 179-187.

[52] Brezočnik L, Fister I, Podgorelec V. Swarm intelligence algorithms for feature selection: a review[J]. Applied Sciences, 2018, 8(9): 1521.

[53] 张峰峰, 张欣, 陈龙,等. 采用改进遗传算法优化神经网络的双目相机标定[J]. 中国机械工程, 2021, 32(12): 1423-1431.

[54] Mirjalili S, Song Dong J, Sadiq A S, et al. Genetic algorithm: Theory, literature review, and application in image reconstruction[J]. Nature-inspired Optimizers, 2020: 69-85.

[55] Ning J, Zhang Q, Zhang C, et al. A best-path-updating information-guided ant colony optimization algorithm[J]. Information Sciences, 2018, 433: 142-162.

[56] Ning J, Zhang C, Sun P, et al. Comparative study of ant colony algorithms for multi-objective optimization[J]. Information, 2018, 10(1): 11.

[57] Xue J, Shen B. A novel swarm intelligence optimization approach: sparrow search algorithm[J]. Systems Science & Control Engineering, 2020, 8(1): 22-34.

[58] Yan P, Shang S, Zhang C, et al. Research on the Processing of Coal Mine Water Source Data by Optimizing BP Neural Network Algorithm With Sparrow Search Algorithm[J]. IEEE Access, 2021, 9: 108718-108730.

[59] Khan A, Sohail A, Zahoora U, et al. A survey of the recent architectures of deep convolutional neural networks[J]. Artificial Intelligence Review, 2020, 53(8): 5455-5516.

[60] Sherstinsky A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network[J]. Physica D: Nonlinear Phenomena, 2020, 404: 132306.

[61] Zhang L, Wang F, Sun T, et al. A constrained optimization method based on BP neural network[J]. Neural Computing and Applications, 2018, 29(2): 413-421.

[62] Li J, Cheng J, Shi J, et al. Brief introduction of back propagation (BP) neural network algorithm and its improvement[M]. Advances in computer science and information engineering. Springer, Berlin, Heidelberg, 2012: 553-558.

[63] 孟雯雯, 胡聪, 赵建平, 徐娟. 基于改进极限学习机的多维参数天线设计方法[J]. 通信技术, 2021, 54(04): 985-991.

[64] Ding S, Xu X, Nie R. Extreme learning machine and its applications[J]. Neural Computing and Applications, 2014, 25(3): 549-556.

[65] 张毅. 提升极限学习机的泛化方法及应用[D]. 无锡: 江南大学, 2021.

[66] 华婷婷. K-means 聚类算法研究[J]. 黄山学院学报, 2013, 15(05): 17-19.

[67] Yuan C, Yang H. Research on K-Value Selection Method of K-means Clustering Algorithm[J]. J, 2019, 2(2): 226-235.

[68] 郭瑞. 鸽群优化算法及其应用研究[D]. 南宁: 广西民族大学, 2017.

[69] 段海滨, 叶飞. 鸽群优化算法研究进展[J]. 北京工业大学学报, 2017, 43(01): 1-7.

[70] 周雨鹏. 基于鸽群算法的函数优化问题求解[D]. 长春: 东北师范大学, 2016.

[71] Katoch S, Chauhan S S, Kumar V. A review on genetic algorithm: past, present, and future[J]. Multimedia Tools and Applications, 2021, 80(5): 8091-8126.

[72] Liu F, Liu Y, Han F, et al. Synthesis of large unequally spaced planar arrays utilizing differential evolution with new encoding mechanism and Cauchy mutation[J]. IEEE Transactions on Antennas and Propagation, 2020, 68(6): 4406-4416.

个人简历、申请学位期间的研究成果及发表的学术论文

一、个人简历

张佳，女，河南濮阳人，1996 年 11 月出生。2017 年 7 月毕业于安阳工学院，获得工学学士学位。2019 年 9 月至今在桂林理工大学攻读软件工程学术型硕士学位，主要研究方向为数据挖掘。

二、攻读硕士学位期间的研究成果

发表论文

1. 基于网格密度距离的 K-means 优化算法 [J].桂林理工大学学报（已录用，第一作者)

参与科研项目

项目名：基于 IPv6 的继续教育教学资源库的应用研究

基金：塞尔网络下一代互联网技术创新项目 基金号：NGII20180512

职位：参与者

获奖情况

1. 2019 年 11 月校研究生学业奖学金三等奖
2. 2020 年 9 月全国计算机等级考试三级信息安全技术合格证书
3. 2020 年 11 月校研究生学业奖学金三等奖
4. 2020 年 11 月第二届广西大学生人工智能设计大赛一等奖
5. 2021 年 5 月计算机技术与软件专业技术资格中级网络工程师
6. 2021 年 11 月校研究生学业奖学金二等奖

致谢

提笔致谢，才知三年时光已然流逝。奔赴千里来到陌生的城市开始求学之路，对新环境，新师长，新朋友满怀期许。从初见到熟知，从陌生到亲切，至此竟是万分不舍，尽管再不舍，也总是要向前看的，人生旅途中的这一站停靠，感谢大家的陪伴，这也将是我非常珍惜的宝贵回忆。在此，诚挚的感谢所有给予我关怀和帮助的老师们和朋友们！

首先，我要郑重的对我的导师崔建明表达谢意，感谢老师在学习上、生活上、感情上给与的诸多帮助与关怀。清楚的记得，初来桂林，初来学校，一切都很陌生，而且刚开学不久就是中秋节，崔老师不仅带我和师兄师姐们一起吃饭，并且给我们每人一盒月饼，温暖就是那么猝不及防撞我满怀。研二开题前期，我对研究方向还很迷茫，跟老师谈话后，老师耐心的给我指导方向，提供经验。然后，我要感谢杨呈永老师对我学习上的督促和指导，在论文修改中提出宝贵意见，我才可以按时完成毕业论文的撰写。同时也要感谢信息科学与工程学院的各位传道授业解惑的老师，尽职尽责，奠定了我的基础知识。还要感谢同学们、室友们三年来的陪伴、帮助和鼓励，感谢智春师兄对论文框架给予指导，感谢马雪皎在各种活动、考试和通知上的提醒，感谢支佩佩对我生活中的照顾，感谢郝薇、霍佳欣、周楠、郑婧、陈锦玉、王悦悦、王真梅等同学们的陪伴，让我三年求学生涯充满欢乐，丰富了我的感情生活，祝大家未来事业有成，万事顺遂。

最后感谢我的家人们给予我大力的支持，在远离家乡的三年求学生涯中，给我心灵上的慰藉以及经济上的支持，感谢他们的关心、支持、理解！

感谢百忙之中参加论文评阅、答辩的各位专家们！祝各位老师们、各位专家们身体健康、平安喜乐！