东 莞 理 工 学 院

本 科 毕 业 论 文

|  |  |  |
| --- | --- | --- |
| 毕业设计题目： | 宿舍分配模型与系统——算法设计与实验 | |
| 学生姓名： | 杨良勇 | |
| 学 号： | 201541413108 | |
| 院 系： | 网络安全学院 | |
| 专业班级： | 2015级网络工程1班 | |
| 指导教师姓名及职称： | | 秦勇 教授 |
| 起止时间： 2018年 12月—— 2019 年 5月 | | |

摘要

针对传统学生宿舍分配效率低，不能按照学生意愿分配等情况，本文提出按学生个性化需求分配宿舍方法，以多种改进的多约束资源分配问题解决算法为基础，搭建智能宿舍分配系统，充分考虑学生的兴趣爱好，作息时间等个性化需求，将满足需求的学生分配到一间宿舍。同时对贪心算法、KNN算法、改进KNN算法、图论最小生成树算法和k-means算法的分配结果进行量化建模，从学生的不相同需求项及其权重入手分析学生之间的匹配情况，从而评估各种算法的分配结果。仿真数据表明，各种算法的分配结果各有其优点与缺点，但与传统宿舍分配方法相比，各种算法的分配结果都更加充分考虑学生自身情况，尊重学生个人意愿，有助于宿舍内学生的和谐相处，构造其乐融融的宿舍氛围，为学生提供一个良好的学习和生活环境。在宿舍分配过程充分考虑学生的住宿偏好，使得学生宿舍的分配更具人性化。同时结合互联网信息技术，实现高校学生宿舍分配管理的个性化与自动化。

**关键词** 宿舍分配、学生偏好、宿舍管理、数据挖掘

**Abstract**

In view of the low efficiency of traditional student dormitory allocation and the inability to allocate according to students' wishes, this paper proposes to allocate dormitory according to individualized needs of students, and builds a intelligent dormitory allocation system based on a variety of improved multi-constrained resource allocation problem solving algorithms. Students' hobbies, work schedules and other personalized needs, students who meet the needs are assigned to a dormitory. At the same time, the distribution results are quantitatively modeled, and the matching between students is analyzed from the different needs of students and their weights, so as to evaluate the distribution results of various algorithms. The simulation data shows that the distribution results of various algorithms have their own advantages and disadvantages. However, compared with the traditional dormitory allocation method, the distribution results of various algorithms fully consider the students' own situation, respect the individual will of the students, and help the dormitory. Students live in harmony, construct a happy dormitory atmosphere, and provide students with a good learning and living environment. In the dormitory allocation process, students' accommodation preferences are fully considered, making the distribution of student residences more human. At the same time, combined with Internet information technology, the individualization and automation of university student dormitory allocation management is realized.

**Keywords** dormitory allocation, student preference, dormitory management, data mining

**目 录**

[第1章 绪论 1](#_Toc6756920)

[1.1研究背景与意义 1](#_Toc6756921)

[1.1.1研究背景 1](#_Toc6756922)

[1.1.2研究意义 2](#_Toc6756923)

[1.1.3国内外研究现状 3](#_Toc6756924)

[1.2论文的章节编排 3](#_Toc6756925)

[第2章 评估模型建立 5](#_Toc6756926)

[2.1 模型构建的理论构想 5](#_Toc6756927)

[2.2学生个性化属性的确定 5](#_Toc6756928)

[2.3模型假设 7](#_Toc6756929)

[2.4定义符号说明 7](#_Toc6756930)

[2.4.1符号定义 7](#_Toc6756931)

[2.4.2两个学生之间匹配度MR（Matching Rate） 8](#_Toc6756932)

[2.4.3宿舍内学生之间匹配度MR（Matching Rate）： 13](#_Toc6756933)

[2.4.4 整体匹配度MR（Matching Rate）： 14](#_Toc6756934)

[2.4.5 宿舍匹配度标准差（Standard Deviation）： 14](#_Toc6756935)

[2.4.6 低匹配度宿舍数量LMRN（Low MR Number ）： 14](#_Toc6756936)

[2.5 模型建立 15](#_Toc6756937)

[2.6 宿舍分配的原则及约束 15](#_Toc6756938)

[2.6.1 宿舍分配中的原则 15](#_Toc6756939)

[2.6.2 宿舍分配中的原则 16](#_Toc6756940)

[2.7 模型分析及求解 17](#_Toc6756941)

[第3章 算法实现 18](#_Toc6756942)

[3.1 算法选择 18](#_Toc6756943)

[3.2 贪心算法 19](#_Toc6756944)

[3.2.1 贪心算法介绍 19](#_Toc6756945)

[3.2.2 贪心算法在学生分配宿舍上的应用 19](#_Toc6756946)

[3.2.3 贪心算法分析 20](#_Toc6756947)

[3.2.4 仿真及实验数据分析 21](#_Toc6756948)

[3.3 k最近邻算法 25](#_Toc6756949)

[3.3.1 k最近邻算法介绍 25](#_Toc6756950)

[3.3.2 k最近邻算法在学生分配宿舍上的应用 25](#_Toc6756951)

[3.3.3 k最近邻算法分析 25](#_Toc6756952)

[3.3.4 仿真及实验数据分析 26](#_Toc6756953)

[3.4 改进k最近邻算法 29](#_Toc6756954)

[3.4.1 改进k最近邻算法介绍 29](#_Toc6756955)

[3.4.2 改进k最近邻算法在学生分配宿舍上的应用 29](#_Toc6756956)

[3.4.3 改进k最近邻算法分析 30](#_Toc6756957)

[3.4.4 仿真及实验数据分析 31](#_Toc6756958)

[3.5基于图论的最小子树生成算法 34](#_Toc6756959)

[3.5.1基于图论的最小子树生成算法介绍 34](#_Toc6756960)

[3.5.2 基于图论的最小子树生成算法在学生分配宿舍上的应用 34](#_Toc6756961)

[3.5.3基于图论的最小子树算法分析 35](#_Toc6756962)

[3.5.4 基于图论的最小子树算法仿真及实验数据分析 35](#_Toc6756963)

[3.6 k-means聚类算法 40](#_Toc6756964)

[3.6.1 k-means聚类算法介绍 40](#_Toc6756965)

[3.6.2 k-means聚类算法在学生分配宿舍上的应用 40](#_Toc6756966)

[3.6.3 k-means聚类算法分析 41](#_Toc6756967)

[3.6.4 k-means聚类算法仿真及实验数据分析 41](#_Toc6756968)

[3.7 蚁群聚类算法 45](#_Toc6756969)

[3.7.1 蚁群聚类算法介绍 45](#_Toc6756970)

[3.7.2 蚁群聚类算法在学生分配宿舍上的应用 45](#_Toc6756971)

[3.7.3 蚁群聚类算法分析 46](#_Toc6756972)

[3.7.4 蚁群聚类算法仿真及实验数据分析 46](#_Toc6756973)

[第4章 结果分析及评估 49](#_Toc6756974)

[4.1 算法分析 49](#_Toc6756975)

[4.1.1 宿舍分配结果比较 49](#_Toc6756976)

[4.1.2 综合分析 49](#_Toc6756977)

[第5章 宿舍分配系统 50](#_Toc6756978)

[5.1 功能模块详细设计 50](#_Toc6756979)

[5.1.1 基础功能模块 50](#_Toc6756980)

[总结与展望 50](#_Toc6756981)

[参考文献 52](#_Toc6756982)

[致 谢 54](#_Toc6756983)

# 第1章 绪论

随着社会经济的发展，人们对精神生活追求也越来越高，同时，新时代下大学生们对高效的宿舍管理要求和优质的服务质量要求也越来越高。传统的高校宿舍管理和服务方式已经不能满足学生的需求，如何提高高校学生宿舍服务的满意度，也是高校后勤宿舍管理工作的重要课题之一。

## 1.1研究背景与意义

## 1.1.1研究背景

教育部在《教育信息化十年发展规划(2011-2020年)》（以下简称《规划》）中将高校信息化建设任务提到战略层面。《规划》中指出高等教育信息化是促进高等教育改革创新和提高质量的有效途径，是教育信息化发展的创新前沿。应推动信息技术与高等教育深度融合，创新服务模式，促进高等教育质量的全面提高[1]。由此可见，高校信息化建设已经成为当前高校建设的重要任务之一。随着高校教学管理制度的改革，很多高校已经实行学分制，班级在学院构建方面也仅仅起着一个结构群体的作用，导致了班级形式的集体意识日益弱化，班级组织活动困难。而大学生宿舍集体形式在高校中显得越来越重要，高校的宿舍环境对大学生成才有着不可忽视的作用。

习近平在中国共产党第十九次全国代表大会上的报告《习近平：决胜全面建成小康社会 夺取新时代中国特色社会主义伟大胜利》中指出，要全面贯彻党的教育方针，落实立德树人根本任务，发展素质教育，推进教育公平，培养德智体美全面发展的社会主义建设者和接班人[2]。也即表明我国对学生的全面发展的重视，其中对学生的素质教育包括学生的道德素质、能力培养、个性发展、身体健康和心理健康等方面。而宿舍是学生的“第一社会”、“第二家庭”、“第三课堂”，能否使得学生在宿舍有一个良好的生活学习环境，将直接影响到对学生素质教育效果。

据教育部发布的统计数据表明，2017年我国高等教育在学总规模达3779万人，如果按照每2-6人住一间宿舍计算，大约需要944万间学生宿舍。面对如此庞大数量的学生宿舍，如何管理好，服务好，是高校后勤工作需要研究的重点课题。

而近年来，我国高校学生宿舍矛盾事件频发，甚至上升为血腥事件，给高校和社会带来了严重的不良影响。如2004年杀害4名舍友的马加爵事件，2009年因舍友打呼噜而引发激烈矛盾的郭力维事件，2012年陕西科技大学7·2事件，2013年复旦投毒案及南航大由钥匙引发的人命案件，2016年四川大学重大案件等，均引起了高校和社会对大学生宿舍安全事件的强烈关注。从这些事件的发生折射出高校学生管理工作存在问题，亟需采取措施。

中青网2017年10月就大学生宿舍关系话题，对全国958名大学生进行调查，结果显示：42.28%的学生与舍友曾经发生矛盾，有28.29%的学生希望更换宿舍舍友。

西安建筑科技大学曾对大学生宿舍人际关系做过问卷调查，结果显示宿舍人际关系的好坏直接影响同学的生活学习，其中最大的影响是使自己感到苦闷（36%），忧郁（23%），仇恨（22%），消极悲观（16%），甚至导致成绩下滑（3%）。

据山西大学马丽萍在《大学生宿舍人际关系调查及干预研究》中指出，目前大学生对自己的宿舍人际关系主观体验倾向于满意的约为73.5％，倾向于不满意的约为19.3％ , 持“无所谓”态度的占7.2％。

中国高校传媒联盟随机对全国134所高校的446名大学生发放了室友关系的问卷调查，其中22.42%的受访者没有遇到过寝室矛盾，77.58%的受访者经历过室友不和，其中3.14%的受访者表示寝室矛盾经常发生，7.40%的受访者甚至想搬离。（中国青年报2016年5月16日12版《大学室友友谊的小船为啥说翻就翻》）

由上述材料可以看出目前大学生宿舍人际关系问题存在的普遍性，而人际关系的好坏也会直接影响宿舍生活的质量，也直接体现了大学宿舍生活质量与大学生的期待存在一定的差距。

## 1.1.2研究意义

学生宿舍是大学生在校生活、学习和休息的重要场所，学生约有60%的时间是在宿舍中度过的[3]。因而学生宿舍生活对大学生的道德培养和个性形成的重要性不言而喻，一个良好的宿舍生活氛围同时也有助于养成学生的身体、心理健康。另一方面，宿舍生活中的人际关系会对人性格产生重要影响。Markey PM[4][5]等人通过在高校实验，对比了宿舍组成的2周后与15周后的宿舍成员的性格变化，证明了宿舍成员会产生性格上的相互的影响，即“Abstract Robert Carson's principle of complementarity(互补性原则)”。

本文通过调研及查阅相关文献发现引发大学生宿舍矛盾的主要原因是是宿舍内学生之间由于日常生活的作息习惯、卫生习惯、风俗习惯、兴趣爱好等的巨大差异。这与高校在学生入住时采取随机分配入住的方式，而没有充分考虑学生的住宿偏好有着密切关系。目前已经国内己经有部分高校开始尝试新的宿舍分配方式，将学生的兴趣爱好、行为习惯等作为宿舍分配的参考因素。例如，2014年，浙江大学通过网络向新生采集“生活习惯”，把习惯相近的同学安排住一起，尽量避免因学生生活习惯的差异而引起的矛盾事件。从2015年开始，北京科技大学研究生可以通过网上选室友；同年，电子科技大学宿舍管理中心首次开通2015级新生网上选择宿舍功能，开始推进网上选宿舍方案。2016年，上海大学通过网络平台采集学生对住宿及室友的相关需求，新生可以根据不同的兴趣爱好行为习惯自选寝室，实现高校公寓的“私人定制”。同济大学也进一步改善了已有的学生网上选宿舍系统，方便学生、服务学生。各大高校纷纷结合互联网技术推出富有特色的“互联网＋公寓”的新型管理方式。

为了打破传统的学生公寓随机分配模式，满足学生多样化的、复杂的住宿需求，提高学生公寓的服务质量，本文运用数据挖掘的相关技术实现大学生个性化需求的宿舍分配，构建了分配结果的评估模型。这对完善高校学生公寓分配管理制度，解决大学生宿舍人际关系紧张现象，构建和谐宿舍具有重要的现实意义。

## 1.1.3国内外研究现状

近年来，我国部分高校开始尝试新的寝室分配方法。2014年以来，中国人民大学、浙江大学、上海交通大学等学校就在新生入学前对学生进行调查，并根据同学的作息习惯分配寝室。

而南京大学于2018年也推出了基于隐语义模型的学生宿舍分配系统，充分考虑了学生的个性化需求。

余启林[6]于2017年在其硕士毕业论文上首次提出并尝试了直觉模糊聚类以及FCM(Fuzzy C-means)聚类在学生偏好分类和宿舍分配领域的应用，并实现了相应的程序，但仅通过算例验证其算法可行性，并未对其分配结果做比较分析。

苏明杰[7]于2011年提出采用蚁群算法及线性规划模型来处理学校的排课问题，对蚁群算法在排课系统中的运用做了详细阐述。

王巧巧等[8]人于2013年首次提出贪心算法在宿舍分配中的应用，使得基于学生偏好的自动分配系统成为可能。但文章中并未未提到计算匹配度的具体方法。

尽管关于高校学生宿舍分配的个性化研究已经实现了从0到1的突破，但是可供借鉴的宿舍分配案例，特别是自动化的宿舍分配的案例，却是寥寥无几。本文经过匹配度计算建模，使用多种聚类算法实现宿舍的自动分配，并通过仿真实验对比说明各算法的特点。

## 1.2论文的章节编排

本课题主要是通过多种算法进行基于学生个性化需求的宿舍分配，并通过建模及仿真对多种算法执行效果进行分析比较。同时设计和实现一个可操作的，多维度可控制的学生宿舍分配系统。本章主要为介绍论文的研究背景与研究意义。

本论文的章节安排如下：

第1章是引言，简要的介绍本课题研究的背景与意义，以及介绍了论文内容的编排。

第2章是本文重点章节之一，是对分配宿舍的效果评估模型进行详细阐述，主要是说明模型构建的理论构想，确定学生个性化属性，定义学生匹配度计算方式，以及评估模型的确定。

第3章是本文重点章节之一，是对多种个性化分配学生宿舍的算法进行算法思想说明，在宿舍分配上的应用，以及算法仿真结果的分析。

第4章是在第三章的基础上，对不同算法执行的结果进行对比评估，并将数据可视化，为全文的核心章节。

第5章是宿舍管理系统的应用，是该课题的生产实际的应用，主要是对系统的界面及相关操作进行展示。

# 第2章 评估模型建立

## 2.1 模型构建的理论构想

在经济学领域，偏好用来表示消费者对某一种消费按照自己的愿望所进行的排序，也就是在决策过程中，决策者对当下决策行为的一种价值倾向。在高校学生公寓分配时，将学生偏好作为重要的参考因素，使用不同的方法将具有相似偏好属性的学生分配在同一宿舍，满足学生的个性化需求，提高高校公寓的“高质量”服务，体现了以人为本的管理理念。

高校学生宿舍分配实际上是一个资源分配问题，如何在学生宿舍分配的过程中加入学生偏好这一变量，就需要对该现实问题进行抽象。将学生集合定义为X={x1,x2,…,xm}，每一个学生都由p个学生个性化属性（即偏好）进行表示，记作xk={xk1,xk2,…xkp}。这样就将一个分配管理问题，转化为分类问题，将m个学生依据p个特征属性(即个性化属性)划分成m/n类（n为宿舍容量，m/n为宿舍数量）。

## 2.2学生个性化属性的确定

文献郝鹃[9]，李巍巍[10]，曹雪雪[11]等通过问卷调查的方式，对高校学生宿舍的影响因素的进行比较，分析调查结果总结得到，在大学生宿舍中影响人际关系的众多因素中，生活习惯、个性特征、日常行为、作息规律、价值观、性格因素、睡眠行为、社会认知、自觉意识等是相对重要的几种。何饶依[12]在文献也指出，认知偏差、情感障碍以及行为不当是引发大学生宿舍人际关系矛盾的重要内部原因，而家庭条件差异、高校德育缺陷、主管部门管理问题，社会影响是是引发大学生宿舍人际关系矛盾的重要外部原因。在文献[13]中，郑志也通过现场调查，问卷调查等方式进行调查，然后采用多种方式对调查结果进行分析，构建了大学生宿舍生活事件量表，为本文确定大学生住宿的个性化属性提供了借鉴。

结合上述己有的研究成果及对在校学生的问卷调查和访谈，本文总结归纳出八种影响大学生住宿满意度重要影响因素，列为住宿个性化属性，分别是：“作息习惯”、“兴趣爱好”、“睡觉是否说梦话打呼噜”、“月均生活费”、“生源地”、“卫生情况”、“热爱电竞类型”、“宿舍氛围期待”。

然后对这八种学生个性化属性进行枚举赋值及相应的需求属性枚举赋值，得到下列属性枚举表

表格 2-1 学生个性化属性值枚举

|  | 属性项 | 属性值枚举 | 有无权重 |
| --- | --- | --- | --- |
| 基本属性 | 姓名 |  | 无 |
| 性别 | 男，女 | 无 |
| 学号 |  | 无 |
| 学院 |  | 无 |
| 专业 |  | 无 |
| 班级 |  | 无 |
| 个性化属性 | 宿舍氛围期待[t](OC1) | 吵闹嬉戏，安静学习 | 无 |
| 卫生情况[t] (OC2) | 每天拖地，每周拖地，从不拖地 | 无 |
| 睡觉是否说梦话打呼噜[t] (OC3) | 是，否 | 无 |
| 作息习惯[t] (OC4) | 早睡早起，正常，晚睡晚起 | 无 |
| 是否热爱电竞[t] (OC5) | LoL，王者荣耀，刺激战场，不热爱电竞 | 无 |
| 兴趣爱好(OC6) | 羽毛球，音乐，篮球，健身，化妆，跳舞，计算机，乒乓球 | 无 |
| 月均生活费(OC7) | 500以下，500-1000,1000-1500,1500-2000,2000以上 | 无 |
| 生源地(OC8) | 潮语区，粤语区，粤西，广东外 | 无 |
| 个性化需求属性 | 宿舍氛围期待[t](OD1) | 吵闹嬉戏，安静学习 | 有 |
| 对室友卫生情况期待[t] (OD2) | 每天拖地，每周拖地，从不拖地 | 有 |
| 睡觉是否说梦话打呼噜[t] (OD3) | 是，否 | 有 |
| 对室友作息习惯期待[t] (OD4) | 早睡早起，正常，晚睡晚起 | 有 |
| 对室友是否热爱电竞期待[t] (OD5) | Lol，王者荣耀，刺激战场，不热爱电竞 | 有 |
| 对室友兴趣爱好需求(OD6) | 羽毛球，音乐，篮球，健身，化妆，跳舞，计算机，乒乓球 | 有 |
| 对室友月均生活费需求(OD7) | 500以下，500-1000,1000-1500,1500-2000,2000以上 | 有 |
| 对室友生源地期待(OD8) | 潮语区，粤语区，粤西，广东外 | 有 |

注：1、其中注[t]项为个性化属性与个性化需求属性的属性值相同，避免出现“不管自己是否打呼噜，都想找不打呼噜的同学做室友”等情况，处理方式为“打呼噜的同学优先和打呼噜的同学组合”。2、OCi：ith Orginal Character（个性化属性）、ODi：ith Original Demand（个性化需求属性）

## 2.3模型假设

本文重点为对各种算法分配的结果进行评估，为降低数据复杂度，剔除部分对算法执行结果不影响的因素，做出如下假设：

1. 假设每间宿舍房间类型相同，即不同的宿舍宿舍容量相同，均为4人或均为6人宿舍。
2. 假设所有学生来自于同一个年级，当学生来自不同年级时，可按同一年级对待处理，也可对不同年级的学生分别执行分配算法，不同年级互不影响。
3. 房间总容量大于学生数量，即宿舍容量足够，分配过程不考虑宿舍不足的情况。
4. 以男生入住宿舍为例，男女生一般不混住，可分别执行分配算法，不同性别互不影响。
5. 假设获取到的所有学生个性化属性为学生真实意愿，与真实情况没有偏差或者偏差很小。

## 2.4定义符号说明

## 2.4.1符号定义

算法选择和效果评估过程中使用以下术语，现分别定义如下

表格2-1 符号定义

| 序号 | 符号 | 符号定义 | 备注 |
| --- | --- | --- | --- |
| 1 |  | a同学对b同学的匹配度(Macthing Rate) |  |
| 2 |  | a,b两个同学之间的匹配度(Macthing Rate) |  |
| 3 |  | a,b两个同学之间的匹配度的差值(Difference Value) |  |
| 4 |  | 个性化需求属性经过0-1规格化后的值(Demand) |  |
| 5 |  | 个性化属性经过0-1规格化后的值(Character) |  |
| 6 |  | 宿舍成员a,b,…,n同学组成之间的匹配度(Macthing Rate) |  |
| 7 |  | 1,2,…,n个宿舍的整体匹配度(Macthing Rate) |  |
| 8 |  | 1,2,…,n个宿舍的匹配度的标准差(Standard Deviation) |  |
| 9 |  | 约束条件 |  |
| 10 |  | 学生 |  |
| 11 |  | 宿舍 |  |
| 12 | 个化项（C） | 学生属性：兴趣爱好，月均生活费，生源地，卫生情况等属性 | 个性化属性 |
| 13 | 需求项（D） | 学生对室友的期待属性：作息习惯，是否热爱电竞，对室友兴趣爱好需求，对室友月均生活费需求 | 个性化需求属性 |

## 2.4.2两个学生之间匹配度MR（Matching Rate）

下以计算a同学和b同学之间的匹配度为例进行说明。

由于不同的属性项对不同学生的重要性程度，不同的学生也有不同的期望，两个学生之间的匹配结果可能出现“a同学对b同学这个室友很满意，而b同学对a同学这个室友很不满意”的情况，如a同学情况为：喜欢羽毛球，但想找喜欢篮球的室友；b同学情况为：喜欢篮球，但也只想和喜欢篮球的同学成为室友，则对两个同学在该属性上的组合可能出现上述情况。

为解决这个矛盾，两个学生之间匹配度计算采用双向匹配的计算方法，同时引入第二个变量：差值，具体算法如下：分别计算a同学对b同学的匹配度（基于a同学的个性化需求属性对b同学的个性化属性的不同的重要性程度和期望），记为，再计算b同学对a同学的匹配度（基于b同学的个性化需求属性对a同学的个性化属性的不同的的重要性程度和期望），记为，则两个同学之间的匹配度，两个同学之间匹配度的差值为，记为。若出现上述分析情况，则a同学与b同学匹配度值不高，差值也较大。 

以上均为定性分析，下面具体讨论a同学对b同学匹配度的计算方法（b同学对a同学匹配度的计算方法相同）。

a同学对b同学匹配度计算公式（公式1）：



其中为a同学个性化属性项中第项需求项（即个性化需求项，下同）；为该项的重要性程度（权重），取值范围；为b同学个化项（即个性化属性项，下同）中与对应的属性项；为a同学个性化需求项总数；为a同学对b同学的个性化项的差值，取值范围；为为a同学对b同学的个性化项加权差值平均数，即为两个同学在多维个化属性项上的曼哈顿距离与属性项数的比值，取值范围，当曼哈顿距离越接近0时，表示两个学生相似度越高，匹配度越高。即可用表示匹配度，表示b同学完全符合a同学的期望，表示b同学完全不符合a同学的期望。

本文在确定的系数是还是时，进行了如下分析。如果某同学只对部分个性化属性有需求，而对其他剩下属性没有需求，个性化项加权差值使用做为系数，将导致匹配度与实际相比偏高。如某同学只对其中一项有需求（该项权重为，其他项权重为0），而并没有未其匹配到符合其要求的室友时，匹配度计算结果为87.5%，与实际期望不符。而使用为个性化项加权差值的系数，匹配度计算结果为0%，与实际期望较接近。同时，在仿真实验中，对随机生成20000条的学生数据进行匹配度计算，使用作为个性化项加权差值的系数，两两学生之间平均匹配度为75.34%（随机生成的学生数据理论平均匹配度：50%），而使用作为个性化项加权差值的系数，两两学生之间平均匹配度为46.95%（多值属性的存在使该值较理论值50%的匹配度有所偏低）。仿真结果说明，比生成的匹配度更合适做为个性化项加权差值的系数。

这里对匹配度计算方式再做进一步说明，目前常用的人与人之间相似度计算方法有很多种，如皮尔森相关系数（Pearson correlation coefficient）或余弦相似度（Cosine simnilarity）[14][15]。根据具体问题的不同，各种相似度的计算方式的适应程度也不一样。如果对象数据稀疏，存在很多零值，并且数值大小不会对相似度造成较大影响，那么可以选用余弦距离（余弦相似度）计算相似度。因为余弦距离即角度距离，对于数值大小来说，它更注重于方向，同一向量各维度比例成倍增加或减少，对余弦距离的值不会产生影响。而如果数据使用了不同的量纲和单位，那么可以使用皮尔逊相关系数来计算相似度。因为皮尔逊相似度与其他计算相似度方式的最大区别在于它更重视数据集的整体性。在本课题中，学生属性的数据稠密，且属性值的大小会影响学生的匹配度（将在本节后半部分计算过程详细说明）故余弦距离并不适用。同时，不同的数据维度都进行了归一化，量纲与单位相同，使用皮尔逊距离与欧式距离，曼哈顿距离计算的结果都呈正相关，对分配结果不产生影响。曼哈顿距离计算用户属性在标准坐标系上的绝对轴距总和，计算复杂度低，故而选择曼哈顿距离作为学生匹配度计算的核心。

下面再具体讨论个性化项差值计算方式。不同的属性用不同的尺度度量（同一属性用不同单位也可能会影响属性差值大小），如果直接比较个性化项属性获得的差值，某些属性的计算结果可能被另外一些使用较大尺度度量的属性完全削弱，所以，我们需要将所有的属性值0-1规格化，即将属性值按比例放缩在0和1之间。通过计算（公式2）：



这里是属性的真实值，最大和最小属性值是从所有实例（即所有待分配的学生的）属性中获得的。同样也需要经过0-1规格化处理。处理后和都在之间，即可保证的取值范围为。

这些公式隐含着一个假设：这些个性化属性为数值属性。这个情况下，个性化项差值就是它们之间的数值差异。但在公式中除了权重变量为数值属性外，其他变量均可能为名词性属性，即属性值是符号值而不是数值。对于个化项及需求项的值是名词性属性（其值是符号值而不是数值的属性，如兴趣爱好项的值为羽毛球），具体处理方式如下：

1. 公式1中需求项及公式2中的为经过预处理后的需求项，如a同学的个化项中兴趣爱好项值为[羽毛球, 音乐]，若该同学选择期望与自己兴趣爱好不同（或相同）的同学组合成宿舍成员，则该个化项对应的需求项为室友与自己爱好不相同（或相同），从而a同学的需求项中的兴趣爱好项为 [篮球，健身，化妆，跳舞，计算机，乒乓球]（或[羽毛球, 音乐]），即需求项与个性项相同时与个化项相同，=；需求项与个性项不相同时为该个化项全集与该同学个性化值的差集，。
2. 两个不同名词的属性值之差常认为是1，如果名词性属性值相同，他们差值为0，这里无需量度尺寸，因为只使用1和0。但是在该生产实际中，不同名词的属性值之差都认为是1，会出现明显不符合实际的情况。比如在作息习惯中，早睡早起的同学和晚睡晚起的同学之间差异较大，差值是1符合实际，但早睡早起的同学和正常生活作息的同学差异明显小于早睡早起的同学和晚睡晚起的同学之间的差异，若差值也是1，即理解成差异相同，与实际差距较大。故对该类分段属性处理为，早睡早起的同学和晚睡晚起的同学在该类个性化属性项的差值为1，但正常作息的同学与上述两类同学在该类个性化属性项的差值为0.5。
3. 若个性化属性项为分段属性，则可以按照分段数量和该分段字符值在分段中的位置确定各个字符值的数值。如有个化项家庭人均月收入或月均生活费，其值为500以下，500-1000，1000-1500，1500-2000及2000以上，对该属性符号值0-1规格化确定字符值对应的数值分别为0，0.25，0.5，0.75及1。该项属性的需求项与个化项的差值为0-1规格化后的数值的差值。如500以下与500-1000差值为0.25，500以下与2000以上差值为1，与实际理解相符。
4. 对残缺值的处理。一个通用的残缺值的处理方式为：对于名词性属性，假设残缺属性值与其他属性值的差值达到最大值。因此如果两个属性值中的一个或两个都残缺，或者如果两个属性值不相同时，那么他们之间差值为1，只有当两个两个属性值都不残缺，并且相同时，他们之间的差值才为0。对于数值属性，两个残缺值之差也为1，但是，如果仅有一个属性值残缺，那么他们的差值是另一个属性值的0-1规格化值，或是1减去那个0-1规格化值，取两者中较大的那个，这意味着如果有属性值残缺，他们之间的差值将达到可能的最大差值。但是该处理方式与生产实际会存在较大差距，个性化属性项一般由学生填写，存在残缺值一般为学生留空，对学生来说属于“无所谓”项。故在处理时不同的残缺情况应该按不同的方式处理，从而提高宿舍分配的匹配效果。若需求项残缺或者需求项与个化项都残缺，则残缺属性值与其他属性值的差值达到最小值，即差值为0，若只有个化项残缺，则残缺残缺属性值与其他属性值的差值达到最大值，按通用处理方式处理。
5. 若需求项为多值属性，而为单值属性时，此时只要个化项不存在于需求项中，即，则认为此属性个化项不符合需求项，两个属性项差值即为1。而如果个化项存在于需求项中，即，则认为此属性个化项符合需求项。

如a同学个化项中生源地为潮语区，，需求项为室友生源地与自己不同， ，b同学生源地为粤语区，，从而得到，故b同学的该项个化项符合a同学的需求项，差值为0。 a同学对b同学在该项属性的匹配度为100%。

1. 若需求项为多值属性，而也为多值属性时，此时对该属性值的距离计算方式类似于Jaccard 距离。若中的元素完全不在中，即，即认为b同学完全不符合a同学的要求，该项差值为1。若此时若中的元素部分在中，即，则考虑b同学个性化项中符合a同学需求项的项数。对只考虑中存在于中的元素，即，故而按计算与之间的差值，其中为的值的项数。

如a同学个化项中兴趣爱好项值为[羽毛球, 音乐]，即，对舍友兴趣爱好期待为[羽毛球，音乐，篮球]，即，而b同学的兴趣爱好为[羽毛球，音乐，健身]，。由于b同学爱好健身与a同学的需求[篮球，音乐，篮球]不相关，所以。可以看出b同学的兴趣爱好有两项符合a同学的需求，故a同学对b同学在该项属性的差值为。 即“a同学想要的3项中，b同学有一项不能满足”

## 2.4.3宿舍内学生之间匹配度MR（Matching Rate）：

宿舍内学生之间匹配度定义为



其中表示同一宿舍内a同学，b同学……n同学之间的匹配度，代表i同学和j同学之间的匹配度，代表宿舍内每两两个同学之间匹配度之和，即为宿舍内每两个个同学之间匹配度的平均值。

由定义可知，一个较好的分配方案，应该使得宿舍内学生之间的匹配度值较大。

## 2.4.4 整体匹配度MR（Matching Rate）：

整体匹配度定义为



其中表示所有n个宿舍的匹配度，表示第i个宿舍的匹配度，表示所有宿舍匹配度之和，即为所有宿舍的匹配度的平均值。

由定义可知，一个较好的分配方案，应该使得整体匹配度值较大。

## 2.4.5 宿舍匹配度标准差（Standard Deviation）：

宿舍匹配度标准差为



其中表示在该分配方案下，n个宿舍的匹配度的标准差，表示第i个宿舍的匹配度，表示整体匹配度，即所有宿舍匹配度的平均值，即为宿舍匹配度的标准差。

由定义可知，标准差体现不同宿舍的匹配度的离散情况，即偏离平均值的距离程度，若分配策略不平衡，导致部分宿舍匹配度较高，而部分宿舍匹配度较低，将会导致宿舍匹配度标准差值较大，一个较好的分配方案，应该使得宿舍匹配度标准差值较小。

## 2.4.6 低匹配度宿舍数量LMRN（Low MR Number ）：

在仿真实验进行时发现，部分算法分配结果整体匹配度较高，但其中个别宿舍匹配度较低。而且宿舍间匹配度可能差异较大，且受数据规模影响，使用宿舍匹配度标准差并不能很好的反映出这个情况。为方便统计这部分宿舍，引入变量LMRN，定义如下：



其中表示在该分配方案下，匹配度低于平均匹配度80%以下的宿舍的数量。表示第i个宿舍的匹配度，表示平均匹配度，即表示匹配度低于平均匹配度80%以下的宿舍的数量。

由定义可知，LMRN值的大小表示分配结果不均的情况，当LMRN值较小，表明整体宿舍分配情况较平衡，效果较好，当LMRN值较大，则表明整体宿舍分配情况较不平衡，部分宿舍匹配度严重低于平均匹配度水平。一个较好的分配方案，应该使得低匹配度宿舍数量LMRN值较小。

## 2.5 模型建立

宿舍分配问题属于组合优化问题，最优化问题由目标函数和约束条件两部分组成，设有n个宿舍，编号为1,2，…,n。m个学生，编号为1,2，…,m。第i个宿舍宿舍匹配度为， 为第i个宿舍人数，每间宿舍宿舍容量为 。经上述分析，数学模型

目标函数为







约束条件为



## 2.6 宿舍分配的原则及约束

## 2.6.1 宿舍分配中的原则

首先，为了安全和生活方便，男生和女生宿舍区一般都分开。如果实际条件无法按区域分隔开来，那么应该尽量按楼栋或楼层进行分隔，避免男女生宿舍交叉的情况。然后，为了方便管理及活动开展，不同学院之间的宿舍区也应该按楼栋及楼层进行分隔，同一学院应尽量安排在同一楼栋或相近楼栋，同一楼层或相近楼层，尽量避免不同学院交叉。其次，为了方便开展迎新活动，毕业就业活动，早晚修管理，日常活动等，在安排相同年级的同学时应该尽量集中，最后，若若安排时当前楼层已经安排完所有宿舍，继续安排下一个楼层时应该按另一个方向开始，按“z”字型顺序安排，不需从头开始，便与同一个类型的宿舍聚集。

## 2.6.2 宿舍分配中的原则

在宿舍安排时，需要考虑的方面比较多，要在分配方案可行的情况下，尽量满足学生的个性化需求。实际设计时，我们考虑老师，学生，及宿舍管理的工作人员三方的要求，分析在要求冲突时应该按谁的要求为标准进行宿舍分配。

宿舍分配其实就是一个多约束资源分配问题，先把设计定义好的限制条件输入分配系统，作为一个资源分配的事件被组合优化。我们可以把约束条件分为必备约束和补充约束，所有可以的和有意义的方案，必须满足必备约束，然后在此基础上尽量满足补充约束

|  |  |  |
| --- | --- | --- |
| 编号 | 必备约束 | 补充约束 |
| 1 | 学院或专业相同要求：当分配时管理员指定按同一宿舍须相同学院或专业方案分配时，学生指定期望室友的学院或专业将无效，不管此时学生的期望值多高 | 学院或专业个性化要求：当分配时管理员不指定按同一宿舍须相同学院或专业，则应按学生个性化要求优先匹配期望的学院或专业的学生 |
| 2 | 性别相同要求：分配时不同性别应该分开安排，即使单一性别安排后末端不整好为满宿舍人数，也不应该将另一性别的同学安排进来 | 作息习惯要求：作息习惯应该按照相同或相近优先考虑 |
| 3 | 粘性末端要求：当出现有宿舍分配后不满人，应该优先处理，避免多个宿舍住不满人或一个宿舍超过两个专业的情况 | 兴趣爱好要求：根据学生指定期望的室友的兴趣爱好优先安排符合条件的室友 |
| 4 | 宿舍人数要求：每个宿舍都应该优先分配满人后再安排下一间宿舍，整体宿舍人员容量应该大于或等于待安排宿舍的同学 | 卫生情况要求：分配时，按整理卫生频率相同的同学优先组合 |

必备约束必须满足，从而产生可行的解决方案，而补充约束的满足会使得分配结果更加满足学生的个性化需求。

## 2.7 模型分析及求解

具体求解算法及求解步骤将在下一章详细描述

# 第3章 算法实现

宿舍分配算法将决定如何按照学生提交的个性化需求为学生分配宿舍，因此宿舍分配算法是整个系统的核心。是否选择合适的算法，将直接对分配结果及效果产生影响。

## 3.1 算法选择

学生宿舍分配问题，属于多约束的资源分配问题[16]，即将一定的资源按照一定的约束条件分配给需求集中的个体。可能有不同的满足约束的分配方式，但不一定都是最优解[17]。对不同的约束条件，不同属性特点的分配对象，多约束资源分配问题有不同的解决方法。迄今为止，并没有哪一种算法可作为通用的算法解决此类问题[18]。国内外研究人员都是根据自身的具体情况选择适合自己的算法。比如动态规划，贪心算法，进化算法等[19][20]。

同时，国内外关于学生宿舍分配算法的研究并不多，主要有回溯法和贪心算法。王文发，马燕等在文献[21]中采用回溯法进行宿舍分配，并通过剪枝及压缩解的空间树的高度，减少了时间复杂度，假定宿舍容量为n，学生人数m，其算法最坏情况时间复杂度为，较同类问题的回溯法（时间复杂度），效率有很大的提高。但是该算法是基于生源地及学习成绩两维属性的固定统一分配策略进行分配的，对于属性维度较多，权重分布分散的个性化学生宿舍分配，该算法时间复杂度的规模为（其中p为属性维度数量），当数据规模稍大时，该算法的运行时间将呈幂数级增长，可行性不高，故本文不考虑此算法。

孙灿等人在文献[22]采用贪心法进行宿舍分配，宿舍容量为m人，学生总人数为n人，时间复杂度为 O( nnm)，本文将在本章3.2节对该算法进行分析，并进行仿真实验对比分析算法分配的结果。

在数据挖掘的相关算法上，由于当下并没有很好的宿舍分配方案及其相应的分配宿舍的数据来作为样本进行分析和训练，故对于需要先定义类别、进行样本训练的监督学习算法，如神经网络，决策树等分类算法，在本课题上无法应用。之后可以在由其他算法生成“较优结果”后，用“较优结果”来训练这类监督学习的分类算法，从而争取达到“更优结果”。本文只考虑如何得到“较优结果”。 但作者将knn分类算法进行一些修改，使得在无需训练样本的情况下将knn应用到学生宿舍分配问题的解决上，具体将在本章中3.3节进行阐明。

与需要先定义类别和训练样本的监督学习算法不同，聚类分析属于无监督学习算法，即给定数据，无需预先划定分类，算法便可以自动按照数据的某种关系测度进行聚类。但是聚类算法在分配宿舍这个问题的解决上需要进行一定的改进应符合该问题的实际，如目前所有的聚类算法都无法指定聚类后类内元素的数量，但是在分配宿舍这个问题上，若将为学生分宿舍问题建模为学生分类，宿舍数量即为类数量，宿舍容量即为类内元素数量，则进行分类时，不仅需要指定类的数量，同时必须对类内元素进行控制。本文选取最常见的聚类分析算法——k-means算法，和蚁群聚类算法进行改进以求在该问题上得到应用。另外，此外还有不属于数据挖掘范畴的算法，如贪心算法，也将被用在此问题的求解上进行尝试。

## 3.2 贪心算法

## 3.2.1 贪心算法介绍

贪心算法也叫贪婪算法，得名于在背包问题中的使用。背包问题说的是一盗贼的背包容量一定，有一批宝物体积和价值都各不同，盗贼如何做到贪心地在容量一定的背包中塞进最大价值量的宝物。虽然后来背包问题被证明是一种组合优化的NP完全问题，有比贪心算法更合适的解决方法。但由此得名的贪心算法也开始在各个领域被使用。贪心算法是在一定约束条件下尽量寻找全局最优的解决方案。贪心算法在解决这类问题时，“总是做出在当前情况下最好的选择”，也就是说，不从整体最优上加以考虑，它所做出的是在某种意义上的局部最优解。所以贪心算法不一定对所有问题都能得到整体最优解。

贪心算法的思想。贪心算法的基本思路是从问题的某一个初始解出发一步一步地进行，根据某个优化测度，每一步都要确保能获得局部最优解。每一步只考虑一个数据，他的选取应该满足局部优化的条件。若下一个数据和部分最优解连在一起不再是可行解时，就不把该数据添加到部分解中，直到把所有数据枚举完，或者不能再添加时，算法停止。

贪心算法的贪心选择。贪心选择是指在问题求解过程中，为得到整体最优解的目标而由贪心策略作出的一系列局部最优的选择。贪心选择是采用从顶向下，以迭代的方法做出相继的选择，每做一次贪心选择就将所求问题简化为一个规模更小的子问题。在理想情况下，可以证明问题的一个整体最优解，是从贪心选择开始的，而且作了贪心选择后，原问题简化为一个规模更小的类似子问题，然后用数学归纳法证明，通过每一步贪心选择，最终可以得到一个整体最优解。对于一个具体问题，要确定它是否具有贪心选择的性质，我们必须证明每一步所作的贪心选择最终能得到问题的最优解。

## 3.2.2 贪心算法在学生分配宿舍上的应用

贪心算法的核心在与贪心选择的策略。将“分配到宿舍的学生”作为“部分最优解”，“未分配到宿舍的学生”为“待选择的解”，则为了尽可能的达到目标函数（最高整体匹配度），贪心策略为“在待选择的解中找到与部分最优解匹配度最高的数据单元做为贪心选择”即“在未分配学生中找到与已分配的学生匹配度最高的学生加入已分配宿舍的学生的宿舍”。第一次执行贪心选择时，部分最优解为空集，由于最后每个学生都要分配宿舍，故无论选择哪一个学生作为部分最优解的元素，都符合贪心策略，即为贪心选择。当部分最优解容量达到上限时，需构造新的部分最优解。即当前宿舍人满之后，将开始为下一个宿舍安排学生。依次将所有待分配宿舍的学生进行分配，直到所有学生分配完毕。

假设学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中贪心算法步骤如下。

1. 算法开始，为第1间宿舍随机选择1个学生入住，执行下一步。
2. 如果剩下的学生里人数小于或等于n-1，将所有剩余学生安排进该宿舍，算法结束。否则执行下一步。
3. 如果剩下的学生人数大于n-1，在剩下的学生里，挑选与该宿舍已有成员匹配度最高的n-1个同学入住。执行下一步。
4. 该宿舍住满，开始安排下一间宿舍，并在剩下的同学里随机选择一个学生入住。执行下一步。
5. 重复ii。

该算法来自[22]，从步骤中我们可以知道该宿舍分配算法中，同一个宿舍后几个分配的同学均为与第一个同学匹配度最高的前n-1个，但并不是最优的选择（特别是当宿舍容量稍大时），已有宿舍成员逐渐增多，此时依旧只考虑与第一个入住的同学属性相匹配明显不合适。故将贪心策略更新为：在待分配的学生中挑选与宿舍已有成员组合后宿舍内学生之间匹配度最高的同学。更新后算法步骤如下。

1. 算法开始，为第1间宿舍随机选择1个学生入住，执行下一步。
2. 该宿舍未住满时，如果待分配宿舍学生人数大于等于1，在剩下的学生里，挑选与该宿舍组合后宿舍内学生之间匹配度最高的一名同学入住，执行下一步。否则算法结束。
3. 如果该宿舍未住满，执行 ii。否则，该宿舍已满，继续安排下一间宿舍，并随机挑选一名学生入住下一间宿舍，执行ii。

## 3.2.3 贪心算法分析

该算法从第一个元素加入局部最优解开始迭代，迭代过程每次选择的元素是确定的，但是由于第一个元素的选择具有随机性，所以此分配算法对同一批待分配的学生可能出现每次分配结果都不一样，也就是说不同次的执行结果的整体匹配度会有所变化。

并且，由于先分配宿舍的学生有较多的待分配宿舍的同学进行选择，而后分配的学生，特别是较后几个宿舍的学生，只能“勉强凑一起”，导致前面部分宿舍匹配度均较高，在尾部宿舍匹配度将急剧下降，从而此算法的宿舍匹配度标准差将较大，LMRN值也将较大。

改进后的宿舍分配方式对于整体匹配度的提高有一定的帮助，但同时也加大了时间复杂度，从加大为，其中m为学生规模，n为宿舍容纳量，p为个化属性数量。

## 3.2.4 仿真及实验数据分析

1. 时间复杂度：更新前为，更新后为，其中m为学生规模，n为宿舍容纳量，p为个化属性数量。一般p，n均可认为是常量，所以时间复杂度为。
2. 仿真环境：硬件环境：2.4GHz基准主频Intel Xeon E5-2680 Broadwell（v4）处理器，DDR4 2G内存；系统环境：linux CentOS 7.3 64位；语言环境：php，python。（其他仿真与本实验相同，将不再特别说明）
3. 试数据说明：

由php随机生成学生数据，男女性别出现概率相等，各个化数据出现概率相等。实验中，每组数据都进行不少于5次的实验，数据分析以多次实验后得到的均值为准。仿真数据可以从https://github.com/Other-Side/student-Domitory-Distribute/blob/master/simulationData.xlsx中获取，仿真代码可以从https://github.com/Other-Side/student-Domitory-Distribute中获取。 （其他仿真与本实验相同，将不再特别说明）

1. 仿真实验分别将贪心算法与传统算法在分配结果中的不同宿舍匹配度（m=500，n=4，约为东莞理工学院一个学院一个年级的数据规模，宿舍情况）、相同数据规模不同数据多次分配平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 1，图3. 2，图3. 3，图3. 4，图3. 5。
2. 结果分析：

在分配后不同宿舍匹配度比较的仿真实验中（图3. 1），模拟产生500个学生数据，宿舍容量为4，分配结果的125个宿舍中，贪心算法分配的平均匹配度为79.79%，传统分配的平均匹配度为 46.55%。贪心算法中，低于均值80%的宿舍数量有5个，传统分配低于均值80%的宿舍数量有11个。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 2），贪心算法分配结果的平均匹配度为79.29%，标准差为0.38，传统分配算法的平均匹配度为46.30%，标准差为0.65。

在不同数据规模平均匹配度的比较仿真实验中（图3. 3），贪心算法匹配度随着数据规模的增大而逐渐增大，传统分配方法匹配度基本不变，与3.2.3算法分析结果一致。

在不同数据规模运行时间的对比仿真实验中（图3. 4），贪心算法程序运行时间基本呈2次幂函数上升，传统分配方法程序运行时间呈一次线性上升，与3.2.3算法分析结果一致。

在不同数据规模LMRN的对比仿真实验中（图3. 5），贪心算法程序分配结果LMRN略有上升，传统分配方法程序分配结果LMRN呈一次函数上升，与3.2.3算法分析结果一致。



图3. 1一次分配不同宿舍匹配度情况



图3. 2对数据规模m=500的多次仿真匹配度稳定性

图3. 3在不同数据规模下的贪心算法分配与传统随机分配匹配度



图3. 4在不同数据规模下的贪心算法分配与传统随机分配程序运行时间



图3. 5在不同数据规模下贪心算法与传统分配方式的LMRN

## 3.3 k最近邻算法

## 3.3.1 k最近邻算法介绍

k最近邻算法（k-Nearest neighbor,下称KNN），也称k近邻算法。该算法是一个机器学习领域的分类算法，由于时间复杂度低，不需要使用训练集进行训练所以训练时间复杂度为0，不像深度神经网络属于黑箱模型而具有可解释性，对在类域存在交叉或重叠的样本集分类效果优良，所以该算法得到了广泛的推广，并被尝试应用在多个领域。但该算法需指定参数k，是否选择了合适的参数k，直接对该算法的分类结果的效果产生影响。

KNN适用情况为：由一些已知分类的样本来确定一个未知类别的实例的所属类别。指定参数k之后，KNN会统计与该未知实例最近的k个样本点的类别，并按该k个样本点出现概率最大的样本点的类别确定为未知类别的实例的类别。由此可见，KNN中参数k会直接影响分类的结果。

## 3.3.2 k最近邻算法在学生分配宿舍上的应用

算法执行前，需要有已知类别的样本作为KNN的执行样本。在学生宿舍分配中，将学生的宿舍分配建模成学生分类，分类类别数量即为宿舍数量，类内元素数量即为宿舍容量。可随机选择与宿舍数量相同的学生作为分类样本，并依次将待分配宿舍的学生作为实例执行指定参数k为1的KNN算法进行宿舍分配，同时避免一间宿舍人数超过宿舍容量的限制。

假设学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中KNN算法步骤如下。

1. 算法开始，随机选取个学生分别入住不同的宿舍。
2. 在剩下的待分配的学生里面选取一位，计算他/她与已知宿舍的宿舍内学生整体匹配度，挑选最大匹配度的学生宿舍入住。
3. 如果有宿舍人数达到n，即退出参与ii已知宿舍的匹配度计算，该宿舍分配完成。
4. 如果待分配学生数量为0，分配完成，算法结束。
5. 重复执行ii步骤，iii步骤，iv步骤，即可完成KNN算法在宿舍分配上的应用。

## 3.3.3 k最近邻算法分析

该算法执行前需要样本，算法执行前生成的个类别的样本具有随机性，如果该样本平衡，即样本的分布情况与整体分布情况相似，则其宿舍分配效果较好，如果随机生成的样本不平衡，则其宿舍分配效果将较为一般。在算法执行过程中，对于待分配的同学来说，也存在分配先后的随机性，先分配的学生，选择范围大，种类多，选择到匹配度较高的可能大，而后分配的学生，由于大部分宿舍已经安排满人， 可供匹配的宿舍少，匹配到匹配度高的可能性小。由于存在第一批分配学生的随机性以及第二批分配学生先后的随机性，所以此分配算法对同一批待分配的学生可能出现多次分配多种结果，也就是说不同次的执行结果的整体匹配度会有一定的变化。

## 3.3.4 仿真及实验数据分析

1. 时间复杂度：，其中m为学生规模，p为个化属性数量，n为宿舍容量。一般p，n均可认为是常量，所以时间复杂度为。
2. 仿真实验分别将KNN算法与传统算法在分配结果中的不同宿舍匹配度（m=500，n=4，约为东莞理工学院一个学院一个年级的数据规模，宿舍情况）、相同数据规模不同数据多次分配平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 6，图3. 7，图3. 8，图3. 9，图3. 10。
3. 结果分析：

在分配后不同宿舍匹配度比较的仿真实验中（图3. 6），模拟产生500个学生数据，宿舍容量为4，分配结果的125个宿舍中，KNN算法分配的平均匹配度为75.82%，传统分配的平均匹配度为45.97%。KNN算法中，低于均值80%的宿舍数量有4个，传统分配低于均值80%的宿舍数量有14个。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 7），KNN算法分配结果的平均匹配度为76.22%，标准差为0.47，传统分配算法的平均匹配度为46.23%，标准差为0.72。

在不同数据规模平均匹配度的比较仿真实验中（图3. 8），KNN算法匹配度随着数据规模的增大而逐渐增大，传统分配方法匹配度基本不变，与3.3.3算法分析结果一致。

在不同数据规模运行时间的对比仿真实验中（图3. 9），KNN算法程序运行时间基本呈2次幂函数上升，传统分配方法程序运行时间呈一次线性上升，与3.3.3算法分析结果一致。

在不同数据规模LMRN的对比仿真实验中（图3. 10），KNN算法程序分配结果LMRN略有上升，传统分配方法程序分配结果LMRN呈一次函数上升，与3.3.3算法分析结果一致。



图3. 6一次分配不同宿舍匹配度情况

图3. 7对数据规模m=500的多次仿真匹配度稳定性



图3. 8在不同数据规模下的KNN算法分配与传统随机分配匹配度



图3. 9在不同数据规模下的KNN算法分配与传统随机分配程序运行时间



图3. 10在不同数据规模下KNN算法与传统分配方式的LMRN

## 3.4 改进k最近邻算法

## 3.4.1 改进k最近邻算法介绍

K最近邻算法在本章3.3.1中已经进行了介绍，这里主要介绍对3.3中由于产生样本不平衡的情况的优化处理，即改进KNN（New k-Nearest Neighbor,下称NKNN）。在随机生成样本时，即随机先对个学生安排进不同宿舍时，不一定与整体数据的分布情况相似，也就是说，可能在整体数据的分布中，稀疏的位置生成了较多的样本点，而在密度较大的地方生成了较少的样本点。一旦出现这样的情况，密度较大的数据样本点容量达到上限之后，其余数据只能勉强被安排进密度稀疏位置的样本点。改进算法主要在分配策略上来解决这个问题，以尽可能高的提高整体分配的匹配度。分配策略将在下一小节3.4.2中具体介绍。

## 3.4.2 改进k最近邻算法在学生分配宿舍上的应用

NKNN算法执行前，同样需要有已知类别的样本作为NKNN的执行样本。在学生宿舍分配中，将学生的宿舍分配建模成学生分类，分类类别数量即为宿舍数量，类内元素数量即为宿舍容量。可随机选择与宿舍数量相同的学生作为分类样本，并依次将待分配宿舍的学生作为实例执行指定参数k为1的NKNN算法进行宿舍分配，同时避免一间宿舍人数超过宿舍容量的限制。为解决样本不均的情况，分配策略相较于KNN更新为：先分配满的宿舍并不退出待分配学生宿舍的名单，若被待分配学生宿舍匹配到，则将该学生跳过并标记。第一轮分配介绍后，如果当前有宿舍未分配满员，则将剩下的学生与未分配满宿舍的学生整体再做一次NKNN算法，迭代直至所有学生分配完毕。这样避免了样本不平衡导致后分配的学生需向稀疏样本点妥协的情况，如果稀疏处的样本点较多或密度集中部分数据最合适的匹配宿舍已经被先占满，则将该部分学生重新执行NKNN分配。

收敛性讨论。学生宿舍数量为，在的情况下，可以确保每次NKNN都至少有一间宿舍分配满人。即每次执行NKNN，m的规模至少会缩小n。当时，为确保收敛性，只执行一次KNN。算法结束。

假设待分配学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中NKNN算法步骤如下。

1. 算法开始，随机选取个学生分别入住不同的宿舍。
2. 剩下的学生按学号排序。
3. 在剩下的待分配的学生里面按顺序选取一位，计算他/她与已知宿舍的宿舍内学生整体匹配度，挑选最大匹配度的学生宿舍，如果该宿舍已经住满人，则跳过该学生。如果该宿舍没满人，则入住。
4. 重复iii的步骤直到所有都学生参与过匹配。
5. 计算所有宿舍未满的学生的数量与在步骤iii中被跳过的学生的数量，并将此两类学生重新初始化为待分配宿舍的状态，更新m值，如果此时，执行一次KNN，算法结束。否则重复步骤i，ii，iii，iv，v。

## 3.4.3 改进k最近邻算法分析

该算法在一定程度上解决了KNN在样本生成时出现的不平衡性，提高了整体的匹配效果，但是由于每一次迭代都会随机生成样本点，不同的样本点可能会产生不同的分配方案，即对同一批待分配宿舍的学生多次使用NKNN算法进行宿舍分配，可能会有多种分配结果，从而导致目标函数整体匹配度的值会有一定的波动。

## 3.4.4 仿真及实验数据分析

1. 时间复杂度：

，其中m为学生规模，p为个化属性数量。一般p，n均可认为是常量，所以时间复杂度为。

1. 仿真实验分别将NKNN算法与传统算法在分配结果中的不同宿舍匹配度（m=500，n=4，约为东莞理工学院一个学院一个年级的数据规模，宿舍情况）、相同数据规模不同数据多次分配平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 11，图3. 12，图3. 13，图3. 14，图3. 15所示。
2. 结果分析：

在分配后不同宿舍匹配度比较的仿真实验中（图3. 11），模拟产生500个学生数据，宿舍容量为4，分配结果的125个宿舍中，NKNN算法分配的平均匹配度为76.15%，传统分配的平均匹配度为46.08%。NKNN算法中，低于均值80%的宿舍数量有2个，传统分配低于均值80%的宿舍数量有10个。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 12），NKNN算法分配结果的平均匹配度为76.17%，标准差为0.53，传统分配算法的平均匹配度为46.36%，标准差为0.52。

在不同数据规模平均匹配度的比较仿真实验中（图3. 13），NKNN算法匹配度随着数据规模的增大而逐渐增大，传统分配方法匹配度基本不变，与3.4.3算法分析结果一致。

在不同数据规模运行时间的对比仿真实验中（图3. 14），NKNN算法程序运行时间基本呈2次幂函数上升，传统分配方法程序运行时间呈一次线性上升，与3.4.3算法分析结果一致。

在不同数据规模LMRN的对比仿真实验中（图3. 15），NKNN算法程序分配结果LMRN略有上升，传统分配方法程序分配结果LMRN呈一次函数上升，与3.4.3算法分析结果一致。



图3. 11一次分配不同宿舍匹配度情况

图3. 12对数据规模m=500的多次仿真匹配度稳定性



图3. 13在不同数据规模下的NKNN算法分配与传统随机分配匹配度



图3. 14在不同数据规模下的NKNN算法分配与传统随机分配程序运行时间



图3. 15在不同数据规模下NKNN算法与传统分配方式的LMRN

## 3.5基于图论的最小子树生成算法

## 3.5.1基于图论的最小子树生成算法介绍

图论是数学的一个分支，它研究的是由顶点和边组成的一种数学模型，即研究的是图。这种图形通常用来描述某些事物之间的某种特定关系，用点代表事物，用连接两点的线表示相应两个事物间具有这种关系。

最小生成树是图论中的重要概念。在一个图中，它有n个顶点，则只需n-1条边,就可以将其组成一个连通图，在各种组合中，所有n-1条边的权重之和最小的连通图，就是所谓的最小生成树。

近年来，图论在聚类分析中得到了广泛的应用[23]，原因之一是人们可以自然地令聚类分析中的数据单元及其产生的相异度矩阵对应一个赋权完全图，图上点即为数据单元，而图上的边的权值为对应相异度矩阵的值。

## 3.5.2 基于图论的最小子树生成算法在学生分配宿舍上的应用

将所有学生建模成在以个化属性为轴多维坐标系上的点，生成学生相异度矩阵即为每两个点之间的距离，从而可以得到一个加权无向完全图。在该完全图上生成n个最小连通子图，并且要求每个连通子图顶点数相同，使得生成连通子图边长总和尽量小。从而将学生宿舍分配转换成图论应用。

假设待分配学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中基于图论的最小树生成算法步骤如下。

1. 算法开始。
2. 生成学生相异度矩阵
3. 获得相异度矩阵中值最小的两个对象a,b（a!=b）
4. 以a,b为初始解集按prim或迪杰斯彻算法生成顶点数为n的最小生成树，将该生成树所有顶点从相异度矩阵中移出，并保存为同一个组。
5. 重复iv直至所有最小生成树生成。
6. 每次由iv执行完的小组即为宿舍分配小组。

## 3.5.3基于图论的最小子树算法分析

与前面几个算法相比，基于图论的最小生成子树算法具有一定的稳定性，从初始结点的选择到宿舍内成员选择以及顺序都有一定的标准，不存在随机性。如果在prim算法中出现初始解集与待选解集边长一样时，指定一定的选择规则，使其不具有随机性，那么该算法对同一数据集的多次执行结果相同。

同样的，该算法与其他算法存在一样的缺点，就是通常最小相异度处于数据密集区域，从密度大的区域开始分宿舍，分到最后剩下的点数越来越稀疏，使得后者差异度较大，即整体匹配较高，但是存在标准差较大的情况。

同时，该算法中，实际目标为整体宿舍匹配度较高，建模目标为生成子树总边长较短。采用最小生成树可实现尽可能的总边长最短，但是实际中对生成树模型可能“不领情”。其不同点在于，生成树模型有一定的传递性，而宿舍匹配度计算是以两两匹配进行计算的。如a同学与b同学匹配度较高，b同学与c同学匹配度较高，以a，b同学为初始解集生成的最小树为a，b，c同学，而a同学与c同学匹配度不一定很高。特别是当数据规模较少，而宿舍人数较多时，该缺点更为突出。为解决这个问题，可考虑将最小生成树模型修正为，最小完全图生成模型。

为提高程序运行效率，算法设计中采用先生成相异度矩阵再遍历生成最小树，这种使用提高空间复杂度从而降低时间复杂度的做法，可能不适用于数据规模较大，而硬件配置中内存较小的运算。

## 3.5.4 基于图论的最小子树算法仿真及实验数据分析

1. 时间复杂度：，其中m为学生规模， p为个化属性数量。一般p可认为是常量，所以时间复杂度为。
2. 测试数据说明：

由php随机生成学生数据，男女性别出现概率相等，各个化数据出现概率相等。实验中，每组数据都进行不少于5次的实验，数据分析以多次实验后得到的均值为准。仿真数据可以从https://github.com/Other-Side/student-Domitory-Distribute/blob/master/simulationData.xlsx中获取，仿真代码可以从https://github.com/Other-Side/student-Domitory-Distribute中获取。由于仿真实验运行的机器内存大小限制，实验只进行到数据规模为2000的仿真，对于其他使用相异度矩阵以降低程序运行时间的算法同理，不再声明。

1. 仿真实验分别将图论算法与传统算法在分配结果中的不同宿舍匹配度（m=500，n=4，约为东莞理工学院一个学院一个年级的数据规模，宿舍情况）、相同数据规模不同数据多次分配平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 16，图3. 17，图3. 18，图3. 19，图3. 20所示。
2. 结果分析：

在分配后不同宿舍匹配度比较的仿真实验中（图3. 16），模拟产生500个学生数据，宿舍容量为4，分配结果的125个宿舍中，图论算法分配的平均匹配度为77.83%，传统分配的平均匹配度为45.97%。图论算法中，低于均值80%的宿舍数量有8个，传统分配低于均值80%的宿舍数量有14个。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 17），图论算法分配结果的平均匹配度为77.42%，标准差为0.58，传统分配算法的平均匹配度为46.45%，标准差为0.60。

在不同数据规模平均匹配度的比较仿真实验中（图3. 18），图论算法匹配度随着数据规模的增大而逐渐增大，传统分配方法匹配度基本不变，与3.5.3算法分析结果一致。

在不同数据规模运行时间的对比仿真实验中（图3. 19），图论算法程序运行时间基本呈2次幂函数上升，传统分配方法程序运行时间呈一次线性上升，与3.5.3算法分析结果一致。

在不同数据规模LMRN的对比仿真实验中（图3. 20），图论算法程序分配结果LMRN略有上升，传统分配方法程序分配结果LMRN呈一次函数上升，与3.5.3算法分析结果一致。



图3. 16一次分配不同宿舍匹配度情况



图3. 17对数据规模m=500的多次仿真匹配度稳定性



图3. 18在不同数据规模下的图论算法分配与传统随机分配匹配度



图3. 19在不同数据规模下的图论算法分配与传统随机分配程序运行时间



图3. 20在不同数据规模下图论算法与传统分配方式的LMRN

## 3.6 k-means聚类算法

## 3.6.1 k-means聚类算法介绍

k-means算法是无监督的聚类算法，它实现起来简单，聚类效果优良，并且由于其强大的可扩展性和出色的速度，因此被广泛应用[24]。k-means算法是一个重复移动类中心点的过程：把类的中心点，移动到其包含成员的平均位置，然后重新划分其类内部成员。k-means可以自动分配样本到不同的类，但是不能决定究竟要分几个类，类的数量（即k值）需要由用户决定。

k-means++是k-means的优化版本，其在选取k个初始中心点时，先对每个数据单元进行密度计算，从而挑选更为合适的中心点，提高k-means的收敛速度。

k-medoids也是k-means的一种衍生版本，其与k-means的区别在于，每次选取类中心点时不以类内数据中心（也称重心）为类的中心点，而是选取最距离中心最近的数据单元做为类中心点。这种做法可以在一定的程度上削弱异常值的影响，但是计算较为复杂，耗费的时间比较多。

在本节的应用上，由于对原算法做了较大的改动，将不再区分各算法具体区别，统称k-means算法。

## 3.6.2 k-means聚类算法在学生分配宿舍上的应用

对学生根据个化属性进行聚类，但是聚类结果类内元素数量并不一定恰好与宿舍容量大小相同，作者对该情况处理方式如下。聚类后，类内元素数量大于或宿舍容量的，随机选取宿舍容量整数倍的学生安排到宿舍中，剩下学生与聚类后类内元素数量小于宿舍容量的学生一同使用贪心算法进行宿舍安排。

假设待分配学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中k-means聚类方法算法步骤如下。

1. 算法开始。
2. 随机选取个学生作为个类的初始中心点。
3. 根据类内元素（学生）更新类的中心点。
4. 根据心的中心点，将所有元素重新划分类别。
5. 重复ii步骤，iii步骤，直到划分情况基本不再变化。
6. 将聚类后类内元素大于n的类随机取出最大n的整数倍的元素进行宿舍安排。将所有剩余学生运行贪心算法进行宿舍安排。
7. 宿舍分配结束。

## 3.6.3 k-means聚类算法分析

本节中，用贪心算法解决聚类后类内元素数量与宿舍容量不相同的情况，可根据实际选择其他算法（如KNN）解决这个情况。

本解决方案先将学生进行一次聚类分配宿舍，再将剩余的部分学生进行贪心算法分配宿舍，由于剩余部分学生数量小于或等于学生总数量的二分之一，在数据规模较大时，可以极大的减少程序运行时间。同时，相比于直接使用贪心算法，此解决方案对匹配后的平均宿舍匹配度影响较小。

由于聚类后对所有类内元素大于宿舍容量的聚类进行随机选取，可能造成部分宿舍匹配度不高，这部分宿舍匹配度波动较大。对剩下其他学生使用贪心算法，也具有贪心算法的缺点，如最后几个贪心生成的宿舍匹配度较低。

## 3.6.4 k-means聚类算法仿真及实验数据分析

1. 时间复杂度：，其中m为学生规模，p为个化属性数量。一般p可认为是常量，所以时间复杂度为。
2. 仿真实验分别将图论算法与传统算法在分配结果中的不同宿舍匹配度（m=500，n=4，约为东莞理工学院一个学院一个年级的数据规模，宿舍情况）、相同数据规模不同数据多次分配平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 21，图3. 22，图3. 23，图3. 24，图3. 25所示。
3. 结果分析：

在分配后不同宿舍匹配度比较的仿真实验中（图3. 21），模拟产生500个学生数据，宿舍容量为4，分配结果的125个宿舍中，图论算法分配的平均匹配度为75.25%，传统分配的平均匹配度为45.97%。图论算法中，低于均值80%的宿舍数量有5个，传统分配低于均值80%的宿舍数量有14个。前约半数部分为从k-means聚类后类内元素数量大于或等于宿舍容量的类中随机取出的学生，可以看到此部分较中后部分波动较大，可以考虑在选取时，再次使用贪心算法提高分配后各宿舍匹配度的稳定性，即k-means聚类结果用来分割进行贪心算法的数据集，降低数据规模，从而提高整体程序的运行速度。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 22），k-means算法分配结果的平均匹配度为74.96%，标准差为0.66，传统分配算法的平均匹配度为46.20%，标准差为0.56。k-means算法分配结果匹配度优于传统分配方式的匹配度，两者分配结果稳定性相近。

在不同数据规模平均匹配度的比较仿真实验中（图3. 23），图论算法匹配度随着数据规模的增大而逐渐增大，数据规模为2000时，匹配度为79.37%，传统分配方法匹配度基本不变，与3.6.3算法分析结果一致。

在不同数据规模运行时间的对比仿真实验中（图3. 24），k-means算法程序运行时间基本呈2次幂函数上升，传统分配方法程序运行时间呈一次线性上升，与3.6.3算法分析结果一致。

在不同数据规模LMRN的对比仿真实验中（图3. 25），两种分配方式分配结果LMRN略有上升，但传统分配方式LMRN上升更快，根据定义可知，LMRN值越小分配结果低匹配度的宿舍越少，故k-means更优。



图3. 21一次分配不同宿舍匹配度情况



图3. 22对数据规模m=500的多次仿真匹配度稳定性



图3. 23在不同数据规模下的kmeans算法分配与传统随机分配匹配度



图3. 24在不同数据规模下的kmeans算法分配与传统随机分配程序运行时间



图3. 25在不同数据规模下kmeans算法与传统分配方式的LMRN

## 3.7 蚁群聚类算法

## 3.7.1 蚁群聚类算法介绍

蚁群聚类是蚁群算法的一种，也是群智能算法的一种[25]。其生物原型为蚁群的蚁穴清理行为：蚁群中的工蚁会将蚁穴中分布分散的蚂蚁尸体堆积成相对集中的几个大堆。基本机制是蚁尸对于工蚁搬运具有一定的吸引力，尸蚁所处位置附近没有其他蚁尸（蚁尸密度低），则工蚁搬起该蚁尸的概率大。反之则反。搬起蚁尸后，工蚁随机移动，随时可能把身上的蚁尸放下，放下的概率由此时工蚁所处位置决定。此时工蚁所处位置蚁尸较多，则放下概率较大，所处位置蚁尸较少，则放下概率较小。通过这样的搬运方式移动蚁尸形成的蚁尸堆，蚁尸堆越大，越吸引工蚁将死蚂蚁堆积到该堆，使得蚁堆的规模越来越大，由此而形成了一个正反馈。

该算法于1991年被DeneubourgJL提出，而后LumerE和FaietaaB将该模型应用到了数据分析的范畴[26][27] [28]，也就是LF算法。核心思想是先将待聚类的数据对象按其属性值摆放在一个坐标轴下，然后在该范围内产生一些虚拟的蚂蚁对其进行聚类分析。这些蚂蚁随机选择一个数据对象，按照该数据对象附近区域的情况随机的不等概率的对该对象执行搬运，放下，不理会这几种操作的一种。经过这个过程的不断迭代，散步的数据就按照其相关性而聚集成几个类。

## 3.7.2 蚁群聚类算法在学生分配宿舍上的应用

由于蚁群聚类从“蚂蚁”对一个数据点的拾起到对一个数据点的放下都具有随机性，无法控制聚类后类内元素数量。作者在本节中采用与蚁群聚类同样的算法思想，但不完全一致的解决方案。具体为，将算法开始前的初始状态置为按类内元素数量相同的原则随机分类，然后算法执行时根据帕累托最优（Pareto Optimality, 也称为帕累托效率,Pareto efficiency）原则对数据点进行搬动。帕累托最优是指资源分配过程，从一种分配状态到另一种状态的变化中，在没有使任何单位境况变坏的前提下，使得至少一个单位变得更好。具体为，根据一定概率挑选一个类，再根据一定概率挑选该类内一个元素，搬起该元素，在数据点的空间内随机移动，按一定概率放下，如果选择放下则再从该类中随机将一个该类中的其他元素搬回原类中。判断该操作是否符合帕累托最优原则。如果符合则优化成功，进行下一次优化。如果不符合则撤销本次操作，重新尝试下一次搬动。每个类被挑选到的概率比为每个类的值之比，PUR计算方式如下：，其中为该类匹配度值，minMR及maxMR由全类得到，这样可保证较小匹配度的宿舍有较大的被选择的机会。类内元素挑选方式与类挑选方式相似，不再赘述。通过上述概率选择，提高蚂蚁正确选择的机会，避免重复做无用功。

假设待分配学生数量m，宿舍容量n，即将m个学生安排到容量为n的宿舍中，并使分配方案尽量达到模型中目标函数值。学生宿舍分配中蚁群聚类方法算法步骤如下。

1. 根据指定参数q，即按q指定优化次数。
2. 算法开始。
3. 按随机分配的方式分配所有宿舍。
4. 模拟工蚁的操作，随机选择两个类，再随机将其中不同类的两个元素进行交换，判断是否符合帕累托最优原则，如果不符合，则撤销该操作。
5. 判断iv步骤执行次数，如果次数达到了q，则算法结束，否则重复iv步骤

## 3.7.3 蚁群聚类算法分析

从表面上看，只要q足够大，每个数据单元被尝试的次数足够多，蚁群聚类最终能达到全局最优的效果。但是由于搬运操作的局限性，工蚁每次只能对换两个类中的元素，很容易陷入类似死锁的局部最优，假设3个类中的不同类的3个元素依次传递能达到更高的匹配度，但是两两之间相互交换并不能提高匹配度，那么无论工蚁如何尝试，都无法达到全局最优。可以将工蚁搬运操作进一步优化来解决这个问题。

由于这个聚类过程是一个模拟蚁群工作的过程，可以通过线程方式并发实现多只工蚁进行聚类分析过程，大大提高程序运行速度。

并且此聚类算法并没有一个严格的算法中止条件，程序可以实现随时停止，查看当前宿舍的分配状态，也可以对已经分配好的宿舍进行蚁群聚类进一步提高匹配度。对于宿舍分配的工作人员来说，蚁群聚类相比于其他解决方案有极大的可操作性。

但是蚁群聚类算法中，工蚁长期处于盲目尝试状态，特别是当宿舍匹配度进一步提高时，工蚁所做的无用功比例也随之增加，越来越难找到符合条件的数据对象进行交换。

## 3.7.4 蚁群聚类算法仿真及实验数据分析

1. 时间复杂度：，其中q为工蚁迭代次数，p为个化属性数量。一般p可认为是常量，所以时间复杂度为，于学生数量无关。
2. 仿真实验分别将蚁群聚类算法与传统算法在相同数据规模不同数据多次分配结果的平均匹配度（m=500，n=4）、不同数据规模平均匹配度、不同数据规模运行时间等方面进行对比，实验结果如图3. 26，图3. 27，图3. 28所示。
3. 结果分析：

在不同数据规模的匹配度比较中（图3. 26），蚁群算法参数q=20000及q=200000的分配结果对比随机匹配的匹配度有了明显的提升。且蚁群算法中q=200000的匹配度高于1=20000的匹配度。这与3.7.3节中分析蚁群算法可对分配结果进行优化的分析一致，且随着q的增大，匹配度将逐渐趋近最大化。

在不同数据规模运行时间的对比仿真实验中（图3. 27），蚁群聚类算法程序q=200000的运行时间远高于q=20000的运行时间，单随着数据规模的增大运行时间稍有降低。

在相同数据规模不同数据多次分配平均匹配度的比较仿真实验中（图3. 28），蚁群聚类算法q=200000分配结果的平均匹配度为54.15%，标准差为0.59，传统分配算法的平均匹配度为46.52%，标准差为0.54。两者分配结果稳定性相近。



图3. 26不同数据规模不同参数q的匹配度比较



图3. 27不同数据规模的运行时间比较



图3. 28多次仿真稳定性比较

# 第4章 仿真结果分析

## 4.1 算法仿真

笔者一共尝试了6种解决方案来解决学生宿舍分配问题，各算法适用情况及优缺点不尽相同。其中由于蚁群聚类算法的特殊性，实验数据不具有实际参考意义，因而不参与前5种算法的比较。

在固定数据规模的多次仿真实验中，每个算法都执行了不少于30次的实验，并记录保存每次执行结果。在不同数据规模的多次仿真实验中，每个算法都执行了不少于10次的实验，并记录保存每次执行结果。由于硬件设备限制，k-means算法和图论算法无法进行规模m=5000的仿真，最大数据规模为m=2000。

## 4.2 仿真结果分析

在不同数据规模不同算法匹配度的对比中（图4. 1），贪心算法在数据规模较小时基本与图论算法匹配度持平，随着数据规模增大，贪心算法匹配度增大量大于图论算法，逐渐与其拉开差距，而全程高于其他算法。在数据规模较小时，NKNN算法匹配度略低于KNN算法，但随着数据规模增大，两算法差距逐渐缩小直至追平甚至反超。K-means算法在数据规模较小时表现尚好，但随着数据规模增大，逐渐落后。如果仅考虑匹配度，则数据规模较小时可以选择贪心算法或图论算法，当数据规模较大时，应该选择贪心算法。

在不同数据规模不同算法运行时间的比较中（图4. 2），各算法运行时间都随着时间的增大而增大，其中贪心算法与NKNN算法运行时间基本持平，而KNN算法运行速度较快。如果仅考虑运行时间，可以选择KNN算法，运行时间约为贪心算法的1/3。

在不同数据规模不同算法LMRN值的对比中（图4. 3），各算法LMRN值都随着数据规模增大而增大，其中k-means增长速度最快。在数据规模较小时，KNN算法表现较好，随着数据规模的增大，NKNN算法表现超过了KNN算法。LMRN值表示分配后不同宿舍匹配度的平衡情况，LMRN值越大，则表明低于平均匹配度80%的宿舍数量越多，不同宿舍匹配度越不平衡。如果仅考虑LMRN值，可以选择KNN算法或NKNN算法，不建议选择k-meanas算法。

在不同宿舍匹配度标准差的比较中，比较结果基本与LMRN值相同，KNN及NKNN算法分配后不同宿舍的匹配度差距较小。但此比较中加入了蚁群聚类算法，由图可见，蚁群聚类算法分配后各宿舍匹配度最平衡，不会出现两个极端的现象。



图4. 1不同算法不同数据规模匹配度比较



图4. 2不同数据规模不同算法运行时间比较



图4. 3不同数据规模不同算法LMRN值对比



图4. 4分配后不同宿舍匹配度标准差

## 总结与展望

图4. 5

本论文主要从学生偏好量化、宿舍匹配度建模，宿舍分配算法设计，算法仿真比较等几个方面详细介绍了宿舍分配自动化、智能化的方法。

在系统实现

数据库的设计根据本系统的五大模块的功能进行设计，对业务中涉及到的实体进行逻辑上的属性设计，并且根据现实需求业务设计实体之间的关联。自动形成数据库创建。

（3）详细设计实现

系统由基础功能模块 个模块包含了处理其对应业务的功能，满足。本系统通过权限管理的方式实现用户功能的划分性。

在本文对房能从以下几个方面展开进一步的研究，完善系统的设计和实现。

（1）系统功能完善

可以针对实际需求完善功能细节，提高系统的可用性 修改前端的信息显示样式等。

（2）系统性能完善

在系统使用人数和使用频率较高的情况下，就需要对系统的性能进行完善，从后台处理逻辑设计、服务器支持等方面完善系统设计，保证系统正常运行。

展望：

算法选择，各种聚类算法

算法结合，蚁群

并行计算，计算时间

应用领域旅行商问题、二次分配问题、车辆调度问题

数据可以更完善，k值选取，蚁群次数，实验误差

# 参考文献

1. 教育部印发《教育信息化十年发展规划（2011-2020年）》[EB/OL]. <http://old.moe.gov.cn//publicfiles/business/htmlfiles/moe/s5892/201203/133322.html>
2. 在中国共产党第十九次全国代表大会上的报告《习近平：决胜全面建成小康社会 夺取新时代中国特色社会主义伟大胜利》[EB/OL]. <http://www.xinhuanet.com//politics/19cpcnc/2017-10/27/c_1121867529.htm>
3. 钱揖斌,赵建岭,刘建光,张向东.基于智慧协同的学生公寓管理与服务——中国矿业大学智慧公寓建设实践[J].高校后勤研究,2017(05):23-26.
4. Markey P M , Funder D C , Ozer D J . Complementarity of interpersonal behaviors in dyadic interactions.[J]. Pers Soc Psychol Bull, 2003, 29(9):1082-1090.
5. Markey P M , Kurtz J E . Increasing acquaintanceship and complementarity of behavioral styles and personality traits among college roommates.[J]. Pers Soc Psychol Bull, 2006, 32(7):907-916.
6. 余启林. 考虑学生偏好的高校公寓分配管理系统研究[D].北京交通大学,2017.
7. 苏明杰. 基于线性规划模型和蚁群算法的智能排课系统[D].武汉科技大学,2011.
8. 王巧巧,韦澍芃. 基于贪心算法的学生宿舍分配系统设计与实现[J]. 计算机光盘软件与应用, 2013, 16(10): 147-148.
9. 郝鹃,路美秀. 大学生个性化宿舍分配算法研究[J]. 信息通信, 2016, 卷缺失(5): 153-154.
10. 李巍巍,张玉兰,姚晶宏,胡秦,牛玉杰,马雪真.大学生宿舍人际关系现状及影响因素的研究[J].商业经济,2014(21):102-103.
11. 曹雪雪. 基于学生个性的宿舍分配管理系统设计与实现[D].杭州电子科技大学,2018.
12. 何饶依. “90后”大学生宿舍人际关系研究[D].武汉工程大学,2013.
13. 郑志. 大学生宿舍生活事件量表的初步构建[D].南方医科大学,2010.
14. 邵煜,谢颖华.协同过滤算法中冷启动问题研究[J].计算机系统应用,2019,28(02):246-252.
15. 杨兴雨,李华平,张宇波.基于聚类和随机森林的协同过滤推荐算法[J].计算机工程与应用,2018,54(16):152-157.
16. 路秀英,崔兴凯,霍新丽.求解多目标资源分配问题的改进蚁群优化算法[J].微电子学与计算机,2011,28(10):87-90.
17. Vathsavayi, S.,Sievi-Korte, O.,Koskimies, K.,Systa, K.. Using Constraint Satisfaction and Optimization for Pattern-Based Software Design[P]. Software Engineering Conference (ASWEC), 2014 23rd Australian,2014.
18. 刘彤. 解多目标优化问题的进化算法[D]. 西安电子科技大学, 2010.
19. 梅舒. 智慧教育云平台中多约束分配问题的算法设计与研究[D]. 2016.
20. 曹雪雪.基于贪心算法的智能宿舍分配方法[J].计算机与现代化,2018(01):23-26+31.
21. 王文发,马燕,李宏达.基于矩阵存储的回溯算法在多约束分配问题中的应用探讨[J].西南民族大学学报(自然科学版),2008(05):935-939.
22. 孙灿,苗南南.改进贪心算法在智能宿舍分配系统中的应用[J].信息通信,2014(09):49.
23. 高敬振.图论在聚类分析中的应用[J].数学的实践与认识,1991(03):36-42.
24. 李双虎,王铁洪.Kmeans聚类分析算法中一个新的确定聚类个数有效性的指标[J].河北省科学院学报,2003(04):199-202.
25. 张赛楠. 基于群智能的蚁群聚类算法研究[D].吉林大学,2013.
26. Deneubourg J , Goss S , Franks N , et al. The dynamics of collective sorting: robot-like ants and ant-like, robots.[C]// International Conference on Simulation of Adaptive Behavior. The MIT Press, 1991.
27. Lumer E D , Faieta B . Diversity and adaptation in populations of clustering ants[C]// International Conference on Simulation of Adaptive Behavior: from Animals to Animats 3: from Animals to Animats. MIT Press, 1994.
28. 曾海群. 蚁群聚类算法研究[D].中南大学,2008.

# 致 谢

**本次的毕业设计课题的完成离不开自己的学习和努力，更离不开秦勇老师的悉心指导和不懈支持，也离不开同学们的帮助和支持。**在这次毕业设计中，遇到无数的困难和障碍，但是在导师和同学的帮助下都解决了，让我的毕业设计能在计划中较为顺利地开展。

在学生宿舍分配的算法设计上，侯爱民老师不只在课堂上教会了我图论的相关知识，同时在该课题中给我提供了图论解决方案的指导，让我快速定位问题，解决问题。在学生宿舍分配的算法实现上，与我同课题的搭档陈俊佳同学做出了大量的付出，同时协助我分析算法优缺点与复杂度等问题。在学生宿舍分配的算法选择上，我的小班同学黄耿森也给我提供了灵感。在课题进行中，我的舍友没有对我深夜的影响和打扰提出过抗议，而是一如既往义无反顾的支持我。还有很多很多人，在课题的进行中给我提供过帮助，在此无法一一列举，但我内心均满怀感激。

感谢母校东莞理工学院的培养，感谢计算机与网络安全学院的栽培，感谢帮助过我的老师同学们。经历此次毕业论文我了解了身为一名计算机专业的学生，编写代码的能力虽然是基础，但并不是一切，更重要的是要有整体系统计划思维模式的能力。我会在以后继续完善自身的能力，为学院争光，为自身职业的发展打好基础。

最后，我也非常感激审查论文的老师教授们，感谢您们能够在百忙中抽出时间来指教和审查我的论文。

差点忘了，还要感谢我多年未出现的女朋友，在我课题进行期间从没责备我对她缺少陪伴，甚至连一句打扰都没有。