**数学建模竞赛**

**参赛人信息**

**负责人： 姓名：**

**数学建模竞赛**

题目： 游乐园客流疏导方案

摘 要

本文第一题先根据理想等候时间确定的权值利用蒙特卡洛模拟方式，得到开园时间内任意时刻游乐园内的分布情况。然后按照一定方案进行分流，使得游客平均等候时间最少。第二题运用分组秩的思想，用符号秩检验和多参数线性回归对预定数进行预测

第一问方案分为两个步骤：一个是游乐园游客模拟，另一个是游客疏导分流。

游乐园游客模拟首先需要假设游乐园入园人数随时间变化的散点图，对指定的一个时刻，对于每一个游客以标准权值按照蒙特卡洛方法进行模拟，获取游客信息表，记录游客入园、出园、到每个项目点开始排队、从这个项目点离开的时刻和历史去过的项目点信息。

游客疏导分流首先由游客信息表按照一定方法计算出指定时刻每个项目点的人数情况。然后对指定时刻的游客考虑到至分流结束的时间段内游客人数的变化情况，按照一定方式进行分流，以达到游客平均等候时间最小。

由于后面时间段对游客的疏导分流方案与前面的重复，所以本文只讨论第一个时间段的游客疏导分流方法（需同时用到第一时间段结束时刻和后面某个时刻的游客模拟情况）。现在针对第一时间段结束后时刻的等待游客进行分流。

记录全部分流到位的时刻，首先得到游客的模拟分布。按照时刻实际权值分别计算每一批人对应的最优项目点。再分别计算如果这批人去最优项目点能节省的总时间（一个人节省的时间这批人的人数）。按照节省总时间进行排序，最多者优先分流。分流完第一批，产生新的实际权值，按照之前的方法分流第二批，直至分完，于是得到了时刻的最优分布。以此类推，得到最终的完整优化方案。

第二问，酒店每日的预定数假设和季节、工作日/周末、法定假日、暑期等（称为影响因子）都有关。通过对不同影响因子分组（例如春夏秋冬分为1、2、3、4组等），再将酒店预定数（已剔除掉病态数据）在每组内的顺序按照从小到大排列，就可以得到预定数组内的秩，通过组间相对位置秩的比较就可以确定不同影响因子的重要性（此结果可由影响因子与酒店预订数的相关分析验证），剔除不重要的影响因子后，用符号秩检验和多参数线性回归就可以预测2016年1月至3月每天的酒店预定数目。

关键词：蒙特卡洛 标准权值 实际权值 秩 相关分析 符号秩检验

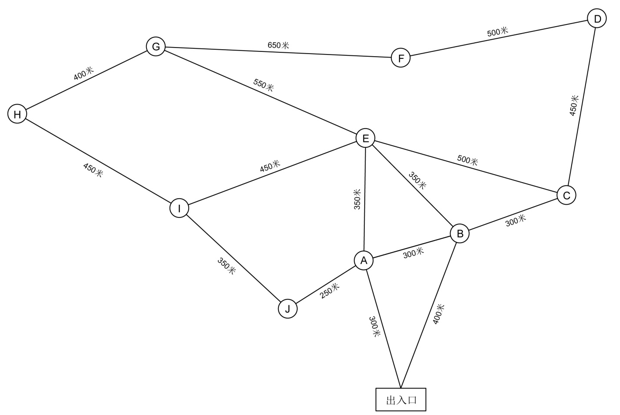
多参数线性回归

**1 问题重述**

Youth游乐园即将开园，预计届时园区将迎来每天1万的大客流。根据客流情况，给出对人群分流的方案，为游客提供游园线路引导，线路的设置以游客的游园体验优先考虑。

（1）图为Youth乐园的规划图，共设A-J共10个项目点，可行的线路图中已经标示出。在保障每位游客体验游乐设施的前提下，建立对每个游乐项目的等候游客进行游览提醒和疏导的模型，以达到游园体验(我们将给出游园体验的定义）最优。每个游乐项目安排请参见表1。

（2）皇冠假日酒店为有需要的游客提供住宿便利。根据该酒店2015年预订数据信息,综合考虑影响房间预定量的主要因素(比如季节,工作日/周末,法定假日,暑期等)建立数学模型。并根据酒店2015年全年预定数据(附件2),预测2016年1月至3月每天预定房间数。



**2 问题分析**

第一题：

题目要求：游客游园体验最优。

方式：建立疏导分流模型。

分流对象：每个项目点排队等候游客。

问题：

1. 游客游园体验最优的数学指标是什么？
2. 题目只给了一个每日预计客流量为10000，怎么模拟每日游客游览情况？
3. 疏导分流的模型怎么建立？

大致解决方案：

1. 定义一个“平均游客满意度 D”：游客总人数与所有游客排队等待时间的总和的比值，D越大，游客体验最优。
2. 我们先设计一个入园人数随时间变化的散点图。对某一个时间段内入园的一群人中的每一个人采用蒙特卡洛方法按照一定权值进行模拟。得到每个时刻游客在每个项目点的分布情况。
3. 游客疏导分流模型首先得到指定时刻项目点游客分布情况，然后对指定时刻的游客考虑到至分流结束的时间段内游客人数的变化情况，按照一定方式进行分流，以达到游客平均等候时间最小。

第二题：

问题2主要涉及一个数据分析的问题，首先要统计好已知日期下每日酒店的预定数，剔除掉病态数据且假设每种影响因子独立的情况下，逐个对影响因子进行分析。因为影响因子与时间相关，所以不能采用ARIMA预测模型，因此本文采用符号秩检验和多参数线性回归进行预测。

**3 概念和符号声明**

1) 定义游客体验最优为排队时间最短，且不重复游玩同一个项目

2）p为游客所在项目点，A B C D E F G H I J G分别对应了游乐园内A点、B点、C点、D点、E点、F点、G点、H点、I点、J点和出入口。

3）项目进行程度T：在t时刻，到项目结束所需的时间段。在一定假设下，T是关于项目点每场持续时间和时刻t的函数，即T=T((p)，t) 。

4）项目点开关：用来记录游客各个项目点游玩情况，对应一个111的矩阵，初值全部设置为1。对于项目点，如果游玩过，则记为0，如果去了出入口，则出入口对应数值改为0，并且之后游客信息不再更新。

5）标准权值Q(p，t：先确定游客所在的项目点p（其中包括出入口），不妨先假设游客在A点，那么B点的标准权值为从A到B路上所需要的时间与在B点理想等待时间和的倒数对应乘以项目点开关。其中在B点理想等待时间只考虑项目进行程度T（，t），不考虑由项目实际等待人数引起的等待时间。Q(p，t每行对应到A B C D E F G H I J K的权值，到所在点权值设为0.

6）实际权值：不妨先假设游客在A点，那么B点的标准权值为从A到B路上所需要的时间与在B点实际等待时间和的倒数对应乘以项目点开关。

7）游客信息表：记录游客入园、出园、到每个项目点玩从排队开始、从这个项目点离开时刻的时刻、和项目点开关，为一个10000的矩阵，即为。

8）游乐园压力度：某一个时刻t，由游客信息表得到每一个项目点当排队人数，由排队人数计算出此项目点所有人玩完这一个项目所要用的总体时间。然后不同项目点所求时间求一个方差。这个方差表示游乐园的压力度。方差越大，游乐园压力度越大；方差越小，游乐园压力度越小。

9）平均游客满意度 D：游客总人数与所有游客排队等待时间的总和的比值（考虑了游客不去重复的地方），D越大，游客体验最优。影响游客体验的因素我们只考虑时间因素，时间因素中主要的是排队等候的时间，排队等候时间越长，游客游园体验越低。我们考虑整体的一个平均排队等待的时间，平均排队等待时间越长，游客平均游园体验最差。于是我们取这个平均排队等待时间的倒数结合已去地方的权值，即平均游客满意度 D，作为衡量游园体验的指标。

10）项目点每场容纳人数N=[400 30 50 30 100 50 30 30 20 50 0]；对应的是 A B C D E F G H I J K 对应容纳的人数。

11）每场持续的时间T0=[33 1.25 2.5 2.5 5 2.5 2 1.5 1.5 2 0]；对应的是 A B C D E F G H I J K 每场对应持续时间。

12）项目点人数分布矩阵 Nr(t，对应为t时刻A B C D E F G H I J K里排队的人数。K点排队人数设置为0 。

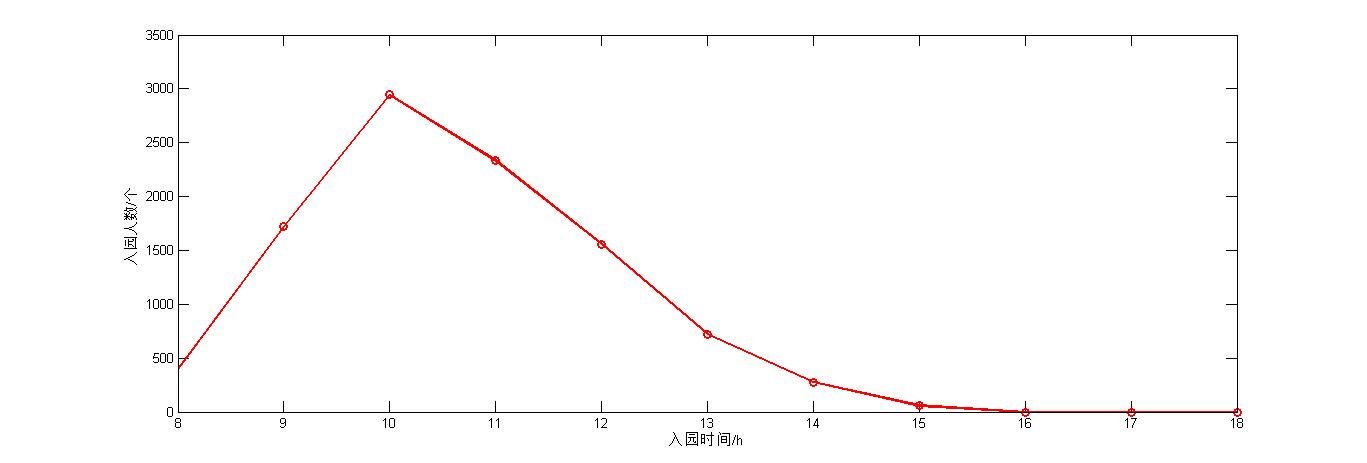
13）第二问中除去病态数据后剩余的天数M天。

**4 模型基本假设**

1）假定游乐园开放时间为 8:0018:00，本实验为了换算方便，开园时间记为0s35999s；

2）假定游乐园每个项目一旦开园立刻进行，并且上一场与下一场间隔不予考虑。也就是说，每个项目点进行的项目进行程度不受每场实际参加人数的影响；

3）游乐园每日入园人数（总入园人数为1万人）随时间变化的散点图可以通过参考文再结合实际情况模拟出，散点图和拟合曲线结果如下：



4）假定每个时间段内进园人数全部是在时间段起始时刻一起进入。例如 8:009:00 间入园人数为389人（见假设3），则认为389人全都是在 8:00 进入；

5）假定游客在某个项目点去另一个项目点的概率遵循标准权值；

6）由于后面时间段对游客的疏导分流方案与前面的重复，所以本文只讨论第一个时间段的游客疏导分流方法（需同时用到第一时间段结束时刻和后面某个时刻的游客模拟情况）；

7）假设所有游客的步行速度为v=1.5m/s ；

8）影响酒店预订的因子相互独立；

9）除了题目中提到的影响因子其它影响因素服从均值为0的正态分布；

10）假设影响因子分组时，同一组内因素无差别（例如工作日的周一至周五等价）；

11）考虑到温度因素，所以定义季节为，1月-3月为春，4月-7月为夏，8月-11月为秋，12月-2月为冬

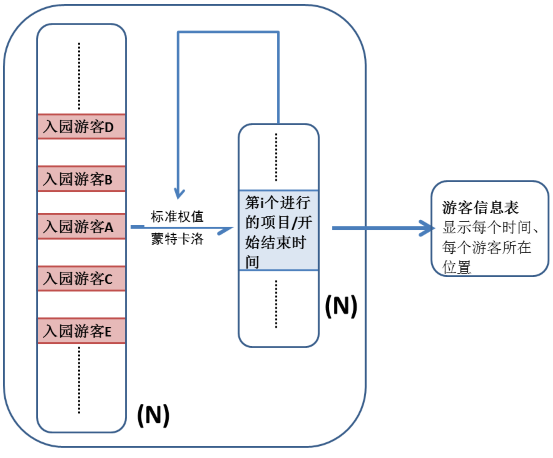
**5 游乐场分流问题（问题1）**

**5.1方案概述**

方案分为两个步骤：一个是游乐园游客模拟，另一个是游客疏导分流。

游乐园游客模拟为了获取游客信息表，针对于每一个游客以标准权值按照蒙特卡洛方法进行模拟，根据游客到某项目点的时刻由游客信息表算出此项目点的排队人数。由排队人数结合每场容纳游客及每场持续时间算出等待时间，再加一场玩的时间，算出游客离开项目点的时刻，对应项目点开关记为0。再以标准权值按照蒙特卡洛方法以进行模拟，依次类推直至，游客离开。此时间段内入园每人同时进行这样的操作。如果存在上一时间段，则上一时间段的人一起考虑。

流程图如下：



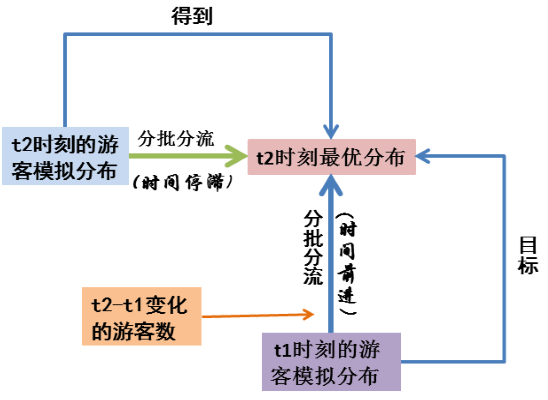
游乐园游客疏导分流目标是保证游乐园压力度最小，分流结束用平均游客满意度进行检验，检验合格则方案可行。游乐园疏导分流是针对第一时间段结束后时刻的等待游客进行分流。分流方案针对的是每批人群，例如A点有1000人排队，则针对三批人（前400、中间400、后200）进行分流。

记录全部分流到位的时刻，首先得到时游乐园压力度最小的游客分布。提取出游客模拟情况，按照时刻实际权值分别计算每一批人对应的最优项目点。再分别计算如果这批人去最优项目点能节省的总时间（一个人节省的时间乘以这批人的人数）。按照节省总时间进行排序，最多者优先分流。分流完第一批，产生新的实际权值，按照之前的方法分流第二批，直至分完。于是得到了时刻的最优分布。

现在对时刻进行分流，期望时刻达到最优分布，其中需要考虑到变化的人数。按照时刻实际权值分别计算每一批人对应的最优项目点。再分别计算如果这批人去最优项目点能节省的总时间（一个人节省的时间乘以这批人的人数）。按照节省总时间进行排序，最多者优先分流。如果分流的那批人的人数大于最优分布需要的人数，则退回多出的人重新作为为一批。再按照时刻实际权值分别计算每一批人对应的最优项目点。再分别计算如果这批人去最优项目点能节省的总时间（一个人节省的时间乘以这批人的人数）。按照节省总时间进行排序，最多者优先分流，直至分完。

验证：比较平均游客满意度d1和分流后平均游客满意度d2，如果d2>d1，则方案成立。

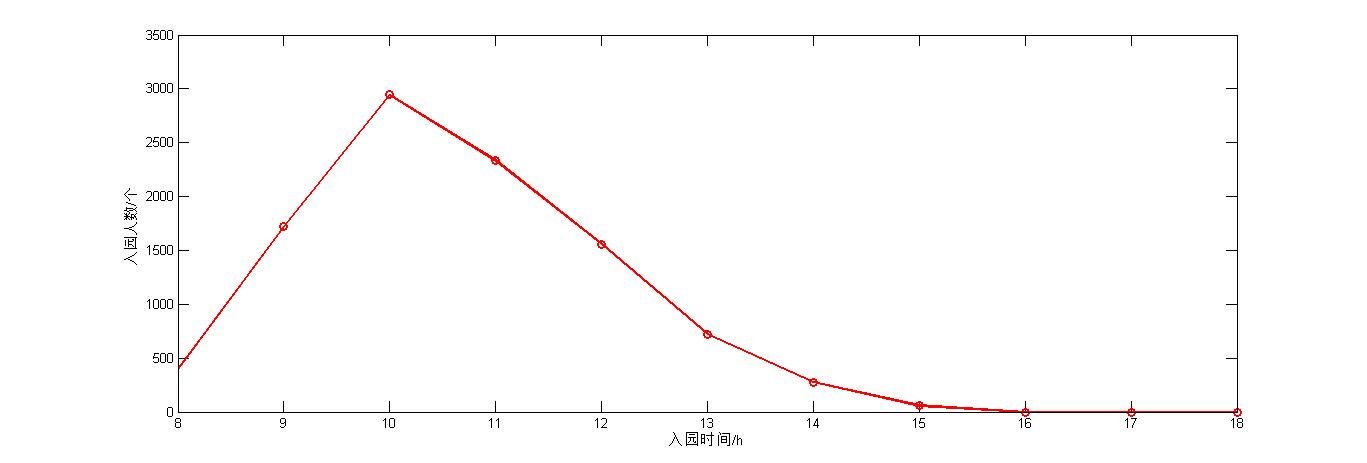
流程图如下：



**5.2游乐园游客模拟**

#### 5.2.1游乐园每日入园人数随时间变化的散点图

首先根据查得文以及结合实际情况，我们先设计一个入园人数随时间变化的散点图。这个散点图需满足所有散点之和达到10000 。本文设计的是：假定开园时间为 8:0018:00 ，为了换算方便，开园时间记为0s35999s 。为了说明的简洁性，我们取3600s （即1小时）为一个时间间隔。假定每个时间段内进园人数全部是在时间段起始时刻一起进入。例如 8:009:00 间入园人数为389人，则认为389人全都是在 8:00 进入。

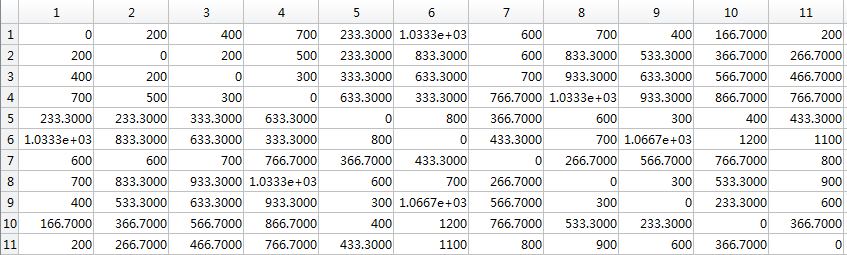


#### 5.2.2在每一点上其他点对应的标准权阵

1）根据所给的路线图，得到每两点之间的距离矩阵。

则时间矩阵=/v：

=



其中1-11对应A-K；

从出发到玩到项目（不考虑排队）用时：

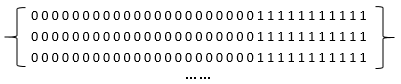
=+ T((p),t)

Q(t)=1/

Q(t)取Q(p，t)的对应行与列的交点。

#### 5.2.3游客信息表建立

对游客信息表初始化



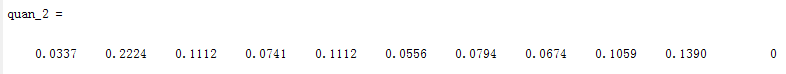
共10000行。每一行对应每一位游客。

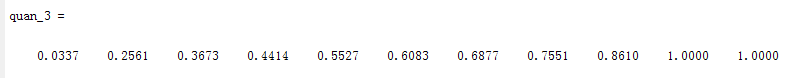
每一行对应数据代表的意思是：

入园时间、到A排队的时间、到B排队的时间、到C排队的时间、到D排队的时间、到E排队的时间、到F排队的时间、到G排队的时间、到H排队的时间、到I排队的时间、到J排队的时间、从A离开的时间、从B离开的时间、从C离开的时间、从D离开的时间、从E离开的时间、从F离开的时间、从G离开的时间、从H离开的时间、从I离开的时间、从J离开的时间、出园时间……

#### 5.2.4首批入园游客蒙特卡洛模拟

游乐园游客模拟为了获取游客信息表，针对于每一个游客以标准权值按照蒙特卡洛方法进行模拟，根据游客到某项目点的时刻由游客信息表算出此项目点的排队人数。由排队人数结合每场容纳游客及每场持续时间算出等待时间，再加一场玩的时间，算出游客离开项目点的时刻，对应项目点开关记为0。再以标准权值按照蒙特卡洛方法以进行模拟，依次类推直至，游客离开。此时间段内入园每人同时进行这样的操作。如果存在上一时间段，则上一时间段的人一起考虑。

初次入园得到的归一化标准权值（matlab模拟）为：

则对389人取随机数，对于任意一个随机数，如果它坐落在下面的任一个区间内，则它被分配去相应的地点。

K

J

I

H

G

F

E

D

C

B

A

1）0s时刻，入园游客389人，则对游客信息表的第1行至389行进行处理。

a）对第1行（即第一位游客），他的起始位置为K点，游客信息表中，入园时间设置为0s。计算标准权值Q（K，0s），用蒙特卡洛方法模拟出他去的下一个点，假定是A点，那么在矩阵中提取O到A所用时间，加上0s，作为到A点的时刻，输入进游客信息表。

（这里需要考虑是否超出假设时间段，本文是3600s，即1小时，超出则后面他放在第二批中一起处理，未超出继续按首批进行处理）。

同理，对后面的388位游客做同样的操作。

b）第1位游客至第389 位游客，统计去A点的人是那几位，这一步可以判断到A时间不为0由一个循环得到。假设共有位，则在位游客中，第i个离开的A时间为 = + T(（p）,t)+（i/N（A）+1）\* （p），数据录入游客信息表，其中i为一个给定整数。（这里需要考虑是否超出假设时间段，本文是3600s，即1小时，超出则后面他放在第二批中一起处理，未超出继续按首批进行处理）

同理统计 B C D E F G H I J 点。

c）这389人的起始位置发生改变，求出对应的标准权值，重复a）、b）步骤。

#### 5.2.5第二、三批入园游客蒙特卡洛模拟

第二批游客入园，在3600s时刻，有1722人入园。

与首批的处理方案类似，但是需要继续考虑前面批次入园游客，即需要考虑 389+1722个人的情况。相关信息录入游客信息表。

**5.3游乐园游客分流**

#### 5.3.1游客分布信息提取

现在我们需要指定时刻t各个项目点游客人数情况。

在游客信息表里，对第1位游客，我们对入园时间、到A排队的时间、到B排队的时间、到C排队的时间、到D排队的时间、到E排队的时间、到F排队的时间、到G排队的时间、到H排队的时间、到I排队的时间、到J排队的时间、从A离开的时间-T(T0（A）,t)、从B离开的时间-T(（B）,t)、从C离开的时间-T(（C）,t)、从D离开的时间-T(（D）,t)、从E离开的时间-T(（E）,t)、从F离开的时间-T(（F）,t)、从G离开的时间-T(（G）,t)、从H离开的时间-T(（H）,t)、从I离开的时间-T(（I）,t)、从J离开的时间-T(（J）,t)、出园时间按时间大小进行排序。看看t排在哪两个数之间，关注角码，如果角码增加，则在数所对应的那个点，如果角码减小，则在路上，我们不考虑。

同理我们对所有游客进行操作，就能得到 Nr（p，t）。

#### 5.3.2游客分布信息提取

我们现在需要对指定的时刻的游客进行分流，期望分流结束时刻，游客的分布能让D取最大值。

1. 时刻的确定。我们首先得到Nr（）。

我们先来看A点。假设A点排队人数为，我们需要对这个人进行分批。因为每场容纳人数限制，第1+k\*N（A）个和第（k+1）\*N（A）位游客情况是一样的（k=0,1,2…）。对于A点从前往后排队的第i个人，他的批次为i/N(A)+1 。

在这第i/N（A）+1批人中，我们看第j个人，来求他的实际等候时间。对应到每一点玩对应的时间矩阵(p=Ts（A）+（Nr（p，t1）/N+1）.\*。对中间元素排序，取出最大的元素。

则我们能得到/N（A）+1个最大值，再在其中取最大值,这个最大值取作 。

这里表示所有批次人群中分配后等待时间最长的。虽然说一批人的分流会影响到后面批次的实际等候时间。但是在追求D最大分配方案下，分配结果的最长等待时间是不会超过的。因此时刻能保证分流基本结束。

1. 得到时刻，D最大的项目点人数分布，即此刻分布为能使游客体验最佳的理想分布。

a）我们先来看A点。假设A点排队人数为XA， A点从前往后排队的第i个人。在这第i/N（A）+1批人中，来求他的实际等候时间。对应到每一点玩对应的时间矩阵(p=Ts（A）+（Nr（p，t1）/N+1）.\*。对中间元素排序，取出最小的元素 min(A，i/N（A）+1)。则这批人分流之后实际节省等候时间(1\*(i/N(A)+1))\*min(A，i/N(A)+1)= (i/N(A)+1)\* N(A)-min(A，i/N(A)+1)。

b）同理算出其他B C D E F G H I J 排队人群中每批人实际节省等候时间。

所有的等候时间进行排序，取最大值对应的那批人进行分配（直接移过去）。

重复a）、b）至分配完全，于是得到了时刻理想分布。

1. 对t1 时刻进行分流，期望到 时刻能得到 时刻的理想分布。

a）我们先来看A点。时刻A点排队人数为Nr（A，），A点从前往后排队的第i个人。在这第i/N（A）+1批人中，来求他的实际等候时间。对应到每一点玩对应的时间矩阵(p=Ts(A).+(Nr(p，t2)/N+1).\*。对中间元素排序，取出最小的元素 min（A，i/N（A）+1）。则这批人分流之后实际节省等候时间 (1\*(i/N(A)+1))\*min(A，i/N(A)+1)= (i/N(A)+1)\* N(A)-min(A，i/N(A)+1)。

b）同理算出其他B C D E F G H I J 排队人群中每批人实际节省等候时间。

所有的等候时间进行排序，取最大值对应的那批人进行分流。

c）比较时刻的理想分布，如果分流过去的人数加上时刻模拟分布那点的人数大于时刻的理想分布那点的人数，则超出的部分返回原位置，并对A重新分批。

d）如果全部返回，则取最大值下一个值对应的那批人进行分流。

重复a）、b）、c）、d）至分流完全。

**6 游乐场分流问题（问题2）**

**6.1病态数据的剔除**

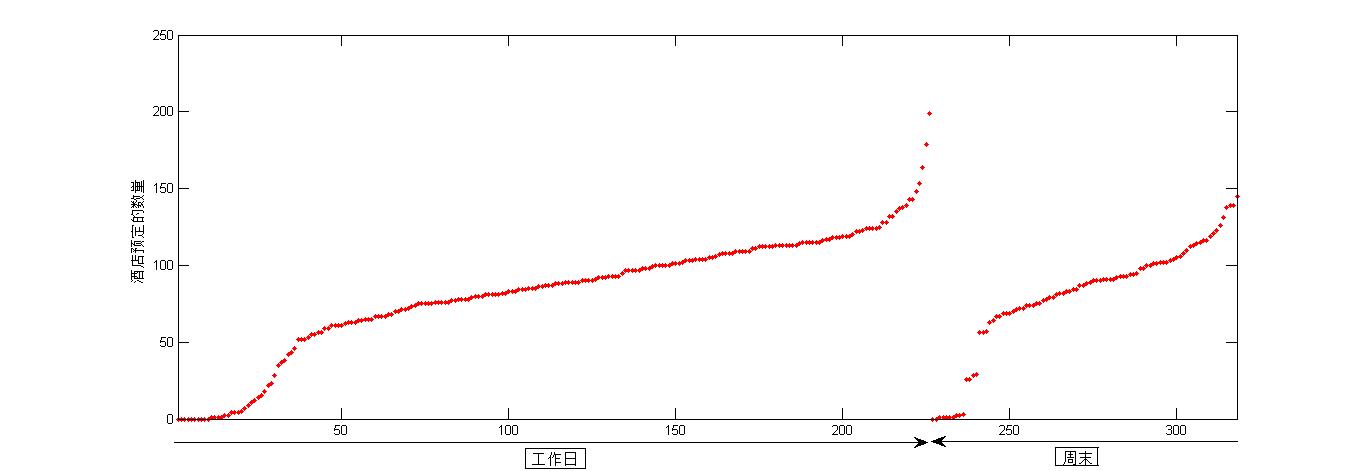
通过2016年年初的房间预订日期可以得到年初很大一批房间数为前一年即2015年年底预订，则2015年年初的预定房间数很大一部分需要在2014年年底，因为2014年年底的数据的缺失，所以可以认为2015年年初的预定数为病态数据，考虑到不能保留病态数据也不能剔除太多有用数据，所以将2015年1月20日（不包括20日）之前的日酒店预订数剔除。

**6.2对数据进行筛选排序统计**

对于不同的影响因子，进行分组，具体分组情况如下：



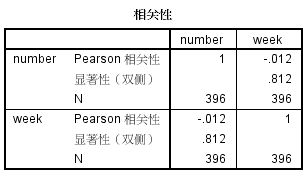
现在对暑期分组情况进行举例说明：

统计有效日期（除去拥有病态数据的日期）中工作日的天数以及是周末的天数。将工作日的类放在一个矩阵的前方，后面跟着周末的类。假设是工作日的天数为N天，则构建一个以为数组[],其中x代表非暑期的酒店预定数，y代表暑期的酒店预定数，后再在不同类中按从小到大排列，结果如图：

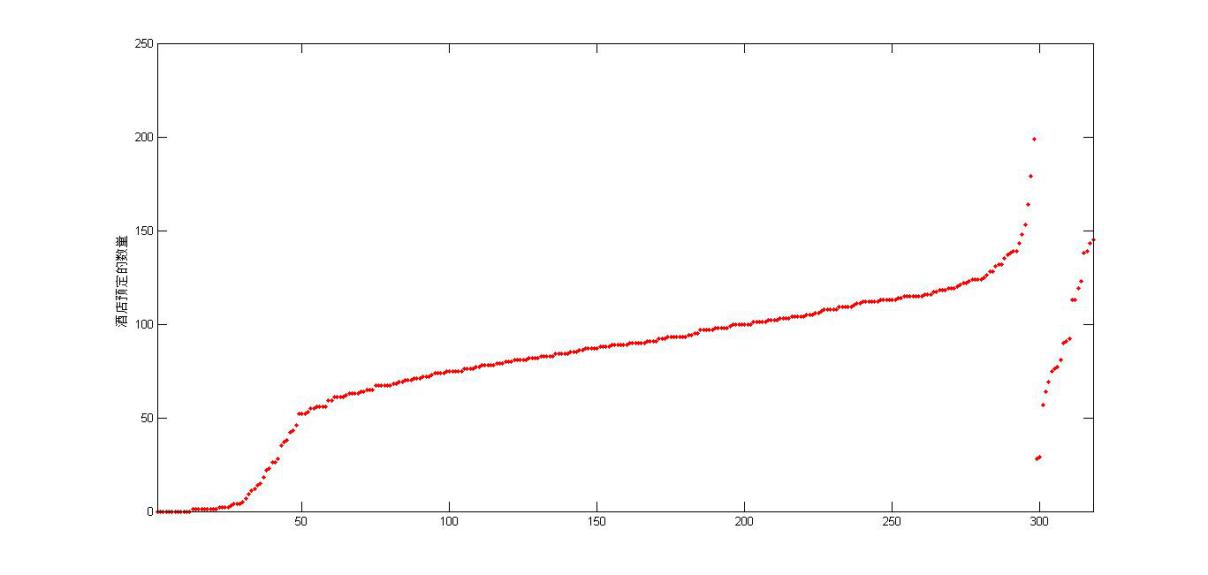
对于两个类求平均，有，

没有显著差别，所以认为工作日和周末对于游乐园酒店预定量没有影响。

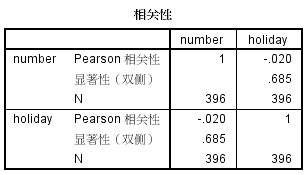
且将是否为周末和酒店预定数做Pearson相关分析，结果如下



相关系数只有-0.012，所以可以认为它们不相关。

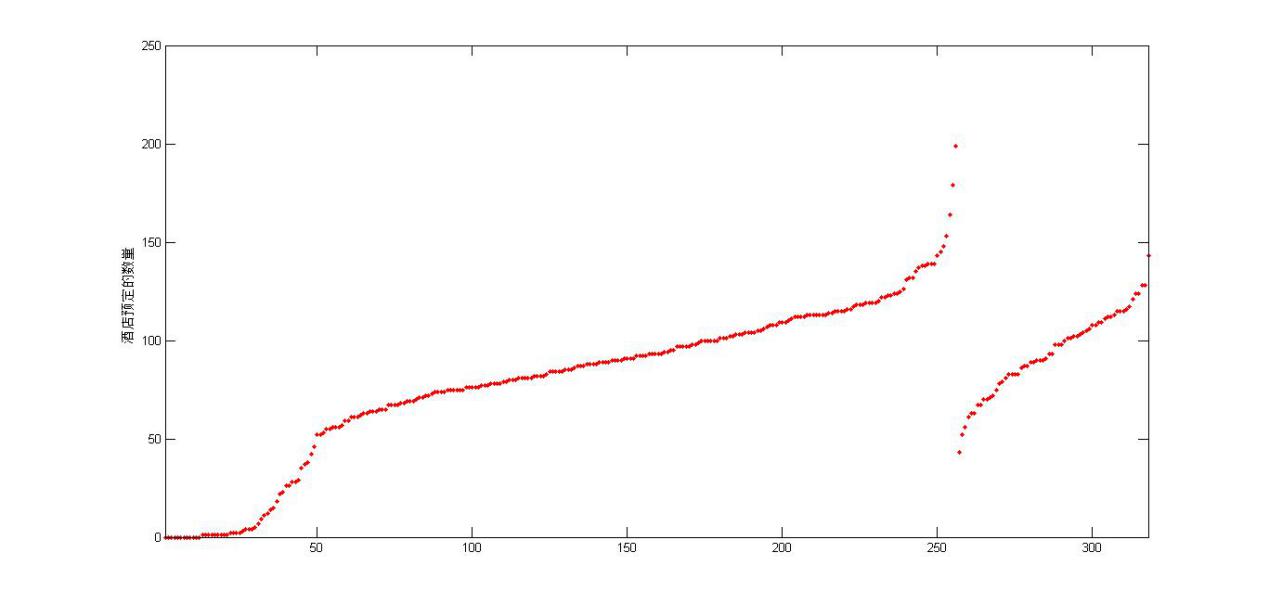
同理可得法定节假日指标如下：

，

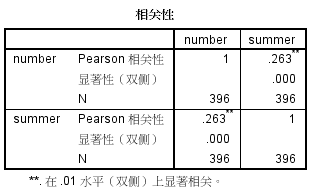


相关系数略微比周末影响因子高。

是否为暑期：

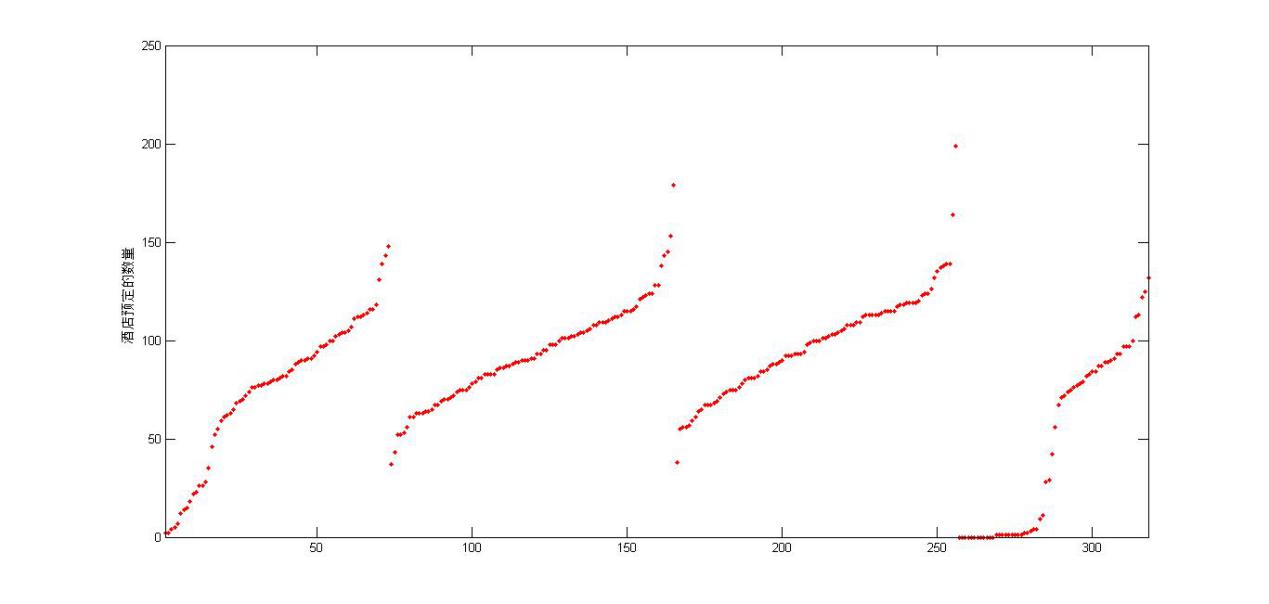


，



相关系数为0.263。

季节：



，



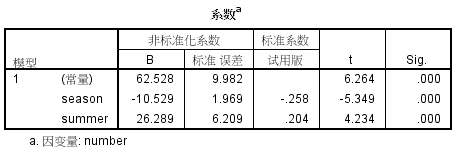
相关系数为-0.305，且很容易看出冬天去的人数远比其它季节的少，这点也很符合实际情况。

**6.3对数据进行预测**

首先对有效的数据进行多参数线性回归，因为由前面的分析知道，周末影响因子和假期影响因子相对于暑期影响因子和季节影响因子都较小，所以此时采样暑期影响因子设为，季节印象因子构建回归方程：

y为模拟的酒店预订数，为一个服从正态分布均值为0的扰动。

得出结果如下：



则有

但其它扰动（不是随即的，是由除了本题其余的影响因素控制的）不知道，所以要用符号秩检验进行预测。

假设为一范围[-s，+s]，运用matlab价格s为0-40中的一个值，对于每一个值进行一次预测。对于暑期影响因子和季节影响因子，其对于酒店预定数的加权平均为81。那么将对于每一个s预测的值与81作比较，大于81的记为1，小于81的记为0，所有的0值和1值服从一个二项分布b（M，1/2），其中M为有效数据量。所以对于每一个s都有最终的概率p。

当时，此时的s即为统计学意义上最优的解。

所以最后得出的预测数据为：

20160101 85

20160102 92

20160103 77

20160104 63

20160105 54

20160106 67

20160107 64

20160108 62

20160109 51

20160110 47

20160111 51

20160112 58

20160113 49

20160114 65

20160115 66

20160116 54

20160117 57

20160118 47

20160119 53

20160120 57

20160121 52

20160122 61

20160123 66

20160124 59

20160125 52

20160126 66

20160127 48

20160128 60

20160129 67

20160130 58

20160131 67

20160201 84

20160202 89

20160203 97

20160204 83

20160205 96

20160206 81

20160207 114

20160208 123

20160209 119

20160210 115

20160211 117

20160212 106

20160213 110

20160214 79

20160215 96

20160216 79

20160217 90

20160218 81

20160219 91

20160220 91

20160221 94

20160222 85

20160223 98

20160224 95

20160225 80

20160226 84

20160227 97

20160228 81

20160229 95

20160230 80

20160301 94

20160302 92

20160303 95

20160304 98

20160305 82

20160306 79

20160307 96

20160308 88

20160309 81

20160310 80

20160311 81

20160312 82

20160313 99

20160314 84

20160315 91

20160316 96

20160317 83

20160318 94

20160319 87

20160320 85

20160321 92

20160322 87

20160323 81

20160324 99

20160325 89

20160326 89

20160327 85

20160328 82

20160329 96

20160330 84

20160331 90

**参考文献：**[1]胡成 城市商业区的客流量聚集等级及时间分布形态判别方法 中国安全科学学报 第24卷第8期 2014年8月

[2]蔡昌俊 基于AFC数据的城轨间客流量分布预测 中国铁道科学 第36卷第1期 2015年1月

[3]邱厌庆 基于九寨沟景点负荷均衡的时空分流导航研究 资源科学 第32卷第1期 2010年1月

[4] 邱厌庆 九寨沟景区初步聚类分流实证研究 资源科学 第32卷第10期

2010年10月

[5]肖熊辉 旅游高峰期景区旅游引力分流调度模型研究-以九寨沟风景区为例

旅游科学 第27卷第6期 2013年12月

**PS:matlab程序**

**1：筛选排序统计程序：**

clear all;

[NUM,TEXT,RAW]=xlsread('data');

date=zeros(6149,0);

for i=1:6149

mid=TEXT{i+1,5};

date(i,1)=str2num(mid);

end

hjm(:,1)=date(:,1);

hjm(:,2)=NUM(:,1);

[NUM2,TEXT2,RAW2]=xlsread('year');

year=zeros(396,2);

for i=1:396

mid=TEXT2{i,2};

year(i,1)=str2num(mid);

end

for i=1:396

for j=1:6149

if hjm(j,1)==year(i,1)

year(i,2)=year(i,2)+hjm(j,2);

end

end

end

% tt=1:1:386;

% last(:,1)=year(11:396,2);

tt=1:1:132;

last(:,1)=year(:,2);

for i=1:132

x(i)=(last(3\*i-2,1)+last(3\*i-1,1)+last(3\*i,1))/3;

end

[NUM3,TEXT3,RAW3]=xlsread('1');

A1=NUM3(79:396,2:3);

for i=1:317

for j=i+1:318

if A1(i,2)>A1(j,2)

mid1=A1(i,1);

mid2=A1(i,2);

A1(i,1)=A1(j,1);

A1(i,2)=A1(j,2);

A1(j,1)=mid1;

A1(j,2)=mid2;

break;

end

end

end %%%226¸ö1

for i=1:225

for j=i+1:226

if A1(i,1)>A1(j,1)

mid1=A1(i,1);

mid2=A1(i,2);

A1(i,1)=A1(j,1);

A1(i,2)=A1(j,2);

A1(j,1)=mid1;

A1(j,2)=mid2;

end

end

end

for i=227:317

for j=i+1:318

if A1(i,1)>A1(j,1)

mid1=A1(i,1);

mid2=A1(i,2);

A1(i,1)=A1(j,1);

A1(i,2)=A1(j,2);

A1(j,1)=mid1;

A1(j,2)=mid2;

end

end

end

a1=mean(A1(1:226,1));

a2=mean(A1(227:318,1))

tt=1:1:318;

plot(tt,A1(:,1),'r.')

axis([1 318 0 250])

ylabel('¾ÆµêÔ¤¶¨µÄÊýÁ¿')

**2：额定权值和初次蒙特卡洛程序：**

t\_la=[1980 75 150 150 300 150 120 90 90 120 0]; %%%%A-J-³öÈë¿Ú³ÖÐøÊ±¼ä

%% A B C D E F G H I J $

T1=1.0e+03 \*[0 0.2000 0.4000 0.7000 0.2333 1.0333 0.6000 0.7000 0.4000 0.1667 0.2000

0.2000 0 0.2000 0.5000 0.2333 0.8333 0.6000 0.8333 0.5333 0.3667 0.2667

0.4000 0.2000 0 0.3000 0.3333 0.6333 0.7000 0.9333 0.6333 0.5667 0.4667

0.7000 0.5000 0.3000 0 0.6333 0.3333 0.7667 1.0333 0.9333 0.8667 0.7667

0.2333 0.2333 0.3333 0.6333 0 0.8000 0.3667 0.6000 0.3000 0.4000 0.4333

1.0333 0.8333 0.6333 0.3333 0.8000 0 0.4333 0.7000 1.0667 1.2000 1.1000

0.6000 0.6000 0.7000 0.7667 0.3667 0.4333 0 0.2667 0.5667 0.7667 0.8000

0.7000 0.8333 0.9333 1.0333 0.6000 0.7000 0.2667 0 0.3000 0.5333 0.9000

0.4000 0.5333 0.6333 0.9333 0.3000 1.0667 0.5667 0.3000 0 0.2333 0.6000

0.1667 0.3667 0.5667 0.8667 0.4000 1.2000 0.7667 0.5333 0.2333 0 0.3667

0.2000 0.2667 0.4667 0.7667 0.4333 1.1000 0.8000 0.9000 0.6000 0.3667 0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%ÈëÃÅ

mid=T1(11,:);

time\_input=mid;

for i=1:11

t\_last(i)=t\_la(i)-mod(time\_input(i),t\_la(i));

end

quan\_1=t\_last+mid;

quan\_2=formulation(quan\_1)

for i=1:11

quan\_3(i)=sum(quan\_2(1:i))

end

ment=rand(t\_la(1),1);

for i=1:in\_num(1)

if ment(i)<quan\_3(1)

main(i,2)=mid(1);

end

if ment(i)>=quan\_3(1)&&ment(i)<quan\_3(2)

main(i,3)=mid(2);

end

if ment(i)>=quan\_3(2)&&ment(i)<quan\_3(3)

main(i,4)=mid(3);

end

if ment(i)>=quan\_3(3)&&ment(i)<quan\_3(4)

main(i,5)=mid(4);

end

if ment(i)>=quan\_3(4)&&ment(i)<quan\_3(5)

main(i,6)=mid(5);

end

if ment(i)>=quan\_3(5)&&ment(i)<quan\_3(6)

main(i,7)=mid(6);

end

if ment(i)>=quan\_3(6)&&ment(i)<quan\_3(7)

main(i,8)=mid(7);

end

if ment(i)>=quan\_3(7)&&ment(i)<quan\_3(8)

main(i,9)=mid(8);

end

if ment(i)>=quan\_3(8)&&ment(i)<quan\_3(9)

main(i,10)=mid(9);

end

if ment(i)>=quan\_3(9)&&ment(i)<quan\_3(10)

main(i,11)=mid(10);

end

end

people=[400 30 50 30 100 50 30 30 20 50 0];

main\_num=formulation2(main) %%%%%%%%¸÷µãÈËÊý·Ö²¼

**3：数据预测程序：**

for i=1:92

if i<32

x(i)=4;

else

x(i)=1;

end

end

y=ones(1,92);

y(1)=2;y(2)=2;y(3)=2;

for i=38:44

y(i)=2;

end

for i=1:40

z(i,:)=62.528-10.529\*x+26.289\*y+i\*rand(1,92);

end

z=z';

z=round(z);

z=z';

a=20160101;b=20160201;c=20160301

for i=1:31

date(i,1)=a;

a=a+1;

end

for i=1:30

date(31+i,1)=b;

b=b+1;

end

for i=1:31

date(61+i,1)=c;

c=c+1;

end

final(:,1)=date;

final(:,2)=z;