

同济大学数学建模校内赛

参赛者信息:

姓名: _____

联系方式: _____

姓名: _____

专业: _____

联系方式: _____

姓名: _____ 学号: 学 _____

院: _____ 专业: _____

联系方式: _____

游乐园客流疏导方案及酒店订房预测

（数学建模校赛 B 题）

摘要

Youth 游乐园即将盛大开园，本次建模根据所给的游乐项目相关数据以及游乐园地图提供对游客疏导建议方案。该游乐园周边的一家酒店给出了 2015 年每天的预订酒店情况表，本次建模通过分析该表的数据做出 2016 年 1-3 月每天预订酒店的预测。

针对问题一，分为预期分析和动态监测两部分。预期分析从游园总路程和流通情况两个方面考虑，将问题分别转化为图论中的无向图和有向图问题，用线性规划的方法对问题进行求解以找到总路程最短的推荐路线和针对易发生阻塞项目的流通情况较好的辅路线推荐，用 Lingo 解决。动态监测部分，我们建立了一个以动态实时监测为基础的数学模型，以路途和等待总时长最短为标准为在各处等待的游客提供疏导建议，用 Matlab 进行模拟。

针对问题二，将它归为回归分析预测类问题，用 Excel 来进行解决。对给出的旧有数据进行简单地分析后找到预订提前天数的概率，再从中分离出影响较为显著的三个影响因素：月份、星期、节假日，然后将其分别量化为相应的修正系数。再将分离后的数据对时间序列进行合理的回归拟合以此作为初步预测值，然后对每一点乘相应的修正系数作为最终预测值，然后依照找到的提前天数概率将结果转换为关于预定日期的对预订房间数的预测。

最后，分别对所建立的模型结合实际进行评价与分析，并寻求改进。

关键词：图论 线性规划 动态模型 分离影响因素 回归分析

一、问题重述

1.1 引言

Youth 游乐园即将盛大开园，作为本市建有最多过山车的游乐园，受到了青少年的热捧。预计届时园区将迎来每天 1 万的大客流。如何根据客流情况，及时分流人群，为顾客提供游园线路引导，保障游客的游园体验显得尤为重要。

1.2 问题的提出

试就园区的整体规划，建立数学模型分析研究下面的问题：

(1) 附件 1 为 Youth 乐园的规划图，共设 A-J 共 10 个项目点，游客可沿着图中标出的线路往返下个游乐项目。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，以达到游园体验最优。每个游乐项目安排请参见表 1。

表 1. 每个游乐项目的时间安排

游乐项目	每场容纳游客数	每场持续时间
A	400	33分
B	30	1分15秒
C	50	2分30秒
D	30	2分30秒
E	100	5分
F	50	2分30秒
G	30	2分
H	30	1分30秒
I	20	1分30秒
J	50	2分



(2) 皇冠假日酒店是游乐园内的酒店, 目前已开业, 为有需要的游客提供住宿便利。请根据该酒店历史预订数据信息, 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.

二、模型假设

2.1. 问题一:

- (1) 进行预期分析模型的建立时, 假设游乐园各项目自身对游客的吸引力一致, 游客对各个游乐项目都没有禁忌; 假设每位游客的步行速度都是 1m/s , 且不在游乐项目之间的道路处停留; 假设游乐场一天的开园时间是 10 小时;
- (4) 进行动态监测模型的建立时, 假设园内所有游客均位于游乐项目区域, 且在任意时刻对于单一游客来说处于任何一游乐项目处的可能性等大。

2.2. 问题二:

- (1) 假设月份、星期、节假日对酒店预订房间数影响最为显著, 排除这些因素以后对每天的研究可以认为是同等的;
- (2) 假设星期与月份的影响作用相互独立;
- (3) 假设 2016 年酒店的预订情况与 2015 年大致相同, 无其他差异影响因素;
- (4) 假设预订房间只在提前一定范围内有可能。

三、符号说明

符号	意义
x_{ij}	地点 i 与地点 j 在路线中的连接关系
d_{ij}	地点 i 与地点 j 之间的最短距离
Z	路线总路程
t_0	所有游乐项目每场持续时间的总和

T	预计游园时间（耗费时间）
Q_i	地点 i 处的流通度
b_{ij}	从地点 i 到地点 j 的相对阻滞度
$O_{A、I 或 G}$	在项目 A、I 或 G 处所设计辅路线的总相对阻滞度
P_o	游乐园所有游乐项目情况
P_e	该时刻园内所有游客位置
Peo	具体某一游客游玩情况
P_i	提前 i 天预订房间的概率
M_i	第 i 月的月份修正系数
p_i	第 i 月任意一天的有订房概率
W_i	星期 i 的星期修正系数
H_d	日期为 d 的节假日修正系数
K_s	当天为到店日期的预定房间数初步预测值
K_e	当天为到店日期的预订房间数最终预测值
K_d	d 日期当天为预订日期的预订房间数估计值

四、问题分析

分析题意发现，第一问与第二问属于不同种类问题，第一问是优化类问题但题目设定较为开放，没有绝对明确的优化目标和最优解要求，需要我们根据实际生活经验作出问题模型的分类和建模；而第二问属于数据回归分析后进行预测的问题，由题意我们先将一些有可能影响的因素进行分离然后以时间序列为回归变量进行一元线性或非线性回归，再分离出的影响因素作为系数乘到回归方程里。

4.1. 问题一：

将问题分为两方面进行建模。

第一方面，基于预期分析模型，先设计出一条针对旅客游玩的推荐路线，用于在游览前对游客游览路线的建议。为了便于分析，我们把各个游乐设施进

行编号，把实际地图简化为赋权无向图，即转化为图论问题。这条路线满足的条件是保证单一旅客游遍所有项目的前提下，从出入口进园到从出入口出园之间的总路程最短，这样这一步过程即为巡回旅行商（TSP）问题。首先用 Floyd 算法求得各景点之间的最短路径，然后我们以从出入口出发历经所有项目回到出入口和总路程最短为约束，建立一个 0-1 线性规划，用 lingo 求解，可以得到最佳旅游路线以及其它各项信息；鉴于游客量大时在个别游乐项目处容易发生拥堵导致等待时间过长，因此我们对在该处等待的游客可以建议另外一条较为通畅的路线，让其先去进行剩下的那些未进行的项目再回来进行这个，如此以来，我们人为定义表示项目流通度和路线阻塞程度的参量按照刚才同样的方法可以在这几处较为容易堵塞的地点设计出建议的辅路线。

第二方面，基于动态监测模型，建立园区的游客疏导模型在游客游览过程当中，建立的动态数组确定到达下一项目并体验的耗费时间，以耗费时间最小为判断标准，得出最优的下一游乐项目，从而给予其每一步的具体指导，进行实时游客疏导，将各步选择进行串联，最终可以得出基于每一步实时数据分析的最佳路线。

4. 2. 问题二：

先假设几个较为可能的影响因素有月份、星期、节假日。然后依次将这三个影响因素进行分离，分别计算出这三种影响因素不同取值下的变动指数。然后对分离后的 2015 年全年预订房数以时间序列作为回归变量进行回归分析，进而以 2015 年的回归分析结果预测 2016 年的预订房间情况。

五、模型的分析、建立与求解

5. 1. 问题一

5. 1. 1. 第一方面：基于预期分析模型的最短游园路线推荐和初步疏导计划

先对游乐园各个地点进行编号，按出入口、A、B、C、D、E、F、G、H、I、J 顺序分别进行编号，对应为 1、2、3、4、5、6、7、8、9、10、11。对于单一旅客从 1 出发，然后历经其余 10 个点后回到 1，找到一条最短的路线。

	1	2	3	4	5	6	7	8	9	10	11
1	0	300	400	700	1150	650	1650	1200	1350	900	550
2	300	0	300	600	1050	350	1550	900	1050	600	250
3	400	300	0	300	750	350	1250	900	1250	800	550
4	700	600	300	0	450	500	950	1050	1400	950	850
5	1150	1050	750	450	0	950	500	1150	1550	1400	1300
6	650	350	350	500	950	0	1200	550	900	450	600
7	1650	1550	1250	950	500	1200	0	650	1050	1500	1800
8	1200	900	900	1050	1150	550	650	0	400	850	1150
9	1350	1050	1250	1400	1550	900	1050	400	0	450	800
10	900	600	800	950	1400	450	1500	850	450	0	350
11	550	250	550	850	1300	600	1800	1150	800	350	0

上表为我们采用 Floyd 算法得出的任意两点之间的最短距离表格。

接下来我们采用线性规划的方法求解，设城市之间距离用矩阵 \mathbf{d} 来表示， d_{ij} 表示点 i 与点 j 之间的距离。设 0-1 矩阵 \mathbf{x} 用来表示经过的各个点之间的路线。设

$$x_{ij} = \begin{cases} 0 & \text{若地点 } i \text{ 到地点 } j \\ 1 & \text{若地点 } i \text{ 不到地点 } j, \text{ 且 } i \text{ 在 } j \text{ 前} \end{cases}$$

考虑每一个地点后只有一个地点，则

$$\sum_{\substack{j=1 \\ j \neq i}}^{11} x_{ij} = 1, \quad i = 1, 2, \dots, 11$$

考虑每个地点前只有一个地点，则

$$\sum_{\substack{i=1 \\ i \neq j}}^{11} x_{ij} = 1, \quad j = 1, 2, \dots, 11$$

仅仅以上约束条件不能避免在一次遍历中产生多于一个互不连通回路。为此我们引入额外变量 $u_i (i = 1, 2, \dots, 11)$ ，附加以下充分约束条件：

$$u_i - u_j + nx_{ij} \leq 11 - 1 = 10, 1 < i \neq j \leq 11$$

该约束的解释为 i 与 j 不会构成回路，若构成回路， $x_{ij} = 1, x_{ji} = 1$ ，则 $u_i - u_j \leq -1$ 且 $u_j - u_i \leq -1$ ，从而 $0 \leq -2$ ，导致矛盾。如 i, j 与 k 不会构成，若构成回路， $x_{ij} = 1, x_{jk} = 1, x_{ki} = 1$ ，则 $u_i - u_j \leq -1, u_j - u_k \leq -1, u_k - u_i \leq -1$ ，从而 $0 \leq -3$ ，导致矛盾。其他情况以此类推。

综合上述三个约束条件，我们得到如下优化模型：

$$\begin{aligned} \min Z &= \sum_{i=1}^{11} \sum_{j=1}^{11} d_{ij} x_{ij} \\ \text{s.t.} &\begin{cases} \sum_{\substack{i=1 \\ i \neq j}}^{11} x_{ij} = 1, & i = 1, 2, \dots, 11 \\ \sum_{\substack{j=1 \\ j \neq i}}^{11} x_{ij} = 1, & i = 1, 2, \dots, 11 \\ u_i - u_j + n x_{ij} \leq 10, & 1 < i \neq j \leq 11 \\ x_{ij} = 0 \text{ 或 } 1, & i, j = 1, 2, \dots, 11 \\ u_i \text{ 为实数}, & i, j = 1, 2, \dots, 11 \end{cases} \end{aligned}$$

上述模型用 lingo 编程求解后得到行程路线如下：

$X(1,2)=1, X(2,11)=1, X(3,1)=1, X(4,6)=1, X(5,4)=1, X(6,3)=1, X(7,5)=1, X(8,7)=1, X(9,8)=1, X(10,9)=1, X(11,10)=1$ ，其它全为 0。则对应该的最短路线为：

出入口 $\rightarrow A \rightarrow J \rightarrow I \rightarrow H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow E \rightarrow B \rightarrow$ 出入口，其最短距离为 4600 米，预计最短游园时间为 $T_{\min} = \frac{4600m}{1m/s} + t_0 = 2 \text{ h } 10 \text{ min } 25 \text{ s}$ 。

进一步分析：我们观察到题目所给的各个项目客容量和每场持续时间的条件，针对每个游乐项目可以以它的每场容纳游客数与每场持续时间作为衡量其流通度的标准，记为 $Q_i, i = 1, 2, \dots, 11$ ，则得到每个游乐项目的流通度如下表：

地点编号	游乐项目编号	流通度 Q_i (人/秒)
1	出入口	0.277777777777
2	A	0.20202020202
3	B	0.4
4	C	0.333333333333
5	D	0.2
6	E	0.333333333333
7	F	0.333333333333
8	G	0.25
9	H	0.333333333333
10	I	0.222222222222
11	J	0.416666666666

其中值得说明的是出入口的流通度按照一天一万人的客流量除以假设的每日开园时间 10 小时进行计算，即 $Q_1 = \frac{10000}{10 \times 60 \times 60} = 0.277777777777$ 人/秒。

鉴于游乐园正值游客高峰期，因此考虑对于易于发生游客堆滞排队等候的游乐项目处的游客进行疏导。换句话说，就是针对流通度 Q_i 较低的游乐项目 A、I、G 进行相应的辅路线规划，让在该处等待的游客先去按照另外一条路线进行其他游乐项目然后最后回到该游乐项目处进行此项目，同时应尽量避免途经已经进行过的项目，在此基础上尽量使此条辅路线的总相对阻滞度最低。

当 i 地点与 j 地点相邻（即无需途经其他地点）时设从 i 地点到 j 地点的相对阻滞度为 $b_{ij} = \frac{Q_i}{Q_j}$ ，当 i 地点与 j 地点不相邻时设其相对阻滞度 b_{ij} 为从 i 到 j 的最小总阻滞度（比如说，地点 i 与地点 j 之间隔有地点 k，则 $b_{ij} = b_{ik} + b_{kj}$ 。

这样我们把该问题转化为一个图论中的有向图最短路问题，我们利用 Dijkstra 算法或者 Floyd 算法可以得到如下相对阻滞度的表格：

	1	2	3	4	5	6	7	8	9	10	11
1	1	1.375	0.694	1.894	3.561	1.894	3.978	3.228	3.978	3.394	1.86
2	0.727	1	0.505	1.606	3.372	0.606	2.689	1.939	2.689	2.106	0.485
3	1.44	1.98	1	1.2	2.897	1.2	3.867	2.533	3.283	2.7	2.465
4	2.273	2.65	0.833	1	1.667	1	2.267	2.333	3.167	2.5	3.135
5	2.873	3.25	1.433	0.6	1	1.6	0.6	1.933	2.683	3.1	3.633
6	2.273	1.65	0.833	1	2.667	1	2.083	1.333	2.083	1.5	2.033
7	4.357	3.733	2.917	3.083	1.667	2.083	1	1.333	2.083	1.5	2.033
8	3.023	2.4	1.583	1.75	2.417	0.75	0.75	1	0.75	2.25	2.783
9	4.44	3.733	2.917	3.083	3.75	2.083	2.083	1.333	1	1.5	2.033
10	2.94	2.317	1.5	1.667	3.333	0.667	2.75	2	0.667	1	0.533
11	2.79	2.062	2.568	3.542	5.208	2.542	4.625	3.875	2.542	1.875	1

我们同样采用线性规划的方法进行求解，设地点之间的相对阻滞度用矩阵 B 来表示， b_{ij} 表示地点 i 与地点 j 之间的距离。依旧用矩阵 X 表示各城市之间的路线。设

$$x_{ij} = \begin{cases} 0 & \text{若城市 } i \text{ 不到城市 } j \\ 1 & \text{若城市 } i \text{ 到城市 } j \text{ 且 } i \text{ 在 } j \text{ 前} \end{cases}$$

则我们依旧可以得到如下等式：

$$\sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, i = 1, \dots, n ; \quad \sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, i = 1, \dots, n$$

对此式需要说明的是，我们约束辅路线的规划要求满足不能途经已经游玩过的项目地点，因此针对 A、I、G 分别的辅路线设计需要对 n 取不同的值。而 n 值的大小的确定方法如下：

以我们之前找到的那条总路程最短的游览线路作为参照，除去所研究项目之前的项目，剩下的项目总数即作为 n 的取值。这样，针对 A、I、G 我们应分别取 n 为 10，8，6。与此同时在问题的求解过程中我们应分别对地点进行重新编号以方便求解：

求解 A 的辅路线时的编号调整

游乐项目	A	B	C	D	E	F	G	H	I	J
调整后的地点编号	1	2	3	4	5	6	7	8	9	10

求解 I 的辅路线时的编号调整

游乐项目	B	C	D	E	F	G	H	I
调整后的地点编号	1	2	3	4	5	6	7	8

求解 G 的辅路线时的编号调整

游乐项目	B	C	D	E	F	G
调整后的地点编号	1	2	3	4	5	6

接下来我们仍旧引入额外变量 $u_i (i = 1, \dots, n)$ 附加充分约束条件避免在一次遍历中产生多于一个互不连通回路：

$$u_i - u_j + nx_{ij} \leq n - 1, \quad 1 < i \neq j \leq n$$

综合上述约束条件我们归纳出如下优化模型：

$$\min O_{A, I \text{ 或 } G} = \sum_{i=1}^n \sum_{j=1}^n b_{ij} x_{ij}$$

$$\text{s. t.} = \begin{cases} n \text{ 为剩余的项目总数} \\ \sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, & j = 1, \dots, n \\ \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, & i = 1, \dots, n \\ u_i - u_j + nx_{ij} \leq n - 1, & 1 < i \neq j \leq n \\ x_{ij} = 0 \text{ 或 } 1, & i, j = 1, \dots, n \\ u_i \text{ 为实数}, & i = 1, \dots, n \end{cases}$$

上述优化模型用 Lingo 分别求解得：

针对 A 的辅路线解得如下： $x(1,10)=1, x(10,9)=1, x(9,8)=1, x(8,7)=1, x(7,6)=1, x(6,4)=1, x(4,3)=1, x(3,2)=1, x(2,1)=1$ ，其他全为 0。对应的最通畅辅路线为

$A \rightarrow J \rightarrow I \rightarrow H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow A$;

针对 I 的辅路线解得如下: $X(1,4)=1, X(4,8)=1, X(8,7)=1, X(7,6)=1, X(6,5)=1, X(5,3)=1, X(3,2)=1, X(2,1)=1$, 其他全为 0。对应的最通畅辅路线为 $I \rightarrow H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow I$;

针对 G 的辅助路线解得如下: $X(1,4)=1, X(4,6)=1, X(6,5)=1, X(5,3)=1, X(3,2)=1, X(2,1)=1$, 其他全为 0。对应的最通畅辅路线为 $G \rightarrow F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow G$ 。

综上所述, 我们可以给出游客一条最短线路作为推荐线路: 出入口 $\rightarrow A \rightarrow J \rightarrow I \rightarrow H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow E \rightarrow B \rightarrow$ 出入口, 并给出理论最短游览时间 $T = 2h\ 10min\ 25s$ 。而与此同时在客流量较大的时刻我们可以分别给在比较容易发生拥堵的 A、I、G 地点排队等候的游客以下建议:

对于在 A 处等候的游客可以建议他们先绕过 A 去 J 然后按照 $J \rightarrow I \rightarrow H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow A$ 的顺序进行游览, 最后再回到 A 处进行 A 项目的游玩;

对于在 I 处等候的游客可以建议他们先绕过 I 去 H 然后按照 $H \rightarrow G \rightarrow F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow I$ 的顺序进行游览, 最后再回到 I 处进行 I 项目的游玩;

对于在 G 处等候的游客可以建议他们先绕过 G 去 F 然后按照 $F \rightarrow D \rightarrow C \rightarrow B \rightarrow E \rightarrow G$ 的顺序进行游览, 最后再回到 G 处进行 G 项目的游玩;

如此保证游客尽量减少在拥堵处等待所浪费的时间而选择一条有可能更为通畅的游览线路。

5.1.2. 第二方面: 基于动态监测模型的实时疏导

作为动态监测的基础, 我们可以通过对每位游客进行实时定位 (例如在每个游乐项目入口扫描门票、在游乐园门票上加装定位装置等), 从而得到某一时刻每位游客的具体位置。

在此基础上, 我们利用 MatLab 建立动态模型:

情景简化: 每日最大客流量一万人, 估计某一时刻园区内游客总人数最大为两千人; 同时, 为了分析方便, 对实际情况进行一定的简化, 认为此时所有游客均位于游乐项目区域, 不考虑区域外的客流。

首先, 我们建立动态数组: $Po [11 \times 3]$ (表示某一时刻 10 个游乐项目的每

场容纳游客数 No 、每场持续时间 To 、实际游客总数 n)、 Pe [2000X1] (表示某一时刻所有游客所在的游乐项目位置)，以此构建园区游客分布的动态模型。

同时，建立其余静态数组：任意两游乐项目的直接距离 x (若无法到达，则为 0)，和任意两游乐项目的最短距离 d (对角线元素为 0)，如图所示：

```
x=[ 0, 300, 400, 0, 0, 0, 0, 0, 0, 0, 0;
    300, 0, 300, 0, 0, 350, 0, 0, 0, 0, 250;
    400, 300, 0, 300, 0, 350, 0, 0, 0, 0, 0;
    0, 0, 300, 0, 450, 500, 0, 0, 0, 0, 0;
    0, 0, 0, 450, 0, 0, 500, 0, 0, 0, 0;
    0, 350, 350, 500, 0, 0, 0, 550, 0, 450, 0;
    0, 0, 0, 0, 500, 0, 0, 650, 0, 0, 0;
    0, 0, 0, 0, 0, 550, 650, 0, 400, 0, 0;
    0, 0, 0, 0, 0, 0, 0, 400, 0, 450, 0;
    0, 0, 0, 0, 0, 450, 0, 0, 450, 0, 350;
    0, 250, 0, 0, 0, 0, 0, 0, 0, 350, 0];
```

```
d=[0, 300, 400, 700, 1150, 650, 1650, 1200, 1350, 900, 550;
    300, 0, 300, 600, 1050, 350, 1550, 900, 1050, 600, 250;
    400, 300, 0, 300, 750, 350, 1250, 900, 1250, 800, 550;
    700, 600, 300, 0, 450, 500, 950, 1050, 1400, 950, 850;
    1150, 1050, 750, 450, 0, 950, 500, 1150, 1550, 1400, 1300;
    650, 350, 350, 500, 950, 0, 1200, 550, 900, 450, 600;
    1650, 1550, 1250, 950, 500, 1200, 0, 650, 1050, 1500, 1800;
    1200, 900, 900, 1050, 1150, 550, 650, 0, 400, 850, 1150;
    1350, 1050, 1250, 1400, 1550, 900, 1050, 400, 0, 450, 800;
    900, 600, 800, 950, 1400, 450, 1500, 850, 450, 0, 350;
    550, 250, 550, 850, 1300, 600, 1800, 1150, 800, 350, 0];
```

在此基础上，我们选取某一确定的游客 Peo [1X12] (其中第一项为该游客某一时刻所在位置，第二到第十二项为游客是否已经到达过该区域，出入口、A、B、C、D、E、F、G、H、I、J 分别以数字 1~11 代表，以 0 表示未到达，1 表示已到达) 为研究对象。

初始时刻，该游客进入游乐园：

$$Peo = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

确定此时的 Pe ： Pe 由实时监测得到，故本次建模中用随机数替代：

$$Pe(i, 1) = randi(10) + 1;$$

确定此时的 Po ： Po 中前两列 (每场容纳游客数 No 、每场持续时间 To) 由题目得到，第三列 (实际游客总数 n) 由 Pe 累加得到 (累加程序如图所示)：

$Po(i,1) = No(i), i = 1, \dots, 11, No = [inf, 400, 30, 50, 30, 100, 50, 30, 30, 20, 50]$

$Po(i,2) = To(i), i = 1, \dots, 11, To = [inf, 1980, 75, 150, 150, 300, 150, 120, 90, 90, 120]$

```
for i=1:2000
    for j=1:10
        if Pe(i,1)==j;
            Po(j,3)=Po(j,3)+1;
        end
    end
end
```

然后，我们针对现有实际情况，为该游客选择下一步合适的游乐项目：

建立耗费时间数组 $t[1 \times 11]$ （表示由当前位置到达下一个游乐项目的所用时间）作为下一个最优的游乐项目判断的指标。 $t(i)$ 由两方面相加而成：第一方面为行程时间，我们用到下一个游乐项目的距离（若无法直接到达，令距离为 inf （无穷大））除以步行速度（人均步行速度估计为 $1m/s$ ）得到行程时间；第二方面为等待时间，我们利用项目现有人数减去每场容纳游客数除以每场容纳游客数再乘以每场持续时间计算出游客游玩下一项目的等待时间，判断程序如下图所示：

```
n=zeros(1,11);
to=zeros(1,11);
for j=1:11
    if x(Peo(1),j)~=0&&Peo(j+1)==0;
        n(j)=Po(j,3);
        to(j)=x(Peo(1),j)/1;
    else
        n(j)=inf;
        to(j)=inf;
    end
end
t=zeros(1,11);
for i=1:11
    if n(i)==inf;
        t(i)=inf;
    elseif n(i)~=inf&&n(i)>No(i);
        t(i)=ceil((n(i)-No(i))/No(i))*To(i)+to(i);
    else
        t(i)=to(i);
    end
end
```

由此我们经过计算可以得出由该游客所处位置到达下一游乐项目的耗费

时间（由程序可见，若无法直接到达，或该项目已经体验过，则耗费时间为 \inf （无穷大）），再从中取得最小值 t_{min} ，即可选择出下一个最优的游乐项目。

若走入死点（ $t_{min} = \inf$ ），则直接前往未去过的游乐项目（中途可经过其他项目），该项目则为下一个游乐项目。

```
[m, l]=min(t);
if m==inf;
    s=[inf, inf, inf, inf, inf, inf, inf, inf, inf, inf];
    for i=3:12
        if Peo(i)==0;
            s(i-2)=d(Peo(1), Peo(i))/1;
        end
    end
    [m, lo]=min(s);
    r(ro)=lo+1;
    Peo(1)=lo+1;
    Peo(lo+2)=1;
else
    r(ro)=1;
    Peo(1)=1;
    Peo(1+1)=1;
end
```

最后，在该游客体验的每个游乐项目处重复进行上述判断（在每次判断过程当中，各游客所在位置和各游乐项目的游客人数都会利用随机数重新生成并且累加得到，以此模拟实时数据动态变化），直到游客体验过所有项目为止：

$$\prod_{i=2}^{12} \text{Peo}(i) \neq 0$$

则可得出一条基于游客实际情况实时分析得出的最优线路 r 。

同时，将每一步的耗费时间累加可以得到总共的非体验耗费时间：

$$T = \sum_{i=1}^{10} t_{min}(i)$$

动态模型建立如上，我们进行模拟分析，验证模型可靠性。以客流随机分布为基础，多次模拟可以得出两条在随机情况下的最优路径：

$$r_1 = [1, 2, 11, 10, 9, 8, 6, 3, 4, 5, 7, 1], r_2 = [1, 2, 11, 10, 6, 3, 4, 5, 7, 8, 9, 1]$$

即路线一：出入口 $\rightarrow A \rightarrow J \rightarrow I \rightarrow H \rightarrow G \rightarrow E \rightarrow B \rightarrow C \rightarrow D \rightarrow F \rightarrow$ 出入口，

路线二：出入口 $\rightarrow A \rightarrow J \rightarrow I \rightarrow H \rightarrow E \rightarrow B \rightarrow C \rightarrow D \rightarrow F \rightarrow G \rightarrow$ 出入口。

其中路线一的总耗费时间约在 $T_1 = 2\text{h } 26\text{min}$ 左右，路线二的总耗费时间约在 $T_2 = 2\text{h } 28\text{min}$ 左右。

因此说明该模型可靠。

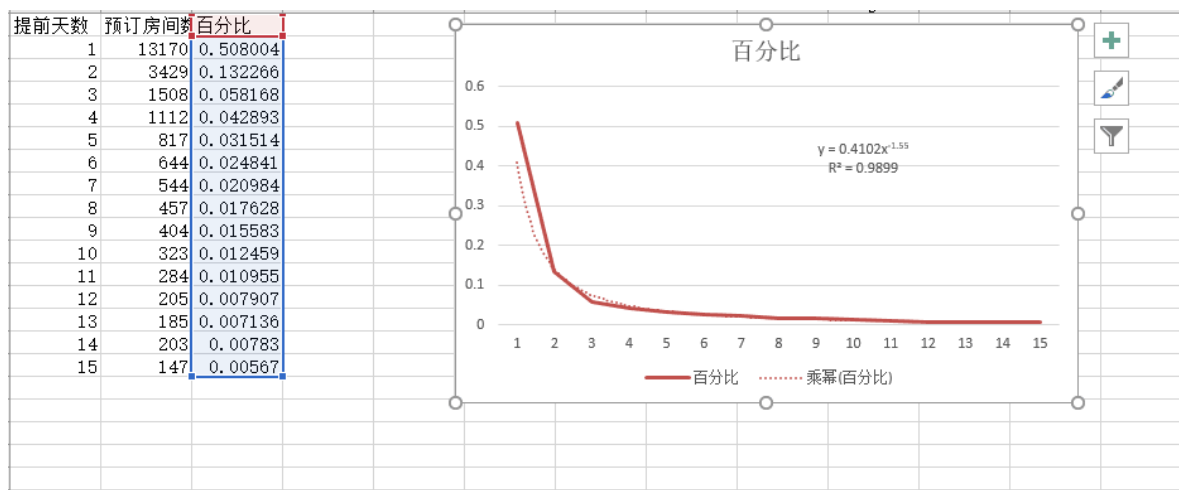
此处需加以强调说明，该模型是针对实时动态客流进行的动态路径分析，每位游客所在的位置都应通过实测而得到，所得出的最终路径会根据实际各个时刻情况的不同发生改变，本次分析只是采用随机数代替实测值，因此该疏导模型是动态模型，为游乐园游客提供实时疏导。

5.2. 问题二

该问题的解决需要我们利用所给的 2015 年的酒店预订与入住情况预测 2016 年 1 至 3 月的酒店预订情况。基于问题的设定，我们将其转化为回归分析预测问题，考虑利用 Excel 解决。

首先，从实际问题出发，根据生活经验我们认为酒店的预订房数量情况与到店日期的关系更为直接，而与预订日期之间的关系不够直接，因此我们先要试图找到预订日期与到店日期之间关系，然后通过预测 2016 年关于到店日期的每天预订房间数，通过预订日期与到店日期之间的关系回推出 2016 年 1 到 3 月关于预订日期的每天预订房间数。

这样，我们设预订日期相对于到店日期的提前天数为 M ，则可以汇总得到预定房间数 N 关于提前天数 M 的表格。然后将不同提前天数所对应的预订房间数与总预订房间数作比值作为描述真实提前天数等于该提前天数的概率百分比。通过分析所得表格我们发现提前天数 $M > 14$ 时，概率小于 0.005，对于曲线趋势的影响较大而对实际值预测的影响较小，因此我们删去该部分数据进行拟合，之后在对拟合结果进行相关修正。拟合结果如下：



对上图加以说明：因为 Excel 在曲线拟合功能上的一些局限，所以我们先将提前天数认为加一，如此可以实现拟合，然后再将函数图像左移一个单位长度，如此得到的函数关系式为：（y 表示预定房间数，x 表示提前天数）

$$y = \frac{0.4102}{(x + 1)^{1.55}}, \quad x = 0, 1, \dots, 14$$

$$R^2 = 0.9899$$

我们发现 R^2 比较接近 1，说明函数关系还较为明显。

然后按该曲线分别计算出 x 为 0 至 14 时的函数值，再分别同比例扩大使其总和为 1，定义该值为提前天数概率 $P_i (i = 0, 1, \dots, 14)$ 。计算结果如下：

提前天数	提前天数概率 P_i
0	0.492318771
1	0.168131815
2	0.089682654
3	0.057418707
4	0.040629632
5	0.030627529
6	0.024118174
7	0.019609066
8	0.016336932
9	0.013875428
10	0.011969823
11	0.010459609

12	0.009239196
13	0.008236599
14	0.007401248

接下来，我们要试图找到预订房间数与到店日期之间的关系。通过初步观察数据即图像，我们发现月份和节假日对当天预订房间数的影响比较大，加之考虑到星期也可能影响到预订房间数，因此我们需要先分离出这三个影响因素：

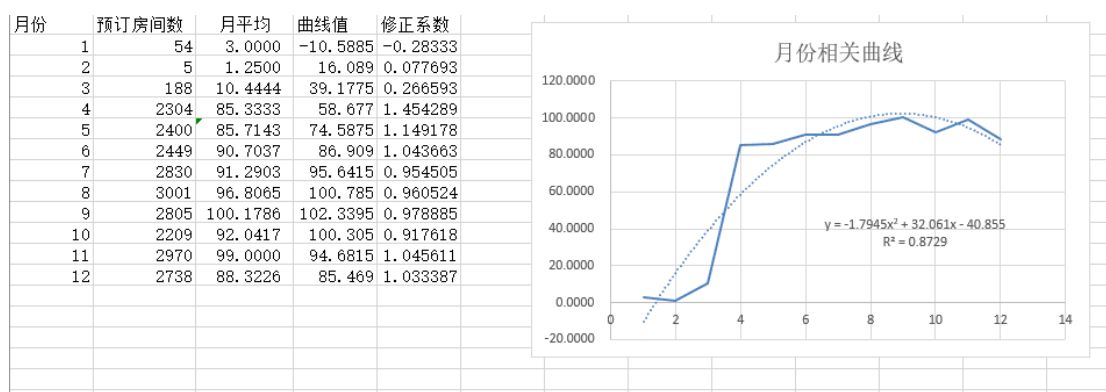
（1）先对月份影响因素进行分离：

因为节假日（元旦：1月1日-1月3日；清明：4月4日-4月6日；五一：5月1日-5月3日；端午：6月20日-6月22日；中秋：9月26日-9月27日；国庆：10月1日-10月7日）对于月份并不是均匀分布的，而星期在每个月接近均匀分布，因此分析月份影响因素时节假日的因素会对它产生较大影响，因此我们先剔除法定节假日和没有订房的日期，对其余有订房的一般日期算出每个月的平均每天订房数，并拟合处一条二次曲线作为预测参考。得到该曲线方程为：（ y 表示该月平均每天预订房间数， x 表示月份）

$$y = -1.7945x^2 + 32.061x - 40.855, \quad x = 1, \dots, 12$$

$$R^2 = 0.8729$$

然后计算根据曲线计算出该月的预测平均每天预订房间数，用每个月实际的平均每天预订房间数除以该月的预测平均每天预订房间数作为该月的月份修正系数 $M_i (i = 1, 2, \dots, 12)$ 。归纳该过程到如下表格及图像：



汇总一下月份修正系数为：

月份	月份修正系数
1	-0.28333

2	0.077693
3	0.266593
4	1.454289
5	1.149178
6	1.043663
7	0.954505
8	0.960524
9	0.978885
10	0.917618
11	1.045611
12	1.033387

而在此计算过程中我们剔除了当天预订房间数为零的点,因此还需要一个针对每个月中任意一天有订房的概率 $p_i = \frac{\text{第 } i \text{ 月订房数不为零的天数}}{\text{第 } i \text{ 月总天数}} (i = 1, \dots, 12)$, 结果如下表:

月份	第 i 月任意一天有订房概率 p_i
1	0.580645161
2	0.142857142
3	0.580645161
4	0.9
5	0.903225806
6	0.9
7	1
8	1
9	0.933333333
10	0.774193548
11	1
12	1

(2) 再对星期影响因素进行分离:

考虑到节假日对星期仍然存在较大干扰, 保险起见我们先将节假日剔除,

然后对剩下的日期按星期对预订房间数进行求和汇总得到星期一到星期日分别的总预定房间数，然后分别除以这七个数的平均数作为分别的星期修正系数 $W_i (i = 1, \dots, 7)$ 。该过程的 Excel 操作过程如下：

星期	预订房间数	修正系数
1	4177	1.1278
2	4117	1.1116
3	2995	0.8087
4	3390	0.9153
5	3883	1.0484
6	3932	1.0617
7	3431	0.9264
总计	25925	3703.5714

图表中修正系数即为相对应于星期的星期修正系数 W_i 。

(3) 对节假日因素进行分离：

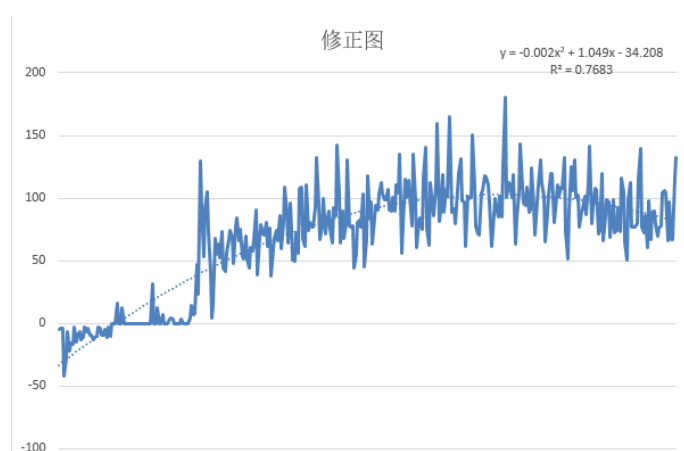
节假日修正系数 H_d (d 表示当天的日期) 是根据节假日期间到店人数的激增而做出的修正，基本方法是根据节假日期间的日均到店人数与当月日均到店人数的比值进行修正，以此评估节假日对到店人数的大小。同时，考虑到 1 月 1 日到 3 日的特殊性，在月份修正的曲线拟合时节假日影响就已包括，故在对节假日影响进行修正时不考虑元旦的影响，其他节假日的影响修正如前所述。

修正系数的值如下表所示。

到店日期	预定数	节假日修正系数 H_d
2016/1/1	76	1
2016/1/2	28	1
2016/1/3	29	1
2015/4/4	139	2.0031
2015/4/5	77	1.1096
2015/4/6	81	1.1673
2015/5/1	143	2.0607
2015/5/2	91	1.3114
2015/5/3	69	0.9943
2015/6/20	145	2.0895

2015/6/21	75	1.0808
2015/6/22	64	0.9223
2015/9/26	123	1.7725
2015/9/27	57	0.8214
2015/10/1	138	1.9887
2015/10/2	119	1.7149
2015/10/3	90	1.2970
2015/10/4	139	2.0031
2015/10/5	117	1.6860
2015/10/6	113	1.6284
2015/10/7	59	0.8502

最后我们去除了上述影响因素后，将每天的预订房间数对时间序列进行二次曲线拟合得到下图：（y 表示该天预定房间数，x 表示时间序列）



其中表达式为：

$$y = -0.002x^2 + 1.049x - 34.208$$

$$R^2 = 0.7683$$

然后按照该函数关系计算出关于到店日期的每天预订房间数的初步预测值 K_s ，然后我们针对每天情况乘以相对应的月份修正系数、有订房概率、星期修正系数、节假日修正系数最终得到以该天为到店日期的每天预定房间数的最终预测值 K_e ：

$$K_e = M_i \times p_i \times W_i \times H_d \times K_s$$

（注：式中的不同系数的 i 根据研究当天的情况取与之相对应的值）

接下来我们需要根据关于到店日期的每天预定房间数来求解 2016 年 1-3 月关于预定日期的每天预订房间数 K_d ，这样我们需要用到最初由预订日期和到店日期之间关系找到的提前天数概率 P_i ：

$$K_d = \sum_{i=0}^{14} K_{d+i} P_i$$

按照上述过程，我们最终求解出 2016 年 1-3 月每天预订房间数的预测值，和 2015 年 1-3 月每天预订房间数的图像对比如下：



最终的详细计算结果及相应系数的选取请见后面的 p.23.附录 1 预测结果汇总。

六、模型评价及改进

6.1. 问题一：

首先，在建立预期推荐路线时我们按总路程最短处理略为理想化，而在建立预期最通畅辅路线时我们定义项目流通度和路线相对阻滞度的时候的处理略微简单，只能简单地反映相关关系而没有做严密的证明也没有试图找到更为准确的衡量标准，而同时也忽略了游客在道路处可能发生的停留及阻塞。而对于路线的相对阻滞度我们也只是就单一路线进行简单的元素之间的相加，而实际阻塞度应该综合所有路线和一个项目与其他几个项目之间的联通关系组合来看，在这方面我们没有进行足够的考量。

再者，在后来建立实时动态监测模型的时候我们没有调查不同游乐项目对游客吸引程度的差异，而且我们理想地认为时间对游客的分布是没有影响的，而实际这个影响在游乐园开闭园时段是会有影响的（比如说，游乐园刚开园的

阶段游客没有足够多的时间能走得比较深入，因此大部分游客应该集中于离出入口近的地点，而闭园前游客们陆续往外走也应在出入口近处较为集中)。还有就是我们假设游乐园每时每刻园内人数是不变的也略为草率，实际上游乐园在一天经营过程中是会存在高峰期的。

6.2. 问题二：

本模型综合考虑了节假日、星期、月份等因素的综合影响，最终的预测结果能大致反映一段时间内预定日期与预定量的平均关系，给出了预定量的大致变化趋势，忽略了许多偶然因素的影响，可以作为酒店经营管理的一个参考依据。但是在模型的曲线拟合过程中，受限于所学知识与工具，以 Excel 的曲线拟合数据为基础，结果较为粗糙，无法得到精准的结果，使结论的可信度受到一定的影响。同时，在影响因素的考虑上还是有所欠缺，部分要素如节假日修正的处理过于粗糙，可以对每个节假日的影响系数分别建立模型进行评估。同时，对某些数据发生激变的点也可以建立模型分析其内在联系，从而使模型更为完善。

七、参考文献

- [1] 肖华勇、周吕文、赵松，大学生数学建模竞赛指南，北京：电子工业出版社，2015.4
- [2] 姜启源、谢金星、叶俊，数学模型（第四版），北京：高等教育出版社，2011.1

附录 1 预测结果汇总

日期	星期修正系数	月份修正系数	节假日修正系数	有订房概率	预定房间数初步预测值(关于到店日期)	预定房间数最终预测值(关于到店日期)	估计预定房间数(关于预定日期)
2016/1/1	1.0484	-0.2833	1	0.5806	-33.1610	5.7189	5.3070
2016/1/2	1.0617	-0.2833	1	0.5806	-32.1180	5.6093	5.1039
2016/1/3	0.9264	-0.2833	1	0.5806	-31.0790	4.7361	4.7009
2016/1/4	1.1278	-0.2833	1	0.5806	-30.0440	5.5737	4.9349
2016/1/5	1.1116	-0.2833	1	0.5806	-29.0130	5.3052	4.5552
2016/1/6	0.8087	-0.2833	1	0.5806	-27.9860	3.7229	3.8181
2016/1/7	0.9153	-0.2833	1	0.5806	-26.9630	4.0597	3.9853
2016/1/8	1.0484	-0.2833	1	0.5806	-25.9440	4.4743	4.0895
2016/1/9	1.0617	-0.2833	1	0.5806	-24.9290	4.3538	3.8974
2016/1/10	0.9264	-0.2833	1	0.5806	-23.9180	3.6449	3.5517
2016/1/11	1.1278	-0.2833	1	0.5806	-22.9110	4.2504	3.6956
2016/1/12	1.1116	-0.2833	1	0.5806	-21.9080	4.0060	3.3701
2016/1/13	0.8087	-0.2833	1	0.5806	-20.9090	2.7815	2.7802
2016/1/14	0.9153	-0.2833	1	0.5806	-19.9140	2.9983	2.8684
2016/1/15	1.0484	-0.2833	1	0.5806	-18.9230	3.2634	2.9054
2016/1/16	1.0617	-0.2833	1	0.5806	-17.9360	3.1325	2.7240
2016/1/17	0.9264	-0.2833	1	0.5806	-16.9530	2.5835	2.4342
2016/1/18	1.1278	-0.2833	1	0.5806	-15.9740	2.9635	2.4867
2016/1/19	1.1116	-0.2833	1	0.5806	-14.9990	2.7426	2.2111
2016/1/20	0.8087	-0.2833	1	0.5806	-14.0280	1.8661	1.7627
2016/1/21	0.9153	-0.2833	1	0.5806	-13.0610	1.9665	1.7731
2016/1/22	1.0484	-0.2833	1	0.5806	-12.0980	2.0864	1.7448
2016/1/23	1.0617	-0.2833	1	0.5806	-11.1390	1.9454	1.5751
2016/1/24	0.9264	-0.2833	1	0.5806	-10.1840	1.5519	1.3414
2016/1/25	1.1278	-0.2833	1	0.5806	-9.2330	1.7129	1.3165
2016/1/26	1.1116	-0.2833	1	0.5806	-8.2860	1.5151	1.0998
2016/1/27	0.8087	-0.2833	1	0.5806	-7.3430	0.9768	0.7957
2016/1/28	0.9153	-0.2833	1	0.5806	-6.4040	0.9642	0.7364
2016/1/29	1.0484	-0.2833	1	0.5806	-5.4690	0.9432	0.6476
2016/1/30	1.0617	-0.2833	1	0.5806	-4.5380	0.7925	0.4831
2016/1/31	0.9264	-0.2833	1	0.5806	-3.6110	0.5503	0.2693
2016/2/1	1.1278	0.0777	1	0.1429	-2.6880	-0.0336	-0.0124
2016/2/2	1.1116	0.0777	1	0.1429	-1.7690	-0.0218	-0.0008
2016/2/3	0.8087	0.0777	1	0.1429	-0.8540	-0.0077	0.0108
2016/2/4	0.9153	0.0777	1	0.1429	0.0570	0.0006	0.0199
2016/2/5	1.0484	0.0777	1	0.1429	0.9640	0.0112	0.0302
2016/2/6	1.0617	0.0777	1	0.1429	1.8670	0.0220	0.0403

日期	星期修正系数	月份修正系数	节假日修正系数	有订房概率	预定房间数初步预测值(关于到店日期)	预定房间数最终预测值(关于到店日期)	估计预定房间数(关于预定日期)
2016/2/7	0.9264	0.0777	1	0.1429	2.7660	0.0284	0.0491
2016/2/8	1.1278	0.0777	1	0.1429	3.6610	0.0458	0.0620
2016/2/9	1.1116	0.0777	1	0.1429	4.5520	0.0562	0.0702
2016/2/10	0.8087	0.0777	1	0.1429	5.4390	0.0488	0.0730
2016/2/11	0.9153	0.0777	1	0.1429	6.3220	0.0642	0.0868
2016/2/12	1.0484	0.0777	1	0.1429	7.2010	0.0838	0.1011
2016/2/13	1.0617	0.0777	1	0.1429	8.0760	0.0952	0.1105
2016/2/14	0.9264	0.0777	1	0.1429	8.9470	0.0920	0.1160
2016/2/15	1.1278	0.0777	1	0.1429	9.8140	0.1229	0.1340
2016/2/16	1.1116	0.0777	1	0.1429	10.6770	0.1317	0.1655
2016/2/17	0.8087	0.0777	1	0.1429	11.5360	0.1036	0.1824
2016/2/18	0.9153	0.0777	1	0.1429	12.3910	0.1259	0.2298
2016/2/19	1.0484	0.0777	1	0.1429	13.2420	0.1541	0.2854
2016/2/20	1.0617	0.0777	1	0.1429	14.0890	0.1660	0.3379
2016/2/21	0.9264	0.0777	1	0.1429	14.9320	0.1535	0.3878
2016/2/22	1.1278	0.0777	1	0.1429	15.7710	0.1974	0.4733
2016/2/23	1.1116	0.0777	1	0.1429	16.6060	0.2049	0.5501
2016/2/24	0.8087	0.0777	1	0.1429	17.4370	0.1565	0.6184
2016/2/25	0.9153	0.0777	1	0.1429	18.2640	0.1856	0.7499
2016/2/26	1.0484	0.0777	1	0.1429	19.0870	0.2221	0.9199
2016/2/27	1.0617	0.0777	1	0.1429	19.9060	0.2346	1.1352
2016/2/28	0.9264	0.0777	1	0.1429	20.7210	0.2131	1.4480
2016/2/29	1.1278	0.0777	1	0.1429	21.5320	0.2696	2.0640
2016/3/1	1.1116	0.2666	1	0.5806	22.3390	3.8440	3.7672
2016/3/2	0.8087	0.2666	1	0.5806	23.1420	2.8971	3.4566
2016/3/3	0.9153	0.2666	1	0.5806	23.9410	3.3922	3.8332
2016/3/4	1.0484	0.2666	1	0.5806	24.7360	4.0145	4.1888
2016/3/5	1.0617	0.2666	1	0.5806	25.5270	4.1954	4.2929
2016/3/6	0.9264	0.2666	1	0.5806	26.3140	3.7736	4.2354
2016/3/7	1.1278	0.2666	1	0.5806	27.0970	4.7307	4.6900
2016/3/8	1.1116	0.2666	1	0.5806	27.8760	4.7968	4.6344
2016/3/9	0.8087	0.2666	1	0.5806	28.6510	3.5867	4.2150
2016/3/10	0.9153	0.2666	1	0.5806	29.4220	4.1688	4.6486
2016/3/11	1.0484	0.2666	1	0.5806	30.1890	4.8994	5.0524
2016/3/12	1.0617	0.2666	1	0.5806	30.9520	5.0870	5.1477
2016/3/13	0.9264	0.2666	1	0.5806	31.7110	4.5476	5.0484
2016/3/14	1.1278	0.2666	1	0.5806	32.4660	5.6680	5.5657
2016/3/15	1.1116	0.2666	1	0.5806	33.2170	5.7158	5.4707
2016/3/16	0.8087	0.2666	1	0.5806	33.9640	4.2518	4.9461

日期	星期修正系数	月份修正系数	节假日修正系数	有订房概率	预定房间数初步预测值(关于到店日期)	预定房间数最终预测值(关于到店日期)	估计预定房间数(关于预定日期)
2016/3/17	0.9153	0.2666	1	0.5806	34.7070	4.9176	5.4345
2016/3/18	1.0484	0.2666	1	0.5806	35.4460	5.7526	6.3421
2016/3/19	1.0617	0.2666	1	0.5806	36.1810	5.9464	6.9483
2016/3/20	0.9264	0.2666	1	0.5806	36.9120	5.2934	7.3375
2016/3/21	1.1278	0.2666	1	0.5806	37.6390	6.5711	9.1951
2016/3/22	1.1116	0.2666	1	0.5806	38.3620	6.6012	9.9882
2016/3/23	0.8087	0.2666	1	0.5806	39.0810	4.8924	10.3078
2016/3/24	0.9153	0.2666	1	0.5806	39.7960	5.6386	11.9552
2016/3/25	1.0484	0.2666	1	0.5806	40.5070	6.5740	13.8479
2016/3/26	1.0617	0.2666	1	0.5806	41.2140	6.7736	15.6557
2016/3/27	0.9264	0.2666	1	0.5806	41.9170	6.0112	17.6013
2016/3/28	1.1278	0.2666	1	0.5806	42.6160	7.4400	21.1701
2016/3/29	1.1116	0.2666	1	0.5806	43.3110	7.4528	25.0961
2016/3/30	0.8087	0.2666	1	0.5806	44.0020	5.5085	30.5156
2016/3/31	0.9153	0.2666	1	0.5806	44.6890	6.3319	42.4824
2016/4/1	1.0484	1.4543	1	1.0000	45.3720	69.1781	
2016/4/2	1.0617	1.4543	1	1.0000	46.0510	71.1041	
2016/4/3	0.9264	1.4543	1	1.0000	46.7260	62.9522	
2016/4/4	1.1278	1.4543	2.0031	1.0000	47.3970	155.7183	
2016/4/5	1.1116	1.4543	1.1096	1.0000	48.0640	86.2162	
2016/4/6	0.8087	1.4543	1.1673	1.0000	48.7270	66.8950	
2016/4/7	0.9153	1.4543	1	1.0000	49.3860	65.7387	
2016/4/8	1.0484	1.4543	1	1.0000	50.0410	76.2969	
2016/4/9	1.0617	1.4543	1	1.0000	50.6920	78.2700	
2016/4/10	0.9264	1.4543	1	1.0000	51.3390	69.1672	
2016/4/11	1.1278	1.4543	1	1.0000	51.9820	85.2588	
2016/4/12	1.1116	1.4543	1	1.0000	52.6210	85.0671	
2016/4/13	0.8087	1.4543	1	1.0000	53.2560	62.6340	
2016/4/14	0.9153	1.4543	1	1.0000	53.8870	71.7301	
2016/4/15	1.0484	1.4543	1	1.0000	54.5140	83.1168	