

## 同济大学数学建模竞赛题 B



参赛队员：

参赛时间：2016 年 4 月 27 号——5 月 4 日

# 游乐园客流疏导方案

## 【摘要】

本文针对 Youth 游乐园内的游客游园体验优化问题和皇冠假日酒店 2016 年 1-3 月预定房间数预测问题建立了游客疏导模型和预订房间数预测模型,分别采用了最短路径算法、最优解原理和指数平滑预测法、模糊综合评价法,利用 MATLAB 和 C++ 进行编程、绘图和分析,给出了使得游客游园体验最优的疏导方案以及 2016 年 1-3 月的每日预定房间数预测结果。

针对问题一,利用最短路径 Dijkstra 算法和最优解原理,建立了游客疏导模型。首先利用 MATLAB 编程,采用 Dijkstra 算法,分别求出各点到其他点的最短距离,由此估计游客在转移过程中消耗的时间。再加上疏导目的地此刻还需等待场数乘以每场时间和项目目前进行到结束的时间,构成判断当前地点推荐路线的时间指标  $Time_i$ [10]。同时设置各地的预警容量,若达到预警容量,即使所耗时间短,仍不会再引导游客前往此地。最后,利用 C++ 编程,选取使得游客消耗时间最少的项目点即为游园体验最优的疏导建议。

本方案的创新点在于,选用游客等待和移动所消耗的时间作为考量游客游园体验的指标,但考虑到游乐园内的项目持续时间差异较大,不仅仅只是考虑使得消耗时间最短,还加入了不容许超过预警人数上限的条件,因此模型设计较为合理。

针对问题二,利用指数平滑预测法和模糊综合评价法,建立了酒店预定房间数预测模型。首先选用一次指数平滑法,根据 2015 年 1-12 月的数据,在不同平滑系数  $\alpha$  的条件下,用 MATLAB 编程并作出图形,进行误差分析,并选择出拟合程度最好的  $\alpha$ 。用此  $\alpha$  作为二次指数平滑法的平滑系数,预测 15 年 1-12 月的数据以及 16 年 1-3 月的数据初步预测结果。根据月份(季节)影响因子修正所得结果,得到 16 年 1-3 月每月的预测房间数。然后平均分配到月内的若干周,再根据从 15 年 1-12 月的数据总结出来的一周每日预订数变化趋势来预测每天的预定房间数。最后根据从 15 年 1-12 月的数据总结出来的节假日影响因子修正结果,得出 16 年 1-3 月每日预定房间数的预测结果。

本方案的创新点在于,用指数平滑预测法对短期内的月份数据进行了预测,再细分到周以及日,减小了直接考虑每日预定量可能造成的误差。同时考虑了节假日,周末,月份(季节)等对预定房间数可能造成的影响,对指数平滑法预测出来的结果进行了模糊综合修正,使得预测结果较为准确,因此模型设计较为合理。

据本文所给出的两个模型,可在实际应用中加以推广。对于游乐园的游客加以提醒和疏导,使游客获得最佳游园体验,同时使游乐园内不会太过拥堵。同时,初步预测酒店的每日预定房间数,对于酒店的数据分析来说意义重大。

【关键词】: 最短路径算法 最优解 指数平滑预测 模糊综合评价

## 一、问题的重述

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，收到了青少年的热捧。预计届时园区将迎来每天一万的大客流。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。

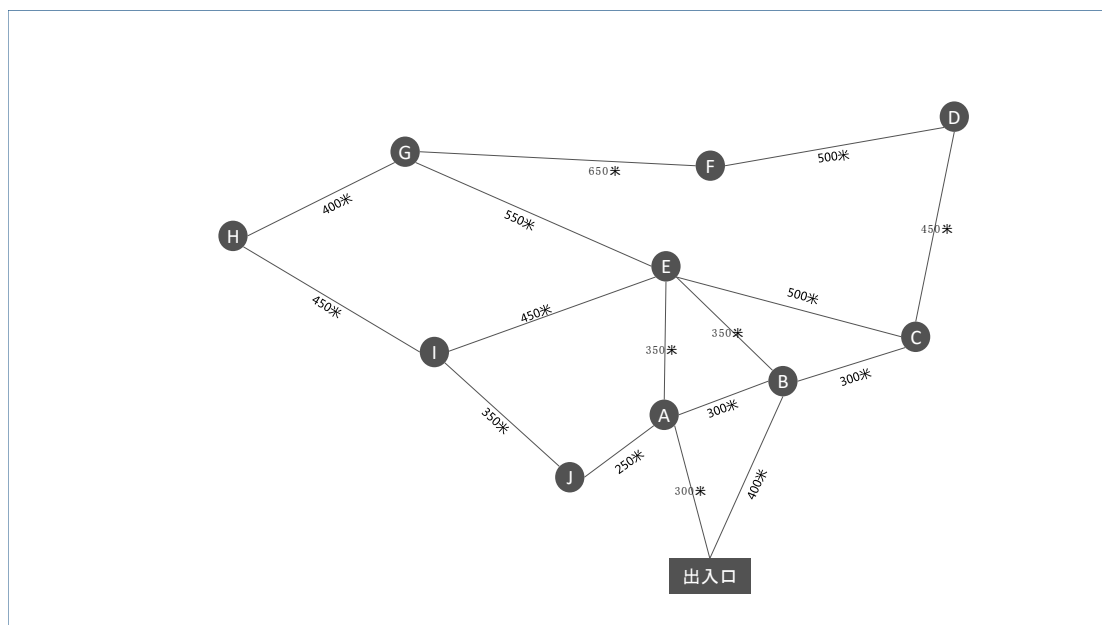
试就园区的整体规划，建立数学模型分析下面的问题：

- (1) Youth 乐园的规划图，共有 A-J 共有 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。

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

图 1：Youth 游乐园的规划图



- (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。请根据该酒店的历史预定数据信息，综合考虑影响房间预订量的主要因素（比如季节，工作日/周末，法定假日，暑假等）建立数学模型。并根据酒店 2015 年全年预订数据，预测 2016 年 1 月至 3 月每天预订房间数。

## 二、模型的基本假设

1. 根据地图，默认每个游客选择第  $i$  个项目点到第  $j$  个项目点的最短路径，到达下一个项目点。
2. 每时刻在每个项目点，离正在进行的项目结束的时间和排队等待的人数是已知的。
3. 所有游客步行速度为一个定值。
4. 每个项目点排队人数都足够多。
5. 2016年周末/工作日，季节，暑期，节假日等因素对预订房间的影响与2015年的大致相同。
6. 2016年1~3月酒店没有店庆等大型活动等未知事件发生。
7. 2016年的月影响因子，日影响因子，非规则因素基本稳定。

## 三、符号说明

模型一的符号说明		
符号	表示意义	备注
$A_0$	入口	
$A_i$	第 $i$ 个项目点	题目中有 $A \sim J$ 共 10 个项目点，为了方便，我们将其表示为 $A_0 \sim A_{10}$
$S_i$	第 $i$ 个项目点能容纳游客数	
$T_i$	第 $i$ 个项目点的每场持续时间	
$P_i$	第 $i$ 个项目点的当前等待人数	每时刻，每个项目点，排队等待的人数是已知的，即 $P_1 \sim P_{10}$ 的值是已知的
$T_{li}$	第 $i$ 个项目剩余的时间	每时刻，在每个项目点，离正在进行的项目结束的时间是已知的
$t_{ij}$	第 $i$ 个项目点到第 $j$ 个项目点的所用的时间	考虑到了进游乐园的青少年比较多，综合考量各个年龄段的步行速率，我们综合选出了一个平均速度，根据这个平均速度推出第 $i$ 个项目点到第 $j$ 个项目点的参考时间
$Time_i$	第 $i$ 个项目点的推荐最优路线所需要的时间	
$Time_{ij}$	第 $i$ 个项目点到到体验第 $j$ 个项目点大致所需要的时间	
$MAX_i$	每个项目点的排队容量预警值	即当此项目点的人数超过该容量时，该项目点将暂时不作为可选择的最优路线
Judge[i]	判断第 $i$ 个项目点的人数是否饱和，即是否超过排队容量预警人数	如果 $P_i \geq MAX_i$ , Judge[i]=9999999 (即表示无穷) 如果 $P_i \leq MAX_i$ , 令 Judge[i]=1 (表示对原数没有影响)

模型二的符号说明		
符号	表示意义	备注
$y^{t+1}$	t+1 期的预测值	本期（t 期）的平滑值 $S_t$
$y^t$	t 期的实际值	
$y^t$	t 期的预测值	上期的平滑 $S_{t-1}$ 值
$S_t^{(1)}$	第 t 周期的一次指数平滑值	
$S_t^{(2)}$	第 t 周期的二次指数平滑值	
$S_{t-1}^{(2)}$	第 t-1 周期的二次指数平滑值	
$\alpha$	加权系数（也称为平滑系数）	
$m$	月影响因子	
$d$	日影响因子	
$y_t$	t 月份的预测值	
$y_t'$	经修正后的预测值	

## 四、问题的分析及模型的建立

### 问题一的数据分析和模型建立

园区每天会迎来一万人的一大客流，根据每个项目点能够容纳的游客数，每个项目每场的持续时间，该项目点正在等待的游客数目，离正在进行的项目结束的时间等多方面因素，分析并为游客提供一个较优的选择方案，既分流人群，又使游客等待时间尽可能短，从而保障游客的游园体验。

第一，园区的线路规划为一个无向图，我们通过 Dijkstra 算法来计算出第 i 个项目点  $A_i$  到第  $A_j$  个项目点的最短路径，算出游客在路上的时间  $t_{ij}$ 。

第二，为每一个项目点设置一个排队预警容  $MAX_i$  量，判断每个项目点的人数是否超过该容量，若当此项目点的人数超过该容量时，该项目点将暂时不作为可选择的最优路线，即令 Judge[i] 为无穷。

第三，对于第 i 个项目点，分别计算出剩余其他九个项目的当场项目剩余时间  $T_{ij}$ ，此刻其他九个项目点的排队时间  $\left[\frac{P_j}{S_j}\right]^{[1]} * t_j$ ，以及从 i 到这九个项目地点路上所用时间  $t_{ij}$  之和  $Time_{ij}$ ，对于其自身来说， $j=i$ ，即  $t_{ij}=0$ 。

此时，  $Time_{ij} = \left[\frac{P_j}{S_j}\right]^{[1]} * t_j + T_{ij}$  （ [1] 表示对商取不大它的最大整数 ）

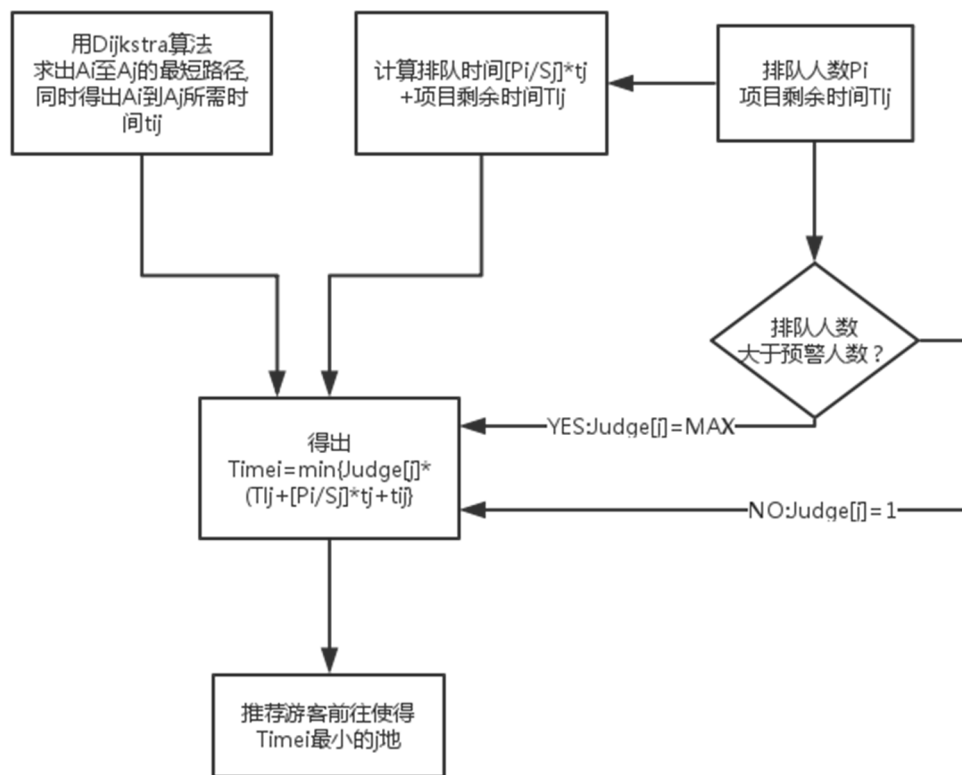
第四，将上述求得的时间和进行比较，选择最短时间  $Time_i$  所对应的项目点，即为此刻的推荐路线，将最优路线显示在显示屏上。

由上述分析可得，我们所建立的数学模型为

$$Time_i = \min \{ Judge[j] * (T_{ij} + [\frac{P_j}{S_j}] * t_j + t_{ij}) \} \quad (1 \leq j \leq 10)$$

综上，我们首先根据拓扑图建立最短路径模型，利用图论算法中的最短路径算法找出从 i 项目点到 j 项目点的最短路径，其次根据以上公式模型找出等待时间最短的推荐地点，并判断所求得地点的排队人数是否超过预警人数，最终为游客提供推荐路线，尽量保障游客等待时间短，并且达到分流人群的目的。

图 2：模型 1 的程序框图



## 问题一的求解

### (一): 方案模型及算法设计

#### 一、计算出 $t_{ij}$

利用 Dijkstra 算法计算出第  $i$  个项目点  $A_i$  到第  $A_j$  个项目点的最短路径，估计游客在路上的时间 $t_{ij}$ 。

具体步骤:

将图3中所有的顶点分成两个顶点集合S和D。以 $i$ 为源点已经确定了最短路径的终点并入S集合中，S初始时只含顶点 $i$ ，D则是尚未确定到源点 $i$ 最短路径的顶点集合。然后每次从D集合中选择S集合点到D路径最短的那个点，并加入到集合S中，并把这个点从集合D删除。直到D集合为空为止。

1、选一项目点 $i$ 为源点，并视从源点 $i$ 出发的所有边为到其余九个顶点的最短路径（确定数据结构：因为求的是最短路径，所以①就要用一个记录从源点 $i$ 到其它各顶点的路径长度数组 $dist[]$ ，开始时， $dist$ 是源点 $i$ 到顶点 $j$ 的直接边长度，即 $dist$ 中记录的是邻接阵的第 $i$ 行。②设一个用来记录从源点到其它顶点的路径数组 $path[]$ ， $path$ 中存放路径上第 $j$ 个顶点的前驱顶点）。

2、在上述的最短路径 $dist[]$ 中选一条最短的，并将其终点（即 $\langle i, j \rangle$ ） $j$ 加入到集合 $s$ 中。

3、调整D中各顶点到源点 $i$ 的最短路径。因为当顶点 $j$ 加入到集合 $s$ 中后，源点 $i$ 到剩余的其它顶点 $j$ 就又增加了经过顶点 $k$ 到达 $j$ 的路径，这条路径可能要比源点 $v$ 到 $j$ 原来的最短的还要短。调整方法是比较 $dist[k]+g[k, j]$ 与 $dist[j]$ ，取其中的较小者。

4、再选出一个到源点 $i$ 路径长度最小的顶点 $k$ ，从D中删去后加入S中，再回去到第三步，如此重复，直到集合S中的包含图中所有顶点。

图3：园区规划无向图

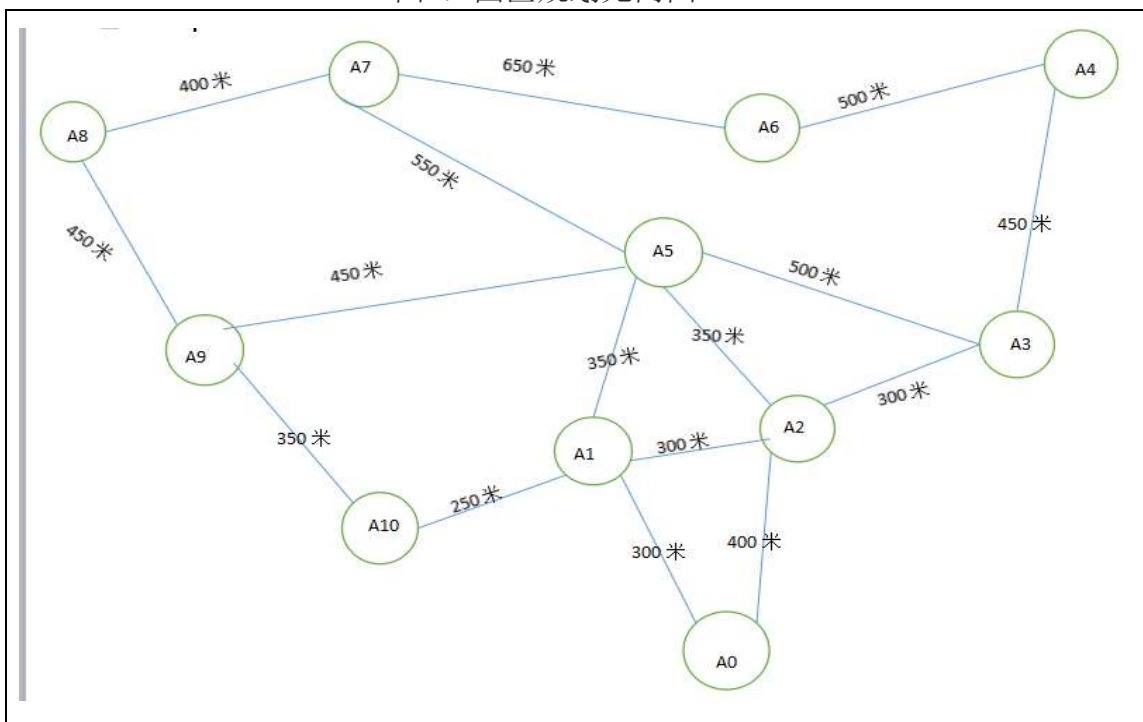


表2：园区项目点距离矩阵（i→j）

项目点 j 项目点 i	A0	A/A1	B/A2	C/A3	D/A4	E/A5	F/A6	G/A7	H/A8	I/A9	J/A10
A0	0	300	400	∞	∞	∞	∞	∞	∞	∞	∞
A/A1	300	0	300	∞	∞	350	∞	∞	∞	∞	250
B/A2	400	300	0	300	∞	350	∞	∞	∞	∞	∞
C/A3	∞	∞	300	0	450	500	∞	∞	∞	∞	∞
D/A4	∞	∞	∞	450	0	∞	500	∞	∞	∞	∞
E/A5	∞	350	350	500	∞	0	∞	550	∞	450	∞
F/A6	∞	∞	∞	∞	500	∞	0	650	∞	∞	∞
G/A7	∞	∞	∞	∞	∞	550	650	0	400	∞	∞
H/A8	∞	∞	∞	∞	∞	∞	∞	400	0	450	∞
I/A9	∞	∞	∞	∞	∞	450	∞	∞	450	0	350
J/A10	∞	250	∞	∞	∞	∞	∞	∞	∞	350	0

## 二、判断是否超过预警人数 $MAX_i$

为每一个项目点设置一个排队预警容 $MAX_i$ 量 ‘

如果 $P_i \geq MAX_i$ ，令  $Judge[i] = 99999999$ （即表示无穷），即该路线时间为无穷，不再作为可选择的推荐路线。

如果 $P_i \leq MAX_i$ ，令  $Judge[i] = 1$ （即表示对原数没有影响）

## 三、求出时间之和 $Time_{ij}$

$$Time_{ij} = Judge[j] * (T_{ij} + [\frac{P_j}{S_j}] * t_j + t_{ij}) \quad (1 \leq j \leq 10)$$

## 四、求最短时间 $Time_i$ ，

$$Time_i = \min \{ Judge[j] * (T_{ij} + [\frac{P_j}{S_j}] * t_j + t_{ij}) \quad (1 \leq j \leq 10)$$

选择最短时间 $Time_i$ ，所对应的项目点，即为此刻的推荐路线，将最优路线显示在每个项目点的显示屏上。

## （二）：方案模型的模拟实践与例证

根据相关数据显示，人的步行平均速度 $v=4\text{km/h}=66.7\text{m/min}$   
据程序（算法及相关程序见附录一）运行结果归纳如下：



表3：园区项目点之间步行所需最短时间

项目点 i/min 项目点 j/min	A/A0	B/A2	C/A3	D/A4	E/A5	F/A6	G/A7	H/A8	I/A9	J/A10
A/A1	0	4.498	8.996	15.742	5.247	23.238	13.493	15.742	8.996	3.748
B/A2	4.498	0	4.498	11.244	5.247	18.741	13.493	19.490	11.994	8.246
C/A3	8.996	4.498	0	6.747	7.496	14.243	15.742	20.990	14.243	12.744
D/A4	15.742	11.244	6.747	0	14.243	7.496	17.241	23.238	20.990	19.490
E/A5	5.247	5.247	7.496	14.243	0	17.991	8.246	13.493	6.747	8.996
F/A6	23.238	18.741	14.243	7.496	17.991	0	9.745	15.742	22.489	27.736
G/A7	13.493	13.493	15.742	17.241	8.246	9.745	0	5.997	12.744	17.991
H/A8	15.742	19.490	20.990	23.238	13.493	15.742	5.997	0	6.746	11.994
I/A9	8.996	11.994	14.243	20.990	6.747	22.489	12.744	6.746	0	5.247
J/A10	3.748	8.246	12.744	19.490	8.996	27.736	17.991	11.994	5.247	0

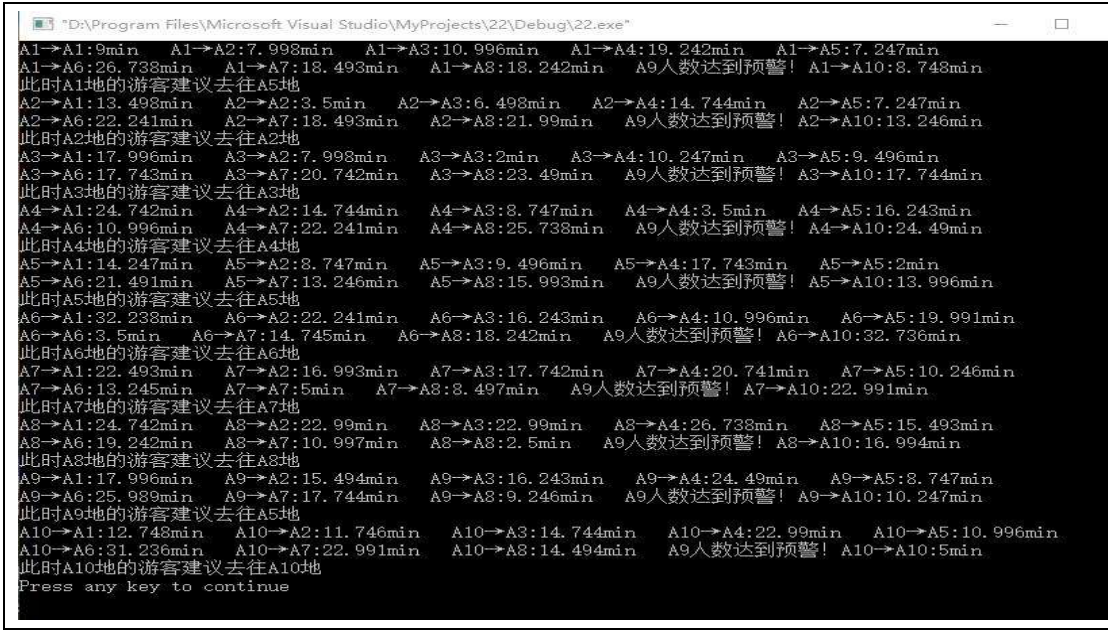
表 4：实例数据

$A_i$	$S_i/\text{人}$	$T_i/\text{min}$	$P_i/\text{人}$	$T_{li}/\text{min}$	$Max_i/\text{人}$
A1	400	33	300	9	700
A2	30	1.25	60	1	250
A3	50	2.5	40	2	300
A4	30	2.5	50	1	250
A5	100	5.0	90	2	450
A6	50	2.5	80	1	300
A7	30	2	60	1	250
A8	30	1.5	40	1	250
A9	20	1.5	250	1	200
A10	50	2	100	1	300

ime/min	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	最优解
A1→	9	7.998	10.996	19.242	7.247	26.738	18.493	18.242	!!	8.748	A5
A2→	13.498	3.5	6.498	14.744	7.247	22.241	18.493	21.99	!!	13.246	A2
A3→	17.996	7.998	2	10.247	9.496	17.743	20.742	23.49	!!	17.744	A3
A4→	24.742	14.744	8.747	3.5	16.243	10.996	22.241	25.738	!!	24.49	A4
A5→	14.247	8.747	9.496	17.743	2	21.491	13.246	15.993	!!	13.996	A5
A6→	32.238	22.241	16.243	10.996	19.991	3.5	14.745	18.242	!!	32.736	A6
A7→	22.493	16.993	17.742	20.741	10.246	13.245	5	8.497	!!	22.991	A7
A8→	24.742	22.99	22.99	26.738	15.493	19.242	10.997	2.5	!!	16.994	A8
A9→	17.996	15.494	16.243	24.49	8.747	25.989	17.744	9.246	!!	10.247	A5
A10→	12.748	11.746	14.744	22.99	10.996	31.236	22.991	14.494	!!	5	A10

注：A9 排队人数超过了排队容量预警值，用!! 表示，不参与最优路线选择

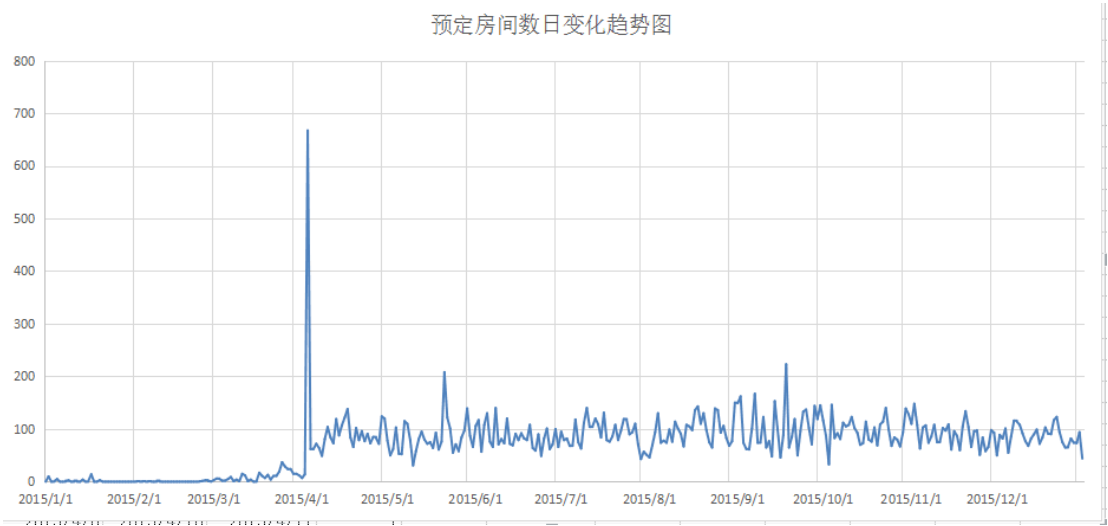
图 4：根据所给实例数据的 c++运行结果（程序见附录二）



问题二的数据分析和模型的建立

根据皇冠假日酒店 2015 年全年预定数据信息，综合考虑到影响房间预订量的主要因素，如季节、工作日/周末、法定假日，暑假等，并利用指数平滑法建立数学模型，预测 2016 年 1 月到 3 月的每天预定房间数。

图 5：2015 年预定房间数日变化趋势图



(一)：用指数平滑法得到 2016 年 1 月、2 月、3 月每月预定房间数的结果

1：用一次指数平滑法，选择不同的平滑常数，即加权系数 $\alpha$ ，并分别作出在不同的平滑常数下预测的 1~12 月的每月房间预定量和实际的每月房间预定量的

折线图，观察拟合效果，选择拟合效果最好的 $\alpha$ 出来。用此 $\alpha$ 作为二次指数平滑法的平滑系数，预测 1~12 月的预定量。再用二次指数平滑法的预测公式算出 2016 年 1-3 月的预定量。

指数平滑法是移动平均法中的一种，其特点在于给过去的观测值不一样的权重，即较近期观测值的权数比较远期观测值的权数要大。根据平滑次数不同，指数平滑法分为一次指数平滑法、二次指数平滑法和三次指数平滑法等。但它们的基本思想都是：预测值是以前观测值的加权和，且对不同的数据给予不同的权数，新数据给予较大的权数，旧数据给予较小的权数。

①一次指数平滑预测：其预测公式为：

$$y^{t+1'} = \alpha y^t + (1-\alpha) y^{t'}$$

②在一次指数平滑的基础上得二次指数平滑 的计算公式为：

$$S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)}$$

③ 预测公式：

$$\begin{aligned} \hat{Y}_{t+T} &= a_t + b_t \cdot T \\ S_t^{(2)} &= a S_t^{(1)} + (1-a) S_{t-1}^{(2)} \\ \begin{cases} a_t = 2S_t^{(1)} - S_t^{(2)} \\ b_t = \frac{a}{1-a} (S_t^{(1)} - S_t^{(2)}) \end{cases} \end{aligned}$$

2：根据已经给出的 2015 年数据作出预定房间数月变化趋势图，考虑到季节，暑假等因素，并参考多名专家的决策经验与我们对数据的深度分析，从中提炼出月影响因子  $m\{m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9, m_{10}, m_{11}, m_{12}\}$ ；并用通过提炼出的月影响因子对之前 2-12 月的预测数据进行修正并验证，再对 2016 年 1~3 月的预测数据进行修正。

## （二）计算得到 2016 年 1 月，2 月，3 月每日房间预订数的结果

1：将（一）中的 1 月、2 月、3 月每月房间预定数分别按周分配，已知 1 月 1 日到 1 月 3 日为周五到周六，1 月 4 日到 1 月 31 日恰为四周，即一月总共为  $4\frac{3}{7}$  周，2 月 1 日到 2 月 28 日恰为四周，2 月 29 日是周一，即二月共有  $4\frac{1}{7}$  周；3 月 1 号到 3 月 6 号为周二到周日，3 月 28 日到 3 月 31 日为周一到周四即三月共有  $(\frac{6}{7}+3+\frac{4}{7})$  周，分配后可得出 1 月、2 月、3 月每月各周的房间预定数。

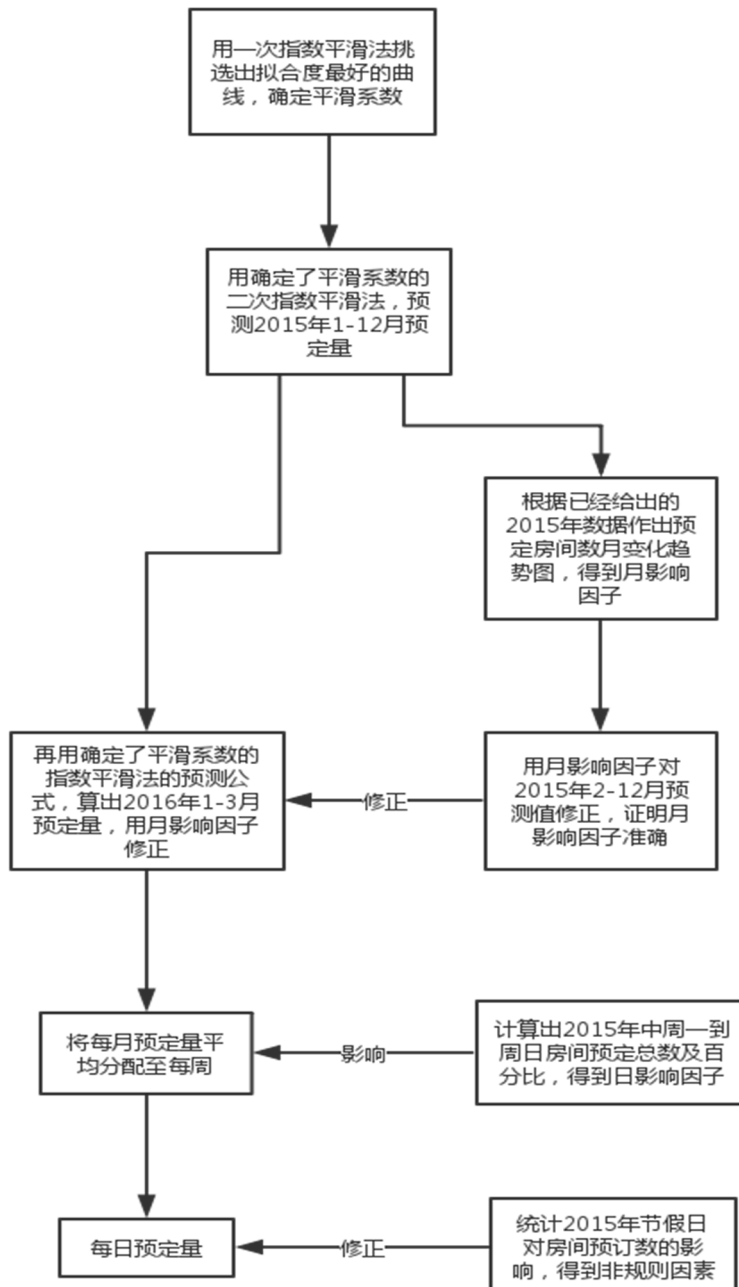
2：做一个日变化趋势图，即计算出 2015 年中在周一，周二，周三，周四，周五，周六，周日的各自所有的日房间预定数，若只考虑到工作日，周末对预定数目的影响，参考多名专家的的决策经验和我们对数据的深度分析，从中提炼出日影响因子，即  $d\{d_1, d_2, d_3, d_4, d_5, d_6, d_7\}$ 。通过考虑日影响因子，得出 1 月、2 月、3 月每一天的房间预订量的初步结果。

3：作出每天预订房间数与提前预订天数（到店时间和预定时间之差）的散点图，从而得出顾客提前预订房间的规律，并得出二者之间的函数。由 2015 年节假日和 2015 年酒店房间预订高峰期的关系，得出非规则因素，即节假日对于日

预订房间数的影响因子，根据该非规则因素，从而修正 2016 年 1 月，2 月，3 月的每天预订房间数，即为最终预测结果。

综上，我们的模型即用指数平滑法首先选出拟合程度最高的平滑常数，其次分析影响每月预定房间数目的因素，求出季节，暑假对月预定房间数目的影响因子，综合考虑从而预测出 2016 年 1 月，2 月，3 月的月房间预定量。其次，将每月的房间预订量按周平均分配，考虑工作日，周末对日预定房间数目的影响因子后，得出日预订房间数的初步结果，最后根据 2015 年节假日对日预定房间数目的影响计算出非规则因素，修正 2016 年的 1 月，2 月，3 月的每日房间预定量。

图 6：模型二的程序框图



问题二的求解

(一)：求得 2016 年 1 月、2 月、3 月每月预定房间数

1: 用一次指数平滑法(程序见附录三), 选择不同的平滑常数, 当 $\alpha=0.1, 0.3, 0.5, 0.8$  时, 分别作出预测的 1~12 月的每月房间预定量和实际的每月房间预定量的折线图, 观察拟合效果, 由下图可知当 $\alpha=0.8$  时, 拟合效果最好。

图 7:  $\alpha=0.1$  时, 2015 年各月预测与实际房间预定量的折线图

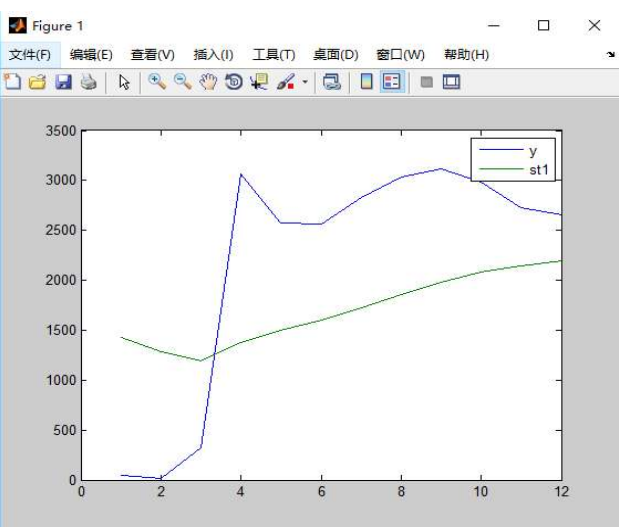


图 8:  $\alpha=0.1$  时, 2015 年各月预测与实际房间预定量的折线图

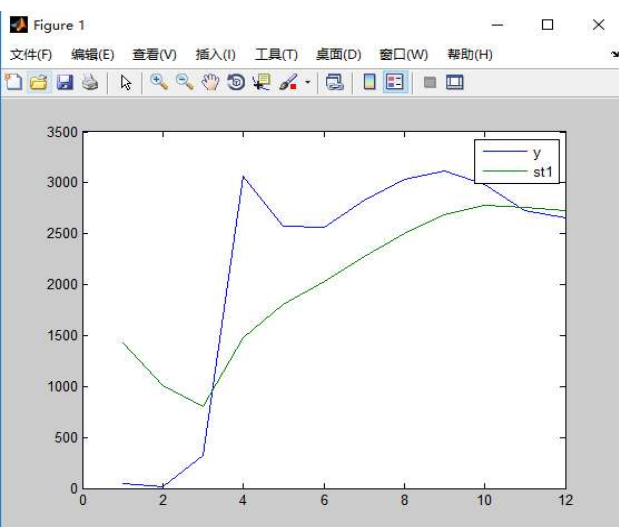


图 9:  $\alpha=0.5$  时, 2015 年各月预测与实际房间预定量的折线图

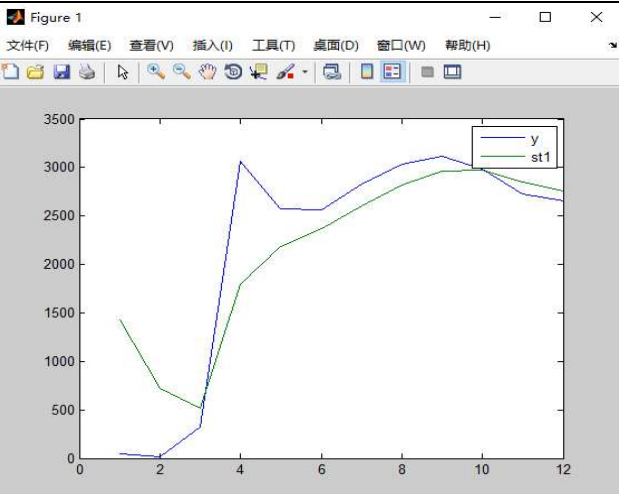
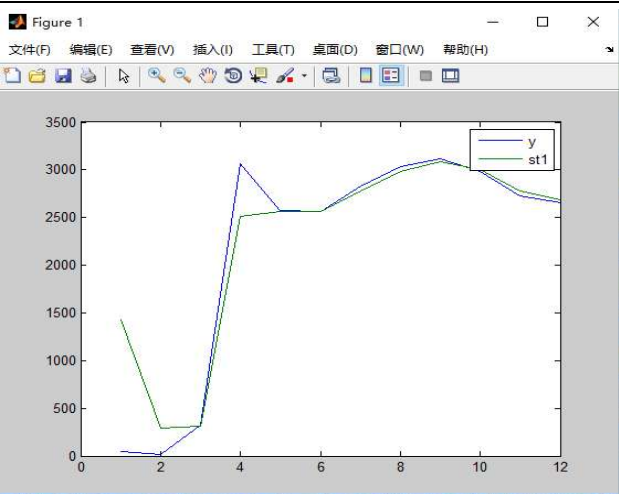


图 10:  $\alpha=0.8$  时, 2015 年各月预测与实际房间预定量的折线图

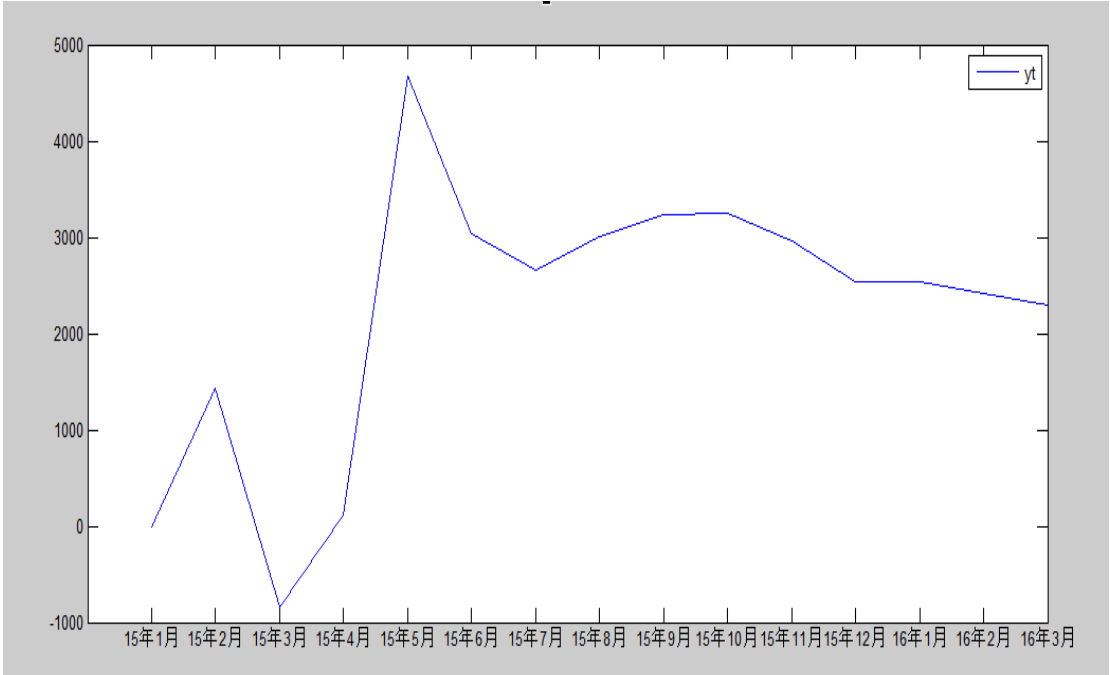


2. 再用二次指数平滑法（程序见附录四）的预测公式算出 2016 年 1-3 月的预定量。

表 5：中间数据

st1	error of st1	st2	yt	error of yt	yt'	error of yt'
1430	1384	1430	0	0		
295.6	283.6	522.48	1430	1418	14.3	2.3
315.12	-4.88	356.592	-838.8	-1158.8	671.04	351.04
2515.024	-549.976	2083.3376	107.76	-2957.24	150.024	-2914.976
2563.8048	-12.1952	2467.71136	4673.456	2097.456	4206.1104	1630.1104
2561.56096	0.56096	2542.79104	3044.272	483.272	2435.4176	-125.5824
2777.112192	-53.887808	2730.247962	2655.41056	-175.58944	2654.211	-176.789
2981.822438	-51.1775616	2931.507543	3011.433344	-21.566656	3613.7196	580.7196
3087.564488	-26.4355123	3056.353099	3233.396915	119.3969152	4526.7558	1412.7558
3007.112898	20.11289754	3016.960938	3243.621432	256.6214323	2594.8968	-392.1032
2780.62258	56.62257951	2827.890251	2957.872696	233.8726963	2662.0857	-61.9143
2681.724516	24.7245159	2710.957663	2544.284221	-112.7157788	2289.8556	-367.1444
	AVER ERROR		1-3 月预测	AVER ERROR	修正后 1-3 月预测	AVER ERROR
	89.25573925		2535.558781	15.22559742	76.0668	-5.5985
			2418.626192		24.1863	
			2301.693604		1841.3552	
月影响修正因子：[ 0.03, 0.01, 0.8, 1.4, 0.9, 0.8, 1.2, 1.2, 1.4, 0.8, 0.9, 0.9 ]						

图 11：2015 年 1 月—2016 年 3 月每月房间预订数量初步结果



3：根据已经给出的 2015 年数据作出预定房间数月变化趋势图，考虑到季节，

暑假等因素，并用通过月影响因子  
 $m[0.03, 0.01, 0.8, 1.4, 0.9, 0.8, 1.2, 1.2, 1.4, 0.8, 0.9, 0.9]$  修改提炼出的月影响因子对之前 2-12 月的数据进行修正并验证，再对 2016 年 1~3 月的预测数据进行修正。

图 12：2015 年预订房间月变化趋势图

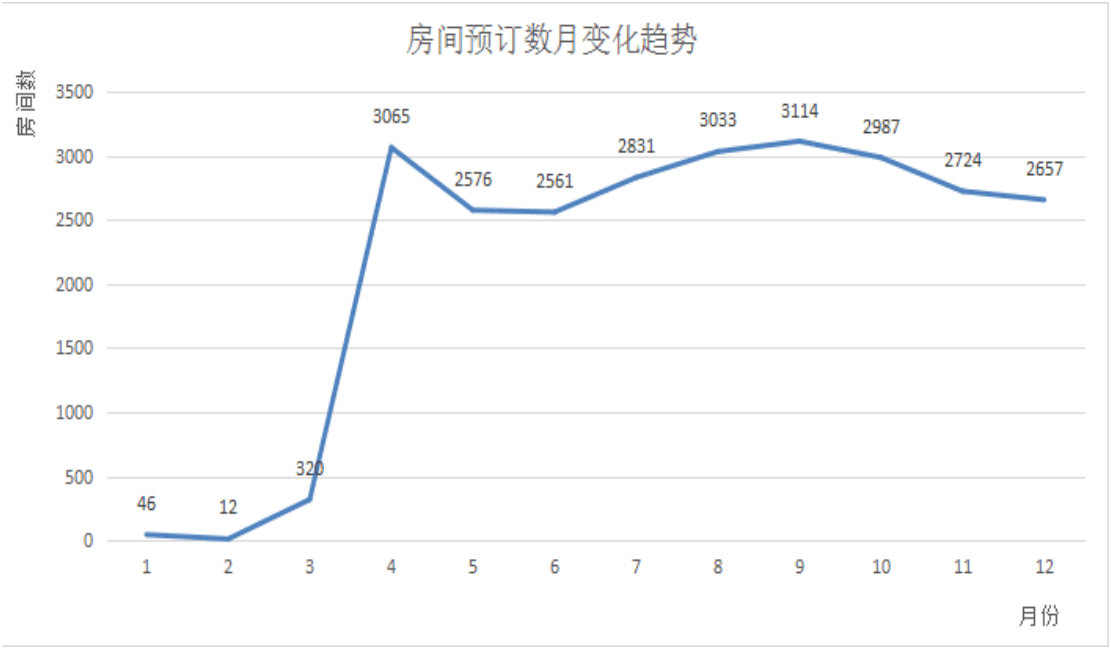
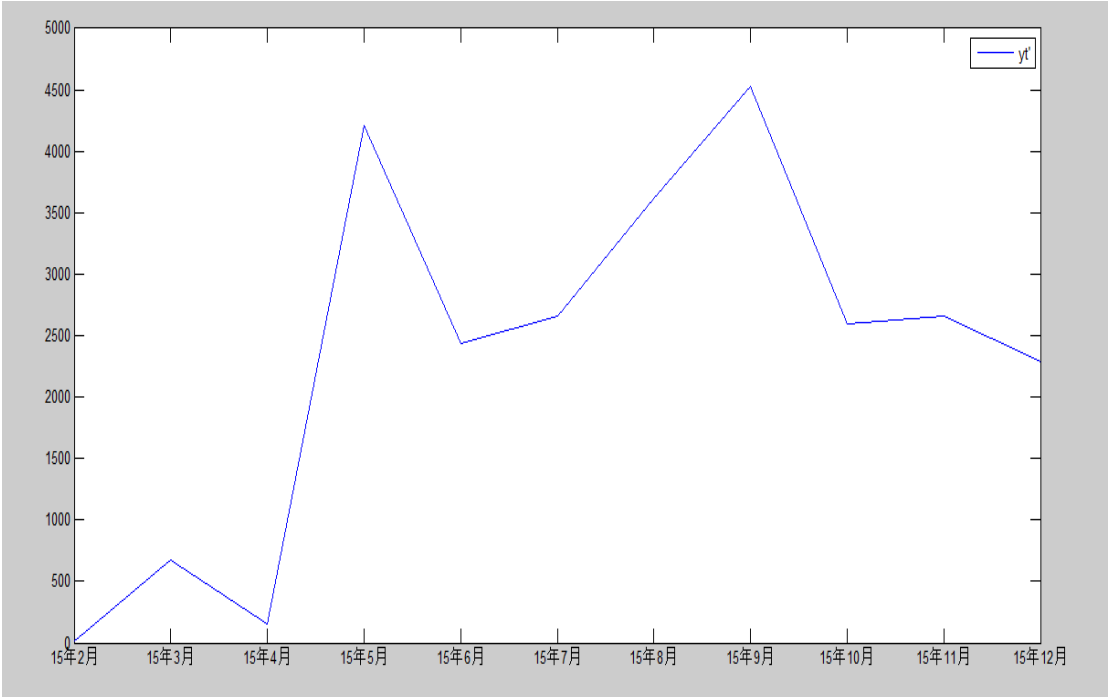


图 13：修正后 2015 年 2 月—2016 年 3 月的每月房间预订数量



(二) 求得 2016 年 1 月，2 月，3 月每日房间预订数的结果

1: 求得 2016 年 1 月, 2 月, 3 月每月的平均每周房间预定数

将（一）中的 1 月、2 月、3 月每月房间预定数分别按周分配，一月总共为  $4\frac{3}{7}$  周，二月共有  $4\frac{1}{7}$  周，三月共有  $(\frac{6}{7}+3+\frac{4}{7})$  周，分配后可得出 1 月、2 月、3 月各月的平均每周房间预定数。

表 6: 2016 年 1~3 月各月的平均每周预定房间数

时间	平均每周预订房间数
2016. 1	76. 0668
2016. 2	24. 1863
2016. 3	1841. 355

2: 求得 2016 年 1 月, 2 月, 3 月每天房间预定量的初步结果

做一个日变化趋势图，即计算出 2015 年中在周一，周二，周三，周四，周五，周六，周日的各自所有的日房间预定数，计算出日影响因子为  $d\{0.162925, 0.136272, 0.129252, 0.150042, 0.179627, 0.122888, 0.118993\}$ 。通过考虑日影响因子，得出 1 月、2 月、3 月每一天的房间预订量的初步结果。

图 14: 2015 年预定房间数日变化

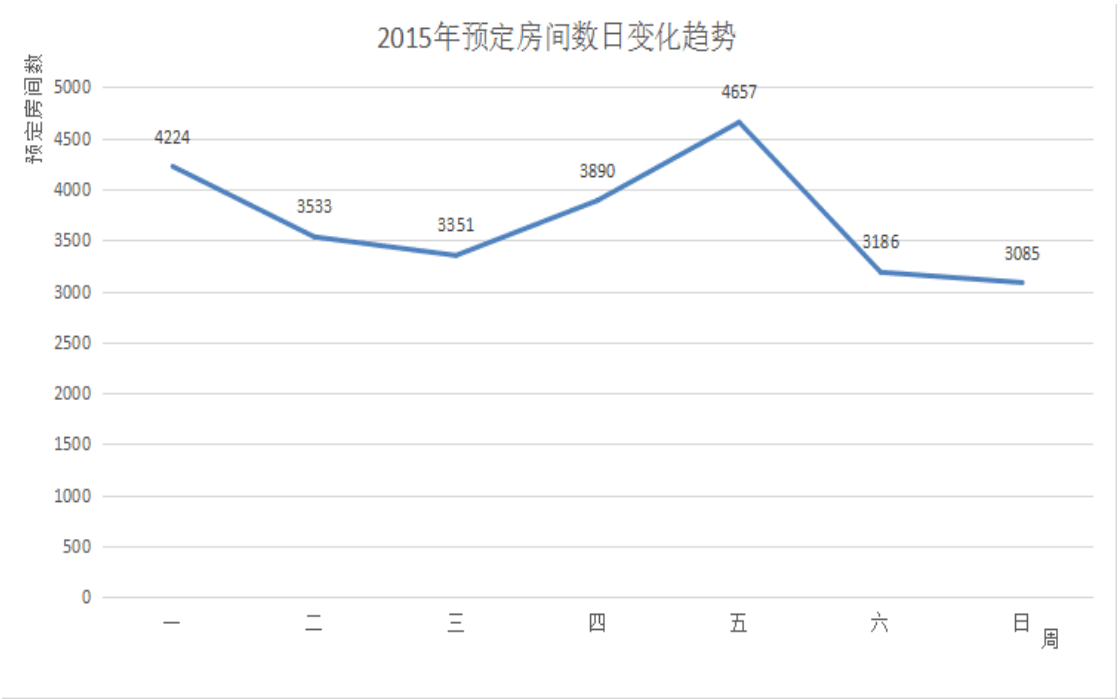


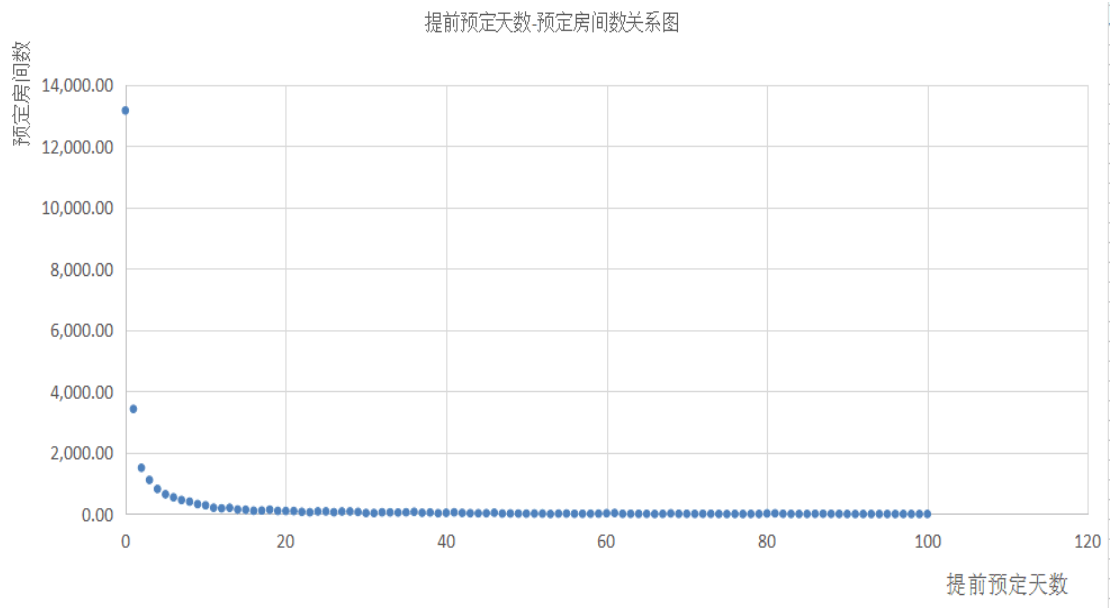


表 7：2016 年 1~3 月每天房间预定量的初步结果

日期	房间数	日期	房间数	日期	房间数	日期	房间数	日期	房间数
1 月 1 日	2.5891	1 月 21 日	2.5772	2 月 10 日	0.7546	3 月 1 日	58.0191	3 月 21 日	67.7427
1 月 2 日	1.7713	1 月 22 日	3.0853	2 月 11 日	0.8760	3 月 2 日	55.0303	3 月 22 日	56.6607
1 月 3 日	1.7152	1 月 23 日	2.1108	2 月 12 日	1.0487	3 月 3 日	63.8818	3 月 23 日	53.7419
1 月 4 日	2.7985	1 月 24 日	2.0439	2 月 13 日	0.7174	3 月 4 日	76.4775	3 月 24 日	62.3861
1 月 5 日	2.3407	1 月 25 日	2.7985	2 月 14 日	0.6947	3 月 5 日	52.3207	3 月 25 日	74.6870
1 月 6 日	2.2201	1 月 26 日	2.3407	2 月 15 日	0.9512	3 月 6 日	50.6620	3 月 26 日	51.0957
1 月 7 日	2.5772	1 月 27 日	2.2201	2 月 16 日	0.7956	3 月 7 日	67.7427	3 月 27 日	49.4759
1 月 8 日	3.0853	1 月 28 日	2.5772	2 月 17 日	0.7546	3 月 8 日	56.6607	3 月 28 日	66.9155
1 月 9 日	2.1108	1 月 29 日	3.0853	2 月 18 日	0.8760	3 月 9 日	53.7419	3 月 29 日	55.9688
1 月 10 日	2.0439	1 月 30 日	2.1108	2 月 19 日	1.0487	3 月 10 日	62.3861	3 月 30 日	53.0856
1 月 11 日	2.7985	1 月 31 日	2.0439	2 月 20 日	0.7174	3 月 11 日	74.6870	3 月 31 日	61.6243
1 月 12 日	2.3407	2 月 1 日	0.9512	2 月 21 日	0.6947	3 月 12 日	51.0957		
1 月 13 日	2.2201	2 月 2 日	0.7956	2 月 22 日	0.9512	3 月 13 日	49.4759		
1 月 14 日	2.5772	2 月 3 日	0.7546	2 月 23 日	0.7956	3 月 14 日	67.7427		
1 月 15 日	3.0853	2 月 4 日	0.8760	2 月 24 日	0.7546	3 月 15 日	56.6607		
1 月 16 日	2.1108	2 月 5 日	1.0487	2 月 25 日	0.8760	3 月 16 日	53.7419		
1 月 17 日	2.0439	2 月 6 日	0.7174	2 月 26 日	1.0487	3 月 17 日	62.3861		
1 月 18 日	2.7985	2 月 7 日	0.6947	2 月 27 日	0.7174	3 月 18 日	74.6870		
1 月 19 日	2.3407	2 月 8 日	0.9512	2 月 28 日	0.6947	3 月 19 日	51.0957		
1 月 20 日	2.2201	2 月 9 日	0.7956	2 月 29 日	0.8340	3 月 20 日	49.4759		

3：求得提前预订天数与预订房间数目的关系  
作出每天预订房间数与提前预订天数（到店时间和预定时间之差）的散点图，从而拟合顾客提前预订房间的规律，得出相应的函数。

图 15：提前预订天数—预定房间关系图



函数为:

General model Rat51:

$$f(x) = \frac{(p1*x^5 + p2*x^4 + p3*x^3 + p4*x^2 + p5*x + p6)}{(x + q1)}$$

Coefficients (with 95% confidence bounds):

p1 = -2.646e-05 (-4.432e-05, -8.607e-06)  
p2 = 0.006533 (0.002697, 0.01037)  
p3 = -0.5777 (-0.8593, -0.296)  
p4 = 22.32 (14.1, 30.54)  
p5 = -402.8 (-487, -318.7)  
p6 = 4892 (4706, 5078)  
q1 = 0.3713 (0.357, 0.3857)

Goodness of fit:

SSE: 1.071e+05  
R-square: 0.9994  
Adjusted R-square: 0.9994  
RMSE: 33.76

4: 由上图可得出, 当提前预定天数大于 100 天时, 游客预订房间数稳定在 0 处, 故 2016 年需要考虑的节假日高峰期仅有清明和五一。

图 16 : 2015 年清明假期对酒店预订的影响分布  
(数据分布见附录五)

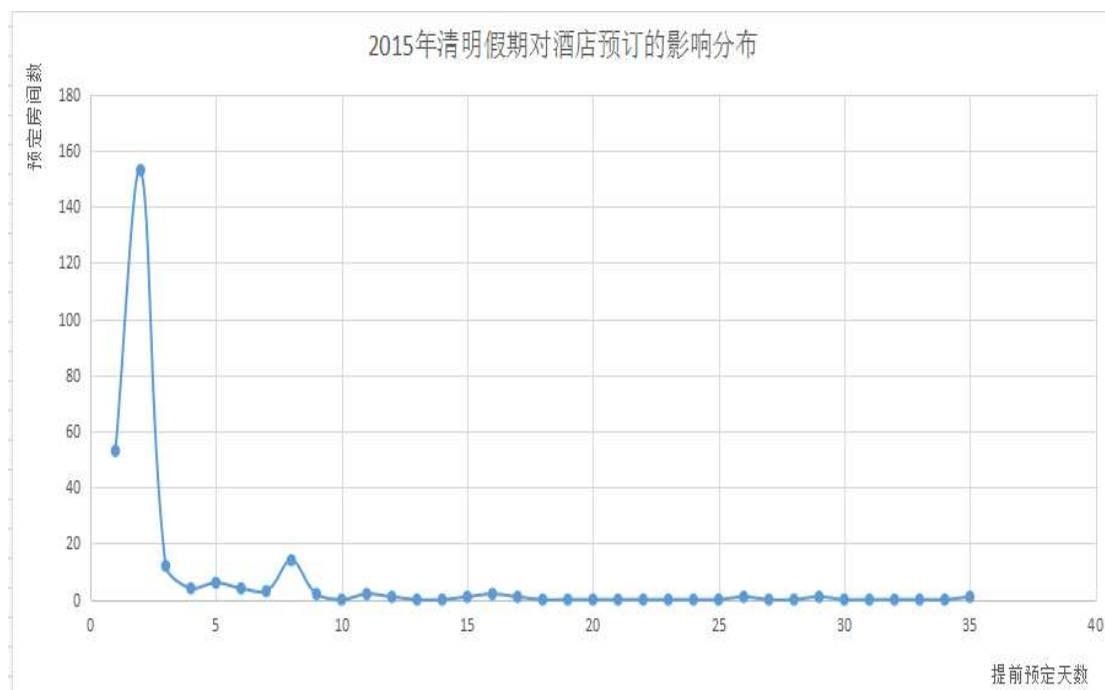
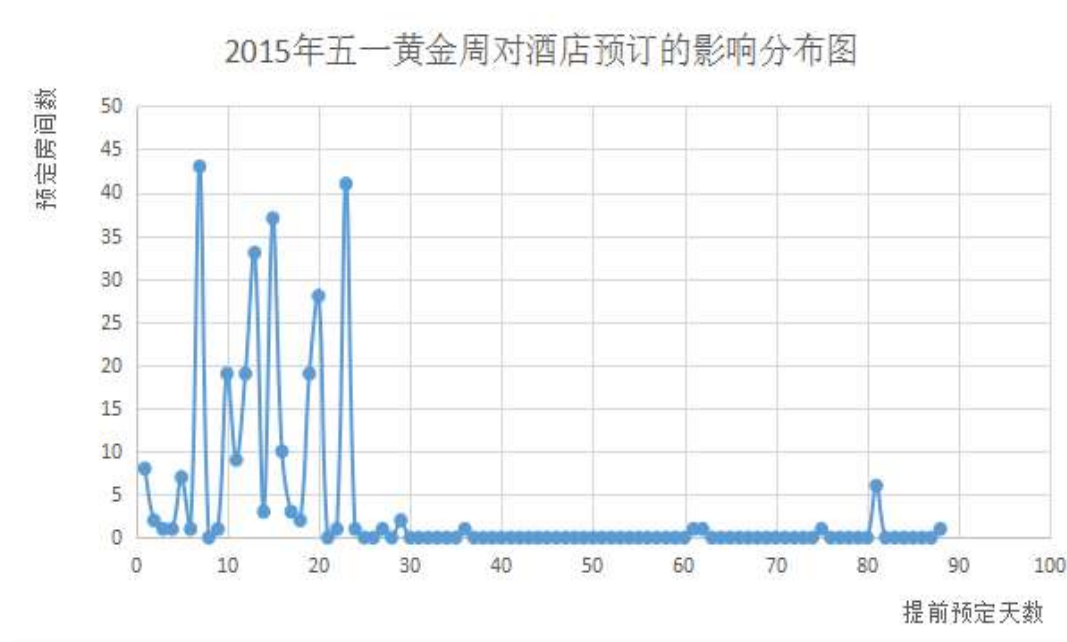


图 17：2015 年五一黄金周对酒店预订的影响分布图  
(数据记录见附录六)



5: 用节假日所造成的非规则因素来修正 2016 年 1 月, 2 月, 3 月的每天预订房间数, 即为最终预测结果。

图 18：2015 年 1 月~3 月的每天房间预定量

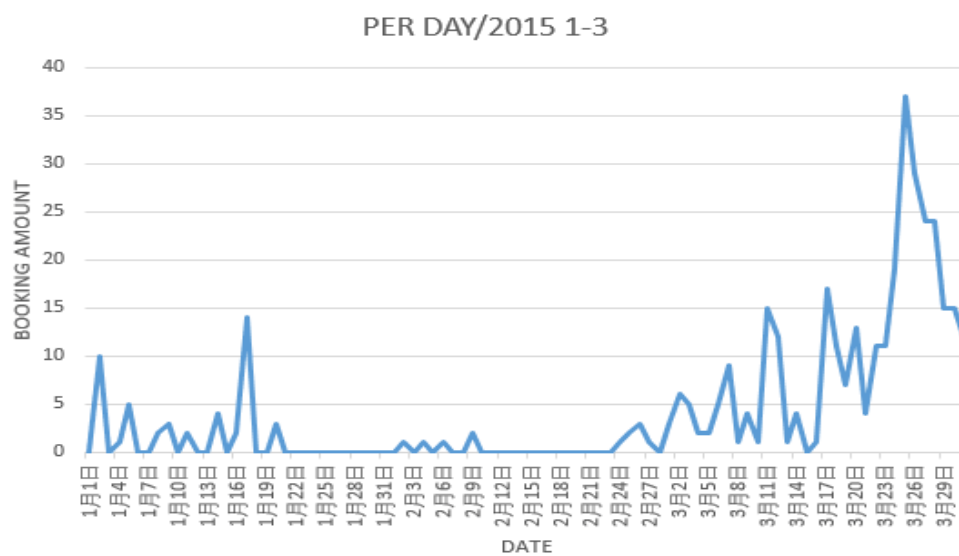


图 19：2016 年 1 月~3 月的每天房间预定量

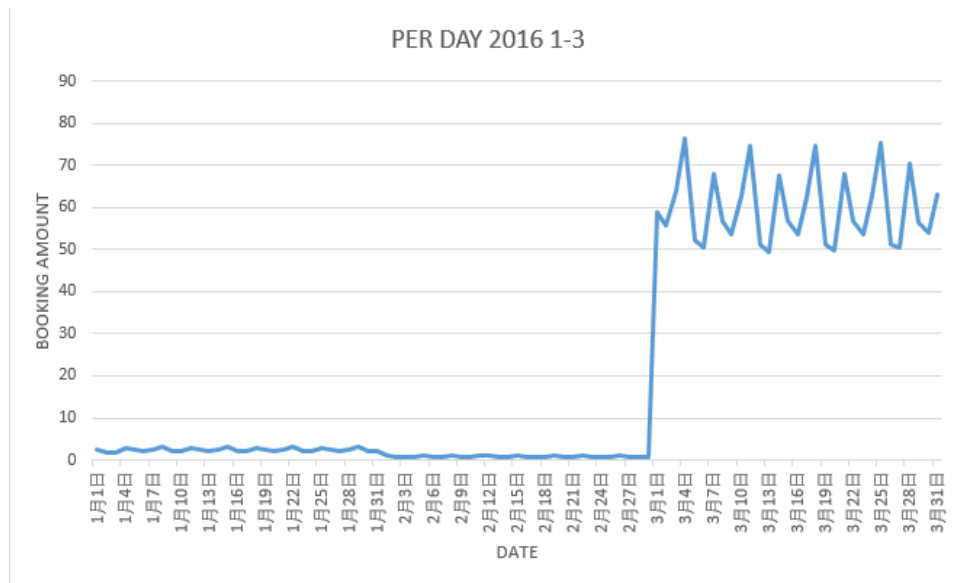


表 7：2016 年 1~3 月每天预订房间数最终结果

星期	日期	预定量	星期	日期	预定量	星期	日期	预定量
周五	1 月 1 日	2.5891	周一	2 月 1 日	0.9512	周二	3 月 1 日	58.9109
周六	1 月 2 日	1.7713	周二	2 月 2 日	0.7956	周三	3 月 2 日	55.6628
周日	1 月 3 日	1.7152	周三	2 月 3 日	0.7546	周四	3 月 3 日	63.8818
周一	1 月 4 日	2.7985	周四	2 月 4 日	0.8860	周五	3 月 4 日	76.4775
周二	1 月 5 日	2.3407	周五	2 月 5 日	1.0487	周六	3 月 5 日	52.3207
周三	1 月 6 日	2.2201	周六	2 月 6 日	0.7174	周日	3 月 6 日	50.6620
周四	1 月 7 日	2.5772	周日	2 月 7 日	0.6947	周一	3 月 7 日	68.0022
周五	1 月 8 日	3.0853	周一	2 月 8 日	0.9512	周二	3 月 8 日	56.6607
周六	1 月 9 日	2.1108	周二	2 月 9 日	0.7956	周三	3 月 9 日	53.7419
周日	1 月 10 日	2.0439	周三	2 月 10 日	0.7546	周四	3 月 10 日	62.6252
周一	1 月 11 日	2.7985	周四	2 月 11 日	0.9364	周五	3 月 11 日	74.6870
周二	1 月 12 日	2.3407	周五	2 月 12 日	1.0487	周六	3 月 12 日	51.0957
周三	1 月 13 日	2.2201	周六	2 月 13 日	0.7174	周日	3 月 13 日	49.4759
周四	1 月 14 日	2.5772	周日	2 月 14 日	0.6947	周一	3 月 14 日	67.7427
周五	1 月 15 日	3.0853	周一	2 月 15 日	0.9512	周二	3 月 15 日	56.6607
周六	1 月 16 日	2.1108	周二	2 月 16 日	0.7956	周三	3 月 16 日	53.7419
周日	1 月 17 日	2.0439	周三	2 月 17 日	0.7633	周四	3 月 17 日	62.3861
周一	1 月 18 日	2.7985	周四	2 月 18 日	0.8760	周五	3 月 18 日	74.6870
周二	1 月 19 日	2.3407	周五	2 月 19 日	1.0487	周六	3 月 19 日	51.2915
周三	1 月 20 日	2.2201	周六	2 月 20 日	0.7174	周日	3 月 20 日	49.8550
周四	1 月 21 日	2.5772	周日	2 月 21 日	0.6947	周一	3 月 21 日	68.0022
周五	1 月 22 日	3.0853	周一	2 月 22 日	0.9512	周二	3 月 22 日	56.6607
周六	1 月 23 日	2.1108	周二	2 月 23 日	0.7956	周三	3 月 23 日	53.7419
周日	1 月 24 日	2.0439	周三	2 月 24 日	0.7546	周四	3 月 24 日	62.6252
周一	1 月 25 日	2.7985	周四	2 月 25 日	0.8760	周五	3 月 25 日	75.2593

周二	1月26日	2.3407	周五	2月26日	1.0487	周六	3月26日	51.0957
周三	1月27日	2.2201	周六	2月27日	0.7174	周日	3月27日	50.4281
周四	1月28日	2.5772	周日	2月28日	0.6947	周一	3月28日	70.5048
周五	1月29日	3.0853	周一	2月29日	0.8340	周二	3月29日	56.6122
周六	1月30日	2.1108				周三	3月30日	53.8992
周日	1月31日	2.0439				周四	3月31日	63.0410

考虑到实际情况，每天预订房间数为整数，经过四舍五入之后，得到以下结果：

图 20：取整后 2016 年 1 月~3 月的每天房间预定量



表 8：取整后 2016 年 1~3 月每天预订房间数最终结果

星期	日期	预定量	星期	日期	预定量	星期	日期	预定量
周五	1月1日	3	周一	2月1日	1	周二	3月1日	59
周六	1月2日	2	周二	2月2日	1	周三	3月2日	56
周日	1月3日	2	周三	2月3日	1	周四	3月3日	64
周一	1月4日	3	周四	2月4日	1	周五	3月4日	76
周二	1月5日	2	周五	2月5日	1	周六	3月5日	52
周三	1月6日	2	周六	2月6日	1	周日	3月6日	51
周四	1月7日	3	周日	2月7日	1	周一	3月7日	68
周五	1月8日	3	周一	2月8日	1	周二	3月8日	57
周六	1月9日	2	周二	2月9日	1	周三	3月9日	54
周日	1月10日	2	周三	2月10日	1	周四	3月10日	63
周一	1月11日	3	周四	2月11日	1	周五	3月11日	75
周二	1月12日	2	周五	2月12日	1	周六	3月12日	51
周三	1月13日	3	周六	2月13日	1	周日	3月13日	49
周四	1月14日	3	周日	2月14日	1	周一	3月14日	68

周五	1月15日	2	周一	2月15日	1	周二	3月15日	57
周六	1月16日	2	周二	2月16日	1	周三	3月16日	54
周日	1月17日	3	周三	2月17日	1	周四	3月17日	62
周一	1月18日	2	周四	2月18日	1	周五	3月18日	75
周二	1月19日	2	周五	2月19日	1	周六	3月19日	51
周三	1月20日	2	周六	2月20日	1	周日	3月20日	50
周四	1月21日	3	周日	2月21日	1	周一	3月21日	68
周五	1月22日	3	周一	2月22日	1	周二	3月22日	57
周六	1月23日	2	周二	2月23日	1	周三	3月23日	54
周日	1月24日	2	周三	2月24日	1	周四	3月24日	63
周一	1月25日	3	周四	2月25日	1	周五	3月25日	75
周二	1月26日	2	周五	2月26日	1	周六	3月26日	51
周三	1月27日	2	周六	2月27日	1	周日	3月27日	50
周四	1月28日	3	周日	2月28日	1	周一	3月28日	71
周五	1月29日	3	周一	2月29日	1	周二	3月29日	57
周六	1月30日	2				周三	3月30日	54
周日	1月31日	2				周四	3月31日	63

## 五、模型的评价

### 针对问题一的方案与模型的优点：

1：创造性地把问题的要求的两个约束（分流人群和保障游客的游园体验）转换为一个新的概念——分析每个项目点的游客数量，并给出合理的游园建议。

2：了解问题的实质，将问题的解决具体分为四个步骤，条理清晰，思路明确，层层递进，这种讨论问题，解决问题的方法简洁易懂，具有很高的可读性

3：把游客的等待时间细化，更贴合实际情况，切身考虑到游客的游园体验，建立游客等待时间的函数的变化模型，在理论上有一定的基础，在实际操作上有可行性，把抽象的事物具体化，数字化，使我们的讨论求解过程更加简便。

4：从游乐园管理角度出发，为保障游客安全，方便管理，设置每个游乐项目的排队等候人数的预警容量，并在考虑不超过该容量的情况下，选择出推荐路线，及时分流了人群，避免人群过于集中，造成不安全事件

5：针对模型，使用了 Dijkstra 算法，求出单源最短路径，模型建立后，编写 c++ 程序，使得采集数据后能够实现自动计算并给出推荐，已应用于上述案例分析，具有科学性，很贴近现实。

### 针对问题一的方案与模型的缺点：

因为随着游客接受路线推荐，在其移动到下一个项目点的过程中，各个项目点的游客排队数据都发生了变化，故各个项目点最优路线的推荐可能也发生了变化，但短时间内最优推荐发生变化的可能性较小，此外，我们默认人的步行速度是定值 $v$ ，与实际情况有偏差，但以上误差在可以接受的范围之内。

### 针对问题二的方案与模型的优点：

1. 采用了指数平滑预测法，根据 2015 年 1-12 月的每月预订数据对 2016 年 1-3 的每月预订数据进行了分析，避免了直接预测每日预订房间数所造成的误差。先确定大范围，再细分到小范围，使模型先预估出大致数据，在对每一天的细节进行修正，所得结果即符合大的趋势，又贴合每日的具体情况。

2. 使用模糊综合评价方法,对节假日、周末和寒暑假等对预定人数的影响做了较为合理的分析,用相应的影响因子修正了非规则因素对预订人房间数的影响,与实际相符。

3. 利用 MATLAB 和 C++进行编程和绘图分析,科学地分析数据并求得结果,使得最终结果误差较小处于合理的范围内。

4. 方法中采用图像与数学计算相结合的形式,从图像中选取最贴合的系数,再用确定的代数表达式推算出结果,过程中采用 Excel 表格对数据进行归纳并做图分析趋势,还用 MATLAB 由散点图得出函数表达式。这种图像、代数、统计相结合的形式使整个求解过程更加多样化、直观、易懂。

5. 该算法背景面向实际情况、针对性强,对生活中酒店的预订预测数量有一定的指导意义,可以在酒店预测方面作直接推广。

**针对问题二的方案与模型的缺点:**

未考虑到月初月末对人们预订因素的影响,近似的认为月内四周的人数是近乎平均分配的,故所得曲线在每个月内有一定的周期性,最终结果只能体现每周每天的大致情况而精确度不高。

## 六、模型的改进和推广

**问题一的模型的改进与推广:**总体来说,我们对问题进行了深入的分析,了解到问题的实质是降低游客等待时间,且分散人群,我们的模型比较贴合实际情况,具有较大的可读性和可实施性。考虑到模型的推广与使用,我们认为可以将该模型完善成一个系统,可与游乐场的检票系统相连接获得排队人数等所需数据,从游乐项目的控制中心获取相应时间数据,同时,与电子显示屏连接将所得结果显示出来,即可在游乐园中使用,能够根据客流情况,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。

**问题二的模型的改进与推广:**该模型的背景即为酒店预订房间数预测,通过此预测可以达到预估酒店预订房间数,今儿预估出一段时间内营业额的目的,并依次为基础作出相关人事变动、翻修装潢、改革升级等酒店内部调整。酒店可利用该模型设计一个预订房间数预测系统,通过以往年份的预订数据,对近期内的酒店预订数量作出合理的估计,这是酒店了解自身产业构架、寻找自身缺陷并及时改进的有力数据。并且该系统预测并不需要多年的大量数据,一年的预订情况同样可以测算出结果,这将是许多新开业酒店的首选。

**参考文献:**

- [1] 于敬岩. 酒店管理决策系统中客流量预测建模与分析[D]. 大连理工大学, 2008. DOI:10.7666/d.y1248062.
- [2] 韩路跃, 杜行检. 基于 MATLAB 的时间序列建模与预测[J]. 计算机仿真, 2005, 04:105-107+182.
- [3] 代西武. Dijkstra 矩阵算法[J]. 北京建筑工程学院学报, 2007, No. 6902:65-67+71.
- [4] 陈晓卫, 李洪. 酒店入住人数模型研究[J]. 统计教育, 2006, 03:13-14.
- [5] 姜启源, 数学模型(第3版), 北京: 高等教育出版社, 1999
- [6] 周炳飞. 基于 matlab 软件的动态指数平滑模型建立与应用[J]. 企业导报, 2014, No. 27317:142-143.

附录:

附录一: MATLAB 求单源最短路径的程序 (Dijkstra 算法)

```
function [min,path]=dijkstra(w, start, terminal)
n=size(w,1); label(start)=0; f(start)=start;
for i=1:n
    if i~=start
        label(i)=inf;
    end, end
s(1)=start; u=start;
while length(s)<n
    for i=1:n
        ins=0;
        for j=1:length(s)
            if i==s(j)
                ins=1;
            end, end
        if ins==0
            v=i;
            if label(v)>(label(u)+w(u,v))
                label(v)=(label(u)+w(u,v)); f(v)=u;
            end, end, end
    v1=0;
    k=inf;
    for i=1:n
        ins=0;
        for j=1:length(s)
            if i==s(j)
                ins=1;
            end, end
        if ins==0
            v=i;
            if k>label(v)
                k=label(v); v1=v;
            end, end, end
        s(length(s)+1)=v1;
        u=v1;
    end
min=label(terminal); path(1)=terminal;
i=1;
while path(i)~=start
    path(i+1)=f(path(i));
    i=i+1 ;
end
path(i)=start;
```



```
L=length(path);  
path=path(L:-1:1);
```

## 附录二：c++程序

```
#include "iostream.h"  
#define MAX 9999  
int min(double *aa)  
{  
    int k;  
    double m=999.0;  
    for(int i=1;i<11;i++)  
    {  
        if(aa[i]<m && aa[i]!=0)  
        {m=aa[i];  
        k=i;}  
    }  
    return k;  
}  
void main()  
{  
    int t1[11]={0,9,1,2,1,2,1,1,1,1,1};  
    double ti[11]={0,33,1.25,2.5,2.5,5,2.5,2,1.5,1.5,2};  
    int p[11]={0,300,60,40,50,90,80,60,40,250,100};  
    int s[11]={0,400,30,50,30,100,50,30,30,20,50};  
    int Maxi[11]={0,700,250,300,250,450,300,250,250,200,300};  
    double jj[11];  
    jj[0]=0;  
    int a[11];  
    double t[11][11]=  
    {  
        0,0,0,0,0,0,0,0,0,0,0,  
  
        0,0,4.498,8.996,15.742,5.247,23.238,13.493,15.742,8.996,3.748,  
  
        0,4.498,0,4.498,11.244,5.247,18.741,13.493,19.490,11.994,8.246,  
  
        0,8.996,4.498,0,6.747,7.496,14.243,15.742,20.990,14.243,12.744,  
  
        0,15.742,11.244,6.747,0,14.243,7.496,17.241,23.238,20.990,19.49  
0,  
  
        0,5.247,5.247,7.496,14.243,0,17.991,8.246,13.493,6.747,8.996,
```

```

0, 23. 238, 18. 741, 14. 243, 7. 496, 17. 991, 0, 9. 745, 15. 742, 22. 489, 27. 73
6,

0, 13. 493, 13. 493, 15. 742, 17. 241, 8. 246, 9. 745, 0, 5. 997, 2. 744, 17. 991,

0, 15. 742, 19. 490, 20. 990, 23. 238, 13. 493, 15. 742, 5. 997, 0, 6. 746, 11. 99
4,

0, 8. 996, 11. 994, 14. 243, 20. 990, 6. 747, 22. 489, 12. 744, 6. 746, 0, 5. 247,

0, 3. 748, 8. 246, 12. 744, 19. 490, 8. 996, 27. 736, 17. 991, 11. 994, 5. 247, 0
};
int judge[11]={0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1};
for(int i=1;i<11;i++)
{
    if(p[i]>Maxi[i])
        judge[i]=MAX;
}
for(i=1;i<11;i++)
{
    for(int j=1;j<11;j++)

        jj[j]=double(double(judge[j])*double(t1[j]+int(p[j]/s[j])*ti[j]
+t[i][j]));
        for(j=1;j<11;j++)
        {
            if(jj[j]>9999)
                cout<<"A"<<j<<"人数达到预警! ";
            else
                cout<<"A"<<i<<"→A"<<j<<"<<":<<jj[j]<<"min"<<"    ";
            if(j==5)
                cout<<endl;
        }
        a[i]=min(jj);
        cout<<endl;
        cout<<"此时 A"<<i<<"地的游客"<<"建议去往"<<"A"<<a[i]<<"地
"<<endl;
    }
}
}

```

### 附录三：一次指数平滑法 MATLAB 程序

```
function [ output_args ] = once( input_args )
a=0.8;
y=[46;12;320;3065;2576;2561;2831;3033;3114;2987;2724;2657];
st1(1)=(y(1)+y(2)+y(3)+y(4)+y(5)+y(6))/6;
e(1)=st1(1)-y(1);
for i=2:12
    st1(i)=a*y(i)+(1-a)*st1(i-1);
    e(i)=st1(i)-y(i);
end
xlswrite('一次指数平滑法 a=0.8 .xls',[st1',e']);
t=1:12;
plot(t,y,t,st1);
end
```

### 附录四：二次指数平滑法 MATLAB 程序

```
function [ output_args ] = twice( input_args )
a=0.8;
y=[46;12;320;3065;2576;2561;2831;3033;3114;2987;2724;2657];
st1(1)=(y(1)+y(2)+y(3)+y(4)+y(5)+y(6))/6;
st2(1)=(y(1)+y(2)+y(3)+y(4)+y(5)+y(6))/6;
at(1)=2*st1(1)-st2(1);
bt(1)=0;
yt(1)=0;
e(1)=0;
for i=2:12
    st1(i)=a*y(i)+(1-a)*st1(i-1);
    st2(i)=a*st1(i)+(1-a)*st2(i-1);
    at(i)=2*st1(i)-st2(i);
    bt(i)=(a/(1-a))*(st1(i)-st2(i));
    yt(i)=at(i-1)+bt(i-1);
    e(i)=yt(i)-y(i);
end
yt(13)=at(12)+bt(12);
yt(14)=at(12)+bt(12)*2;
yt(15)=at(12)+bt(12)*3;
xlswrite('yt.xls',[yt']);
xlswrite('st2.xls',[st2']);
xlswrite('e.xls',[e']);
t=1:12;
plot(t,y,t,st2);
```

```
t=1:15;
plot(t, yt);
end
```

附录五：

(2015 年) 五一小长假非规则因素表			
预定提前 天数 PD	2015 预定房间数 R	非规则因素	对应影响 2016 年的日期
0	8	1.091954023	
1	2	1.022988506	
2	1	1.011494253	
3	1	1.011494253	
4	7	1.08045977	
5	1	1.011494253	
6	43	1.494252874	
7	0	1	
8	1	1.011494253	
9	19	1.218390805	
10	9	1.103448276	
11	19	1.218390805	
12	33	1.379310345	
13	3	1.034482759	
14	37	1.425287356	
15	10	1.114942529	
16	3	1.034482759	
17	2	1.022988506	
18	19	1.218390805	
19	28	1.32183908	
20	0	1	
21	1	1.011494253	
22	41	1.471264368	
23	1	1.011494253	
24	0	1	
25	0	1	
26	1	1.011494253	
27	0	1	
28	2	1.022988506	
29	0	1	

30	0	1	
31	0	1	2016/3/31
32	0	1	2016/3/30
33	0	1	2016/3/29
34	0	1	2016/3/28
35	1	1. 011494253	2016/3/27
36	0	1	2016/3/26
37	0	1	2016/3/25
38	0	1	2016/3/24
39	0	1	2016/3/23
40	0	1	2016/3/22
41	0	1	2016/3/21
42	0	1	2016/3/20
43	0	1	2016/3/19
44	0	1	2016/3/18
45	0	1	2016/3/17
46	0	1	2016/3/16
47	0	1	2016/3/15
48	0	1	2016/3/14
49	0	1	2016/3/13
50	0	1	2016/3/12
51	0	1	2016/3/11
52	0	1	2016/3/10
53	0	1	2016/3/9
54	0	1	2016/3/8
55	0	1	2016/3/7
56	0	1	2016/3/6
57	0	1	2016/3/5
58	0	1	2016/3/4
59	0	1	2016/3/3
60	1	1. 011494253	2016/3/2
61	1	1. 011494253	2016/3/1
62	0	1	2016/2/29
63	0	1	2016/2/28
64	0	1	2016/2/27
65	0	1	2016/2/26
66	0	1	2016/2/25
67	0	1	2016/2/24
68	0	1	2016/2/23

69	0	1	2016/2/22
70	0	1	2016/2/21
71	0	1	2016/2/20
72	0	1	2016/2/19
73	0	1	2016/2/18
74	1	1. 011494253	2016/2/17
75	0	1	2016/2/16
76	0	1	2016/2/15
77	0	1	2016/2/14
78	0	1	2016/2/13
79	0	1	2016/2/12
80	6	1. 068965517	2016/2/11
81	0	1	2016/2/10
82	0	1	2016/2/9
83	0	1	2016/2/8
84	0	1	2016/2/7
85	0	1	2016/2/6
86	0	1	2016/2/5
87	1	1. 011494253	2016/2/4

附录六：

(2015 年) 清明非规则因素表			
预定提前天数 PD	2015 房间预订数 R	非规则因素	对应 2016 年的影响日期
0	53	1. 203065134	
1	153	1. 586206897	
2	12	1. 045977011	
3	4	1. 01532567	
4	6	1. 022988506	2016/3/31
5	4	1. 01532567	2016/3/30
6	3	1. 011494253	2016/3/29
7	14	1. 053639847	2016/3/28
8	2	1. 007662835	2016/3/27
9	0	1	2016/3/26
10	2	1. 007662835	2016/3/25
11	1	1. 003831418	2016/3/24
12	0	1	2016/3/23
13	0	1	2016/3/22
14	1	1. 003831418	2016/3/21

15	2	1. 007662835	2016/3/20
16	1	1. 003831418	2016/3/19
17	0	1	2016/3/18
18	0	1	2016/3/17
19	0	1	2016/3/16
20	0	1	2016/3/15
21	0	1	2016/3/14
22	0	1	2016/3/13
23	0	1	2016/3/12
24	0	1	2016/3/11
25	1	1. 003831418	2016/3/10
26	0	1	2016/3/9
27	0	1	2016/3/8
28	1	1. 003831418	2016/3/7
29	0	1	2016/3/6
30	0	1	2016/3/5
31	0	1	2016/3/4
32	0	1	2016/3/3
33	0	1	2016/3/2
34	1	1. 003831418	2016/3/1