ATM 交易状态特征分析与异常检测

完成者：储著敏 顾思远 赵东阳

指导教师：梁恒

清华大学

2017 年 8 月

目　录

摘　要

随着 ATM 机使用的日益普及，ATM 机的管理成为商业银行管理的重要组成部分，ATM 机现金流管理、ATM 操作行为监控、ATM 交易状态异常检测等是 ATM 管理的重要课题。目前绝大多数商业银行对 ATM 管理的研究均集中在 ATM 机现金流管理和 ATM 操作行为监控方面。利用交易信息对 ATM 交易状态进行异常检测，通过对分行的汇总统计信息做数据分析，捕捉整个 ATM 系统前端和后端整体应用系统运行情况以及时发现异常或故障，快速做出响应，对提高系统效率，减少停机检修成本，提高用户满意度具有重要意义。

本文采用分而治之的方法对商业银行总行数据中心监控系统的 ATM 交易信息进行分析与建模，利用已知的业务量、交易成功率、交易响应时间三维时间序列数据，将 ATM 交易系统分为用户行为（业务量时间序列）和系统特性（交易响应时间、交易成功率）两部分分别建立模型进行分析，利用分析得到的结论进行 ATM 交易系统异常检测。

对于任务 1，在对已知数据进行分析的基础上，对于 ATM 交易系统本身，利用计算机网络工程中常用的排队模型对业务量和交易响应时间的关系进行了分析，建立了 ATM 交易系统多线程服务时间不等的 M/M/k 排队模型，解释了交易响应时间随业务量增大反而降低的原因，确定了交易响应时间异常的类型。引入新变量交易总响应时间，通过对总响应时间与业务量关系的分析，发现总响应时间与业务量只存在着两种确定的函数关系，通过回归确定该函数和该函数在给定置信水平下的预测区间。

引入新变量交易成功量，分析发现交易成功量与业务量成线性关系，根据得到的回归模型确定平均交易成功率和该模型在给定置信水平下的预测区间。

对于任务 2 中的业务量异常检测部分，首先对业务量随时间变化的 Pattern 进行预处理，利用回归提取出的参数对该业务量时间序列进行聚类分析。根据回归模型和数据范围，引入评分机制对业务量进行打分，利用该评分机制在给定置信水平的阈值下进行了业务量异常值提取。

类比信息检索领域的思路，提出了基于向量空间和可靠性的贡献值分配模型，并以此进行业务量异常检测。在这个模型的基础上，将业务量拆分成日业务总量和业务量占比两个随机变量，将统计模型与机器学习相结合，提出了二维贡献值分配模型。

基于任务 1 中得到的的响应时间异常类型，给出了区分不同日期和不同时刻的分层交易响应时间异常检测方法。利用交易成功量与业务量的线性关系，给出交易成功量的异常检测方法。提出了减少虚警误报的措施。

对于任务 3，针对用户行为、系统特征和实际管理过程分别提出了基于扩展数据的方案优化思路。

关键词：ATM 交易系统；聚类分析；回归分析；排队过程；贡献值分配模型；异常检测

1 问题重述

某商业银行的 ATM 应用系统包括前端和后端两个部分。前端是部署在商业银行营业部和各自助服务点的 ATM 机（系统），后端是总行数据中心的处理系统。前端的主要功能是和客户直接交互，采集客户请求信息，然后通过网络传输到后端，再进行数据和账务处理。持卡人从前端设备提交查询或转账或取现等业务请求，到后台处理完毕，并将处理结果返回到前端，通知持卡人业务处理最终状态，我们称这样完整的一个流程为一笔交易。

商业银行总行数据中心监控系统为了实时掌握全行的业务状态，每分钟对各分行的交易信息进行汇总统计。汇总信息包括业务量、交易成功率、交易响应时间三个指标，各指标解释如下：

业务量：每分钟总共发生的交易总笔数；

交易成功率：每分钟交易成功笔数和业务量的比率；

交易响应时间：一分钟内每笔交易在后端处理的平均耗时(单位：毫秒)。

交易数据分布存在以下特征：工作日和非工作日的交易量存在差别；一天内，交易量也存在业务低谷时间段和正常业务时间段。当无交易发生时，交易成功率和交易响应时间指标为空。

商业银行总行数据中心监控系统通过对每家分行的汇总统计信息做数据分析，来捕捉整个前端和后端整体应用系统运行情况以及时发现异常或故障。常见的故障场景包括但不限于如下情形：

分行侧网络传输节点故障，前端交易无法上送请求，导致业务量陡降；

分行侧参数数据变更或者配置错误，数据中心后端处理失败率增加，影响交易成功率指标；

数据中心后端处理系统异常（如操作系统 CPU 负荷过大）引起交易处理缓慢，影响交易响应时间指标；

数据中心后端处理系统应用进程异常，导致交易失败或响应缓慢。

附件是某商业银行 ATM 应用系统某分行的交易统计数据。

任务：

（1）选择、提取和分析 ATM 交易状态的特征参数；

（2）设计一套交易状态异常检测方案，在对该交易系统的应用可用性异常情况下能做到及时报警，同时尽量减少虚警误报；

（3）设想可增加采集的数据。基于扩展数据，你能如何提升任务（1）（2）中你达到的目标？

2 问题分析

2.1 文献检索评述

本题是有关 ATM 交易状态特征分析与异常检测问题，涉及到数据挖掘和故障预警两个方面。

随着信息技术发展的日新月异，以机器取代人力实现办公自动化，降低办公的时间、空间成本，提高办公效率，成为各行各业的共识，也是现代企业在激烈的市场竞争中生存的必然要求。自助取款机（ATM）正是在这一背景下诞生的产物，它的出现极大降低了商业银行业的运行成本，突破了工作时间和空间的限制，极大满足了用户的需求，得到了突飞猛进的发展。自中国商业银行深圳分行于 1988 年推出国内第一台联网 ATM 机以来，联网的 ATM 机数量不断增加，到 2015 年第三季度末，国内联网 ATM 机已达 84.08 万台，且呈现快速增加的趋势。[1]随着 ATM 机使用的日益普及，ATM 机的管理成为商业银行管理的重要组成部分，ATM 机现金流管理、ATM 操作行为监控、ATM 交易状态异常检测等是 ATM 管理的重要课题。目前绝大多数商业银行对 ATM 管理的研究均集中在 ATM 机现金流管理和 ATM 操作行为监控方面，利用回归分析、时间序列分析、人工智能方法进行 ATM 机现金流预测[2, 3]，以及利用机器视觉的方法进行 ATM 机操作异常行为

的智能检测[4-11]近年来得到飞速发展。

利用交易信息对 ATM 交易状态进行异常检测，通过对分行的汇总统计信息做数据分析，捕捉整个 ATM 系统前端和后端整体应用系统运行情况以及时发现异常或故障，快速做出响应，对提高系统效率，减少停机检修成本，提高用户满意度具有重要意义。

2.2 问题的分析与研究思路

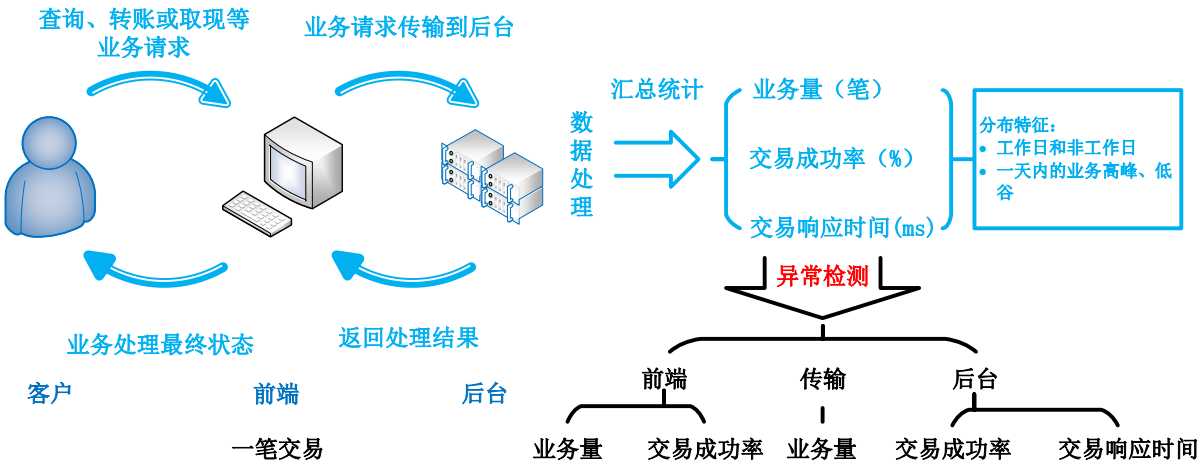


图 1 ATM 交易状态特征分析与异常检测

ATM 交易系统包括前端和后端两部分，数据中心监控系统的汇总的交易信息包括业务量、交易成功率和交易响应时间三个指标。已知的数据为如下的五维数组：

$$(\text{日期}\boldsymbol{d},\text{时间}\boldsymbol{t},\text{业务量}\boldsymbol{m},\text{交易响应时间}\boldsymbol{T},\text{交易成功率}\boldsymbol{n})$$

本文采用分而治之的方法对商业银行总行数据中心监控系统的 ATM 交易信息进行分析与建模，利用已知的业务量、交易成功率、交易响应时间三维时间序列数据，将 ATM 交易系统分为用户行为（业务量时间序列）和系统特性（交易响应时间、交易成功率）两部分分别建立模型进行分析，利用分析得到的结论进行 ATM 交易系统异常检测。

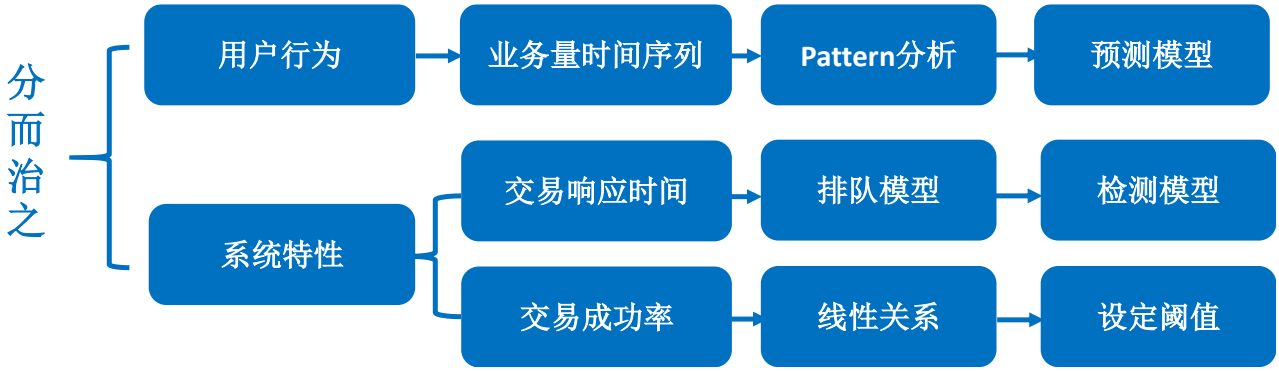


图 2 研究思路

3 任务 1：ATM 交易状态特征参数的选择、提取与分析

首先对数据进行分析。

3.1 业务量-交易响应时间-交易成功率的关系

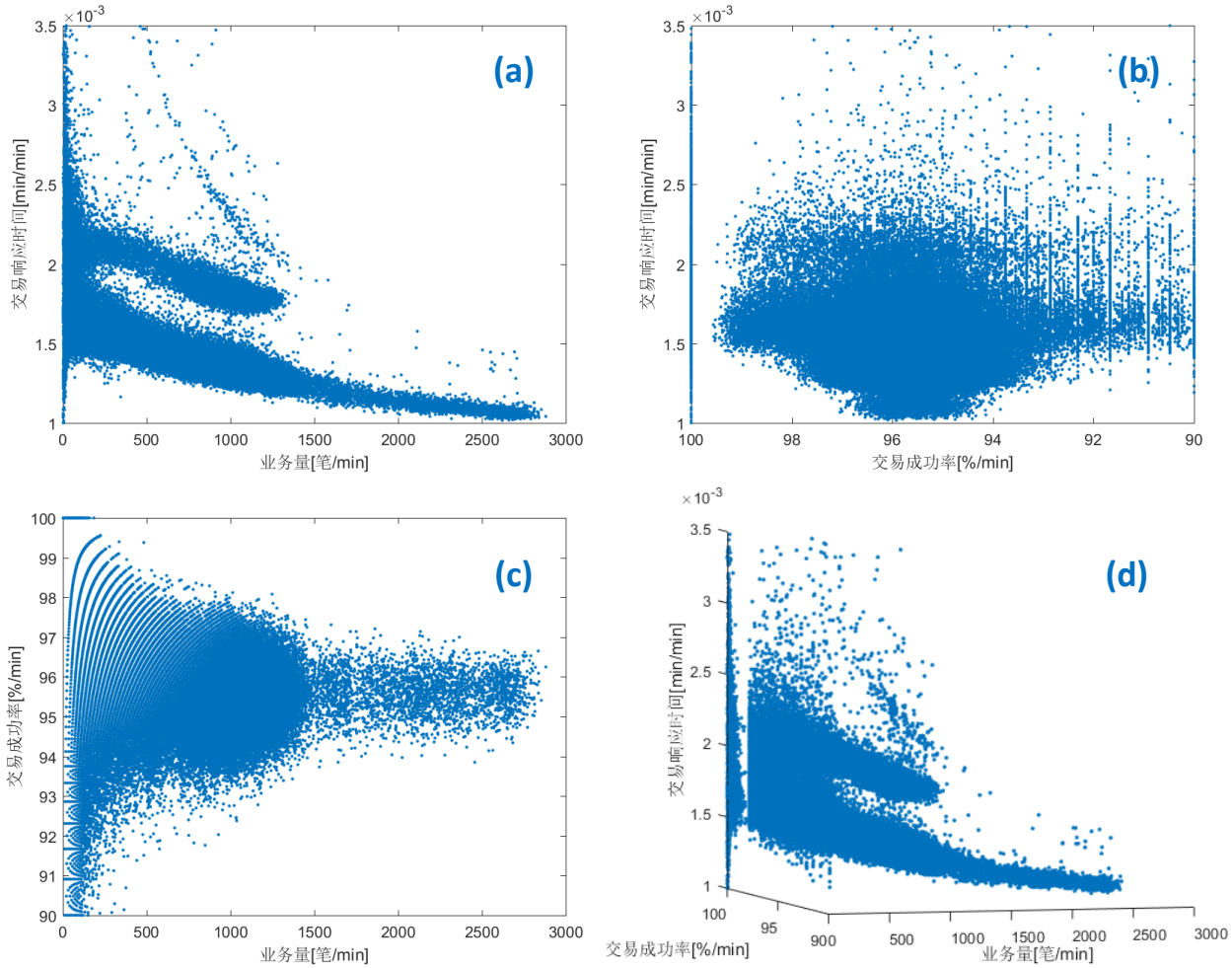


图 3 业务量-交易响应时间-交易成功率的关系

根据已知的业务量、交易成功率和交易响应时间信息可得到三者的三维分布图，如图 3 所示，图 3(a)-(c)是图 3(d)在三个方向上的投影，分别代表了“交易响应时间-业务量”、“交易响应时间-交易成功率”、“交易成功率-业务量”的关系。从图 3(d)可以看出，业务量、交易成功率和交易响应时间三者是相互关联的。

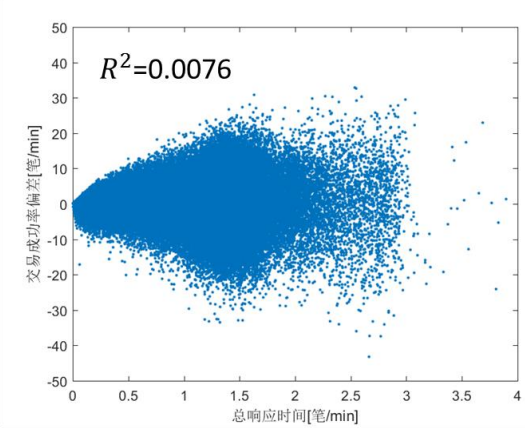


图 4 交易成功率偏差和交易响应时间的关系

经过后面的分析，可得到交易成功率偏差和总响应时间的关系，如图 4 所示，即为残差图，交易成功率偏差与总响应时间的相关系数很低，可知二者基本相互独立。

3.2 交易成功量与业务量的关系

图 3(b)为交易成功率和业务量的关系，有如下特点：

（1）交易成功率在 95%-96%上下的范围内分布，这可从图 5 更明显的看出。

（2）交易成功率存在明显的分立曲线关系。

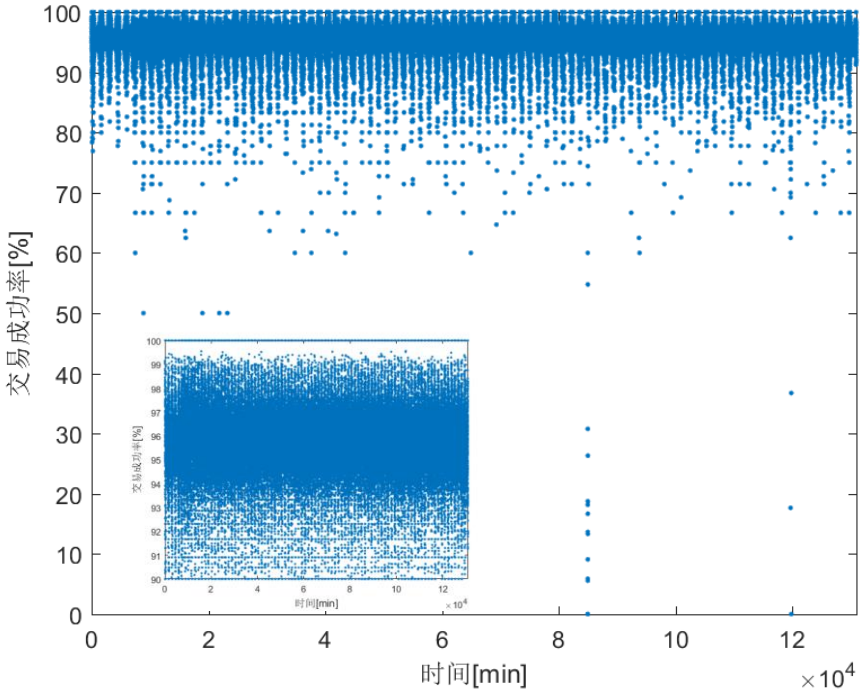


图 5 交易成功率随时间的变化

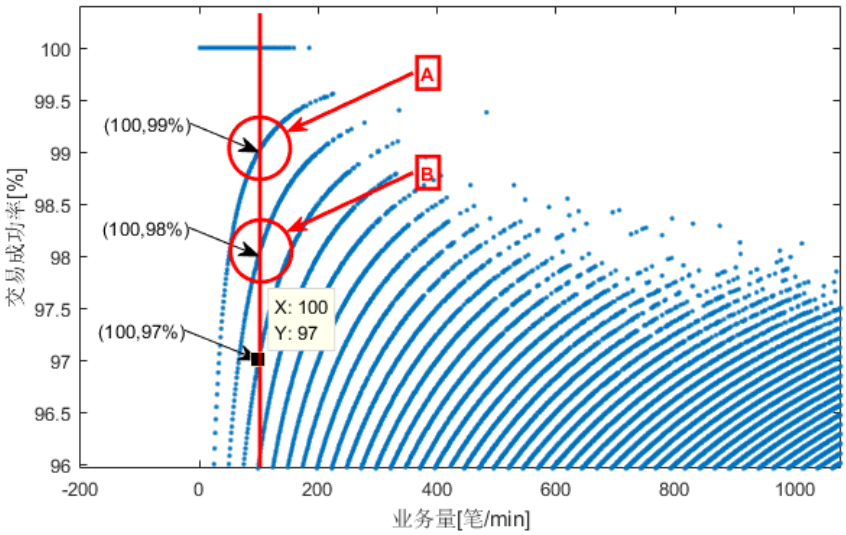


图 6 交易成功率随业务量变化的局部放大图

关于交易成功率呈现分立的曲线关系的解释如下。如图 6，在业务量为 100 时，各分立曲线与业务量

为 100 的竖直线的交点，交易成功量即交易成功率×交易量，由上到下分别为 100、99、98、97……，说明分立曲线分布是由交易成功量必须为整数造成的。取图中 A 和 B 两个范围内的点，计算交易成功量，如表 1 和图 7 所示。可知分立曲线上交易成功量与业务量呈线性关系，即满足

$$\boldsymbol{n \times m = m + k}$$

$$\boldsymbol{n = k/m + 1}$$

其中，n 为交易成功率；m 为业务量；k 为整数。可知分立曲线确实为双曲线，分立的原因是系数 k 只能取整数。

因此，接下来将交易成功率转化为交易成功量分析。

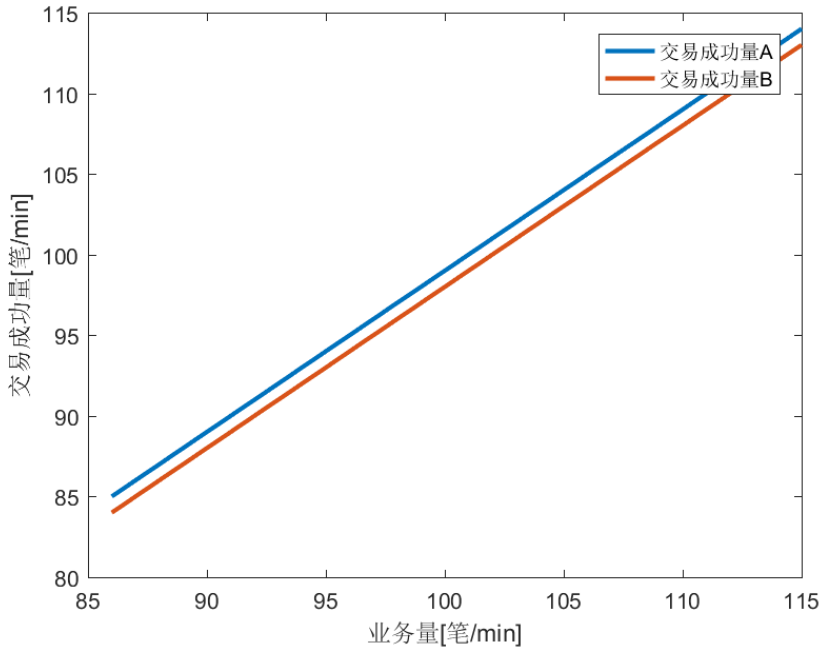


图 7 图 6 中 A、B 局部点交易成功量随业务量的变化

表 1 图 8 中的局部点数据

业务量	交易成功率 A	交易成功率 B	交易成功量 A	交易成功量 B
115	99.13%	98.26%	113.9995	112.999
114	99.12%	98.25%	112.9968	112.005
113	99.12%	98.23%	112.0056	110.9999
112	99.11%	98.21%	111.0032	109.9952
111	99.10%	98.20%	110.001	109.002
110	99.09%	98.18%	108.999	107.998
109	99.08%	98.17%	107.9972	107.0053
108	99.07%	98.15%	106.9956	106.002
107	99.07%	98.13%	106.0049	104.9991
106	99.06%	98.11%	105.0036	103.9966
105	99.05%	98.10%	104.0025	103.005
104	99.04%	98.08%	103.0016	102.0032
103	99.03%	98.06%	102.0009	101.0018
102	99.02%	98.04%	101.0004	100.0008
101	99.01%	98.02%	100.0001	99.0002

100	99.00%	98.00%	99	98
99	98.99%	97.98%	98.0001	97.0002
98	98.99%	97.96%	97.0102	96.0008
97	98.97%	97.94%	96.0009	95.0018
96	98.96%	97.92%	95.0016	94.0032
95	98.95%	97.89%	94.0025	92.9955
94	98.94%	97.87%	93.0036	91.9978
93	98.92%	97.85%	91.9956	91.0005
92	98.91%	97.83%	90.9972	90.0036
91	98.90%	97.80%	89.999	88.998
90	98.89%	97.78%	89.001	88.002
89	98.88%	97.75%	88.0032	86.9975
88	98.86%	97.73%	86.9968	86.0024
87	98.85%	97.70%	85.9995	84.999
86	98.84%	97.67%	85.0024	83.9962

交易成功量与业务量的关系如图 8 所示。可知交易成功量与业务量的关系为线性关系，利用最小二乘法对二者的关系进行线性拟合，拟合式为：

$$m_s = \overline{n}m + b_s$$

拟合参数与其 95%置信区间：

$$\overline{n} = 0.9561, (0.9561, 0.9562)$$

$$b_s = 0.3806, (0.333, 0.4283)$$

曲线斜率为平均交易响应时间 95.61%。拟合曲线和置信水平$\mathbf{1-\varphi}$为 99.99999999999%的预测区间如图 8 所示。

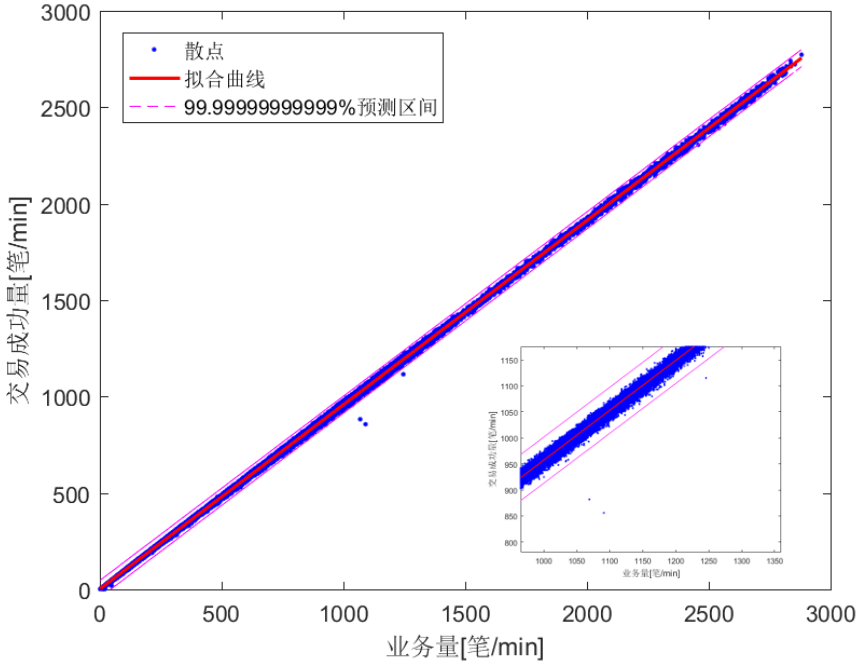


图 8 交易成功量和业务量关系的拟合结果

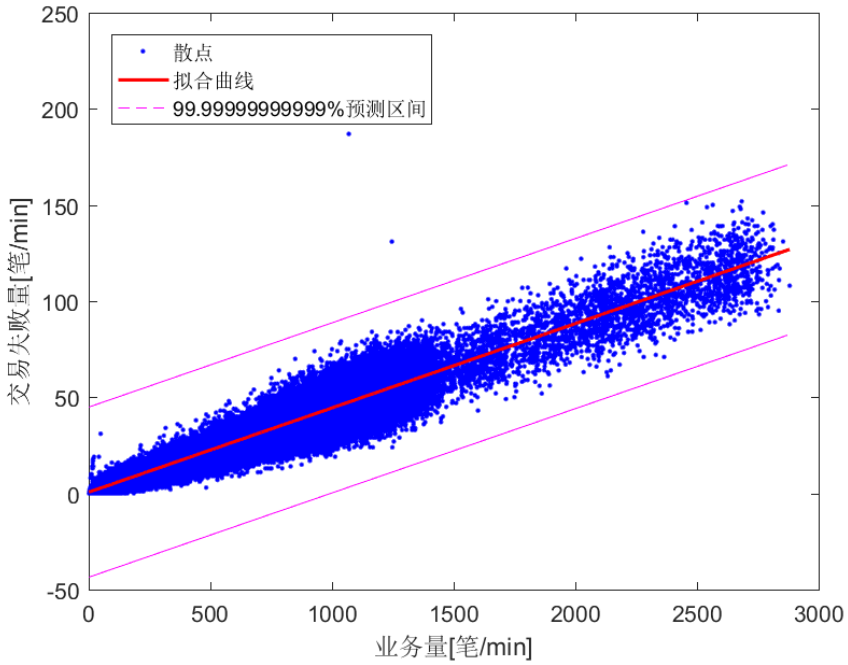


图 9 交易失败量和业务量关系的拟合结果

同样可做出交易失败量和业务量的关系及拟合曲线，如图 9 所示。从图中可以更清楚地看出交易失败量与业务量之间的线性关系，且由于春节之前业务量较平时大，业务量在 1500-3000 笔/min 段散点较稀疏。

由图 8 和图 9 可知为将明显离群的点排除，预测区间的置信水平$\mathbf{1-\varphi}$需取为 99.99999999999%，为了更合理地判断交易成功率的异常点，应事先给定交易成功量预测区间的置信水平$\mathbf{1-\varphi}$，据此确定交易成功量预测区间下界$\underline{\boldsymbol{\theta}_{m_s}}\big|_{1-\varphi}$，当交易成功量低于$\underline{\boldsymbol{\theta}_{m_s}}\big|_{1-\varphi}$时，为交易成功率异常。交易成功量预测区间的置信水平$\mathbf{1-\varphi}$可由用户体验、银行的工作效率要求等确定。

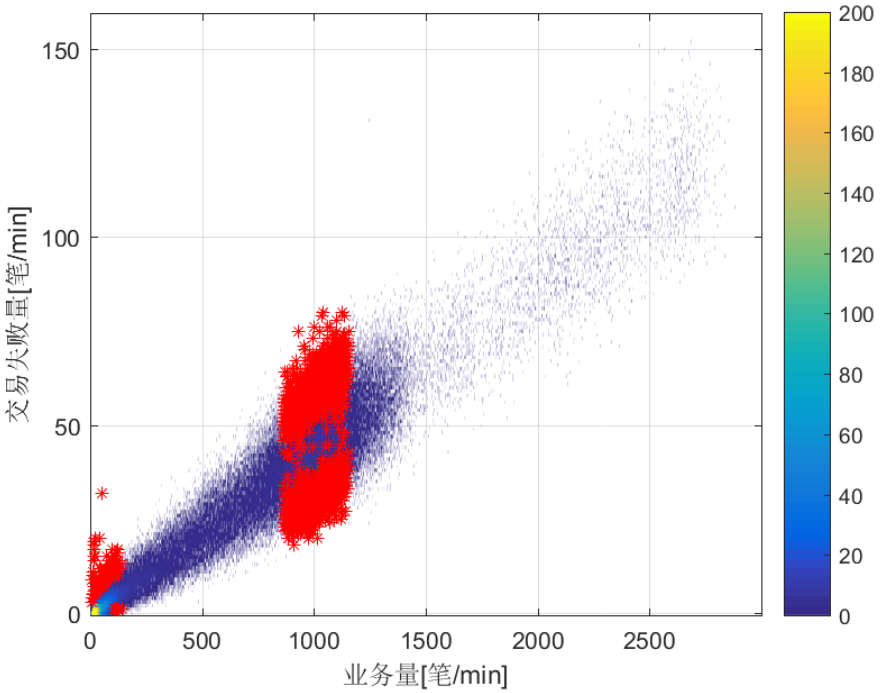


图 10 交易失败量与业务量之间关系的点数密度分布图（图中红色点代表点的概率密度小于 0.01）

交易失败量与业务量之间关系的点数密度分布图如图 10 所示。由图 10 可知，交易失败量在业务量为 1000 左右时的分布很分散，为非正态分布。由后文对于业务量时间序列的分析可知，此处大体处于一天的正午时段，业务量出现峰值，且不同日期的业务量峰值存在剧烈的波动，业务量峰值服从一定的概率分布。同时，每天的业务量时间序列随时间变化的 Pattern 也存在一定的概率分布，两种分布叠加的效果使得交易失败量和业务量的关系呈现特殊的概率分布形式。

3.3 交易总响应时间与业务量的关系——排队模型

由图 3(a)可知：

（1）交易量很低时，交易响应时间很高，服务器处理速度较低；

（2）交易响应时间随业务量的升高而降低，说明服务器处理速度逐渐加快，且交易响应时间主要由大小聚集的三部分组成；

（3）业务量主要集中在 500-1500 笔/min，超过 1500 笔/min 的业务量为春节前的业务量，此处的点较稀疏，且只有一簇。

做出 1 至 4 月份每天的交易响应时间随时间的变化，如图 11 所示。白天尤其是上班时间（9:00-18:00）的交易响应时间短且平稳，夜间尤其是 23:00-7:00 期间的交易响应时间长，且波动剧烈，数据分散。两段时间之间为过渡阶段。

1 月份的数据中，白天的交易响应时间值分为三部分，23-26 日为低值，27 日为过渡，28-31 日为高值。

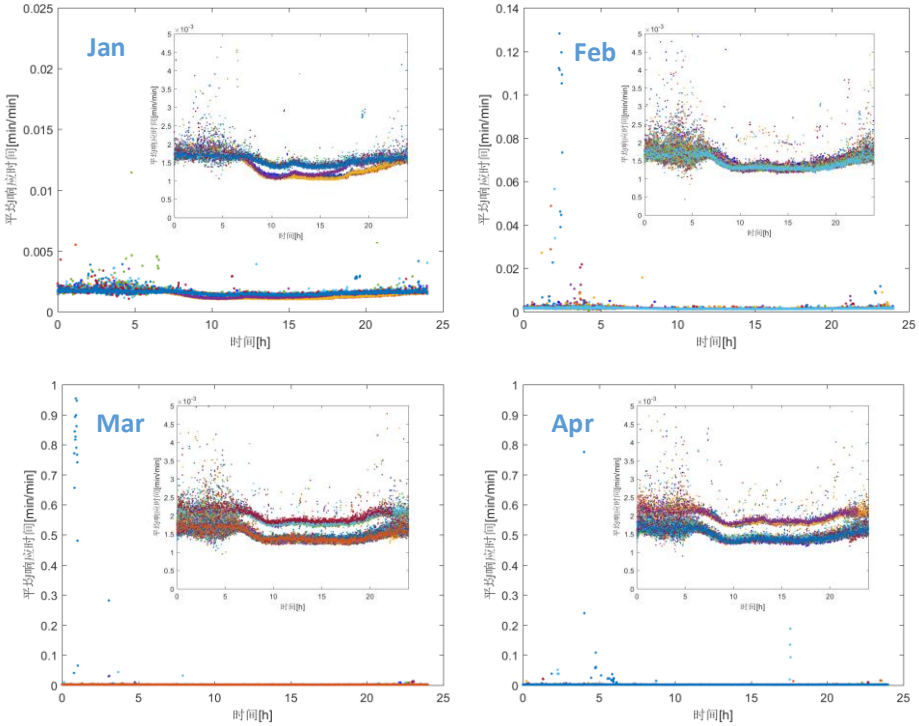


图 11 一至四月份每天的交易响应时间随时间的变化

2 月份的数据每天的交易响应时间的变化情况基本重叠。

3 月份和 4 月份每天的交易响应时间存在高低不同的两个值。

下面定义总响应时间=业务量×交易响应时间，即

$$T_{total} = mT$$

如图 12 所示，做出 1-4 月份总响应时间与业务量的关系。可知 1 月份和 2 月份总响应时间与业务量的关系较集中，3 月份和 4 月份的总响应时间与业务量的关系出现分叉。

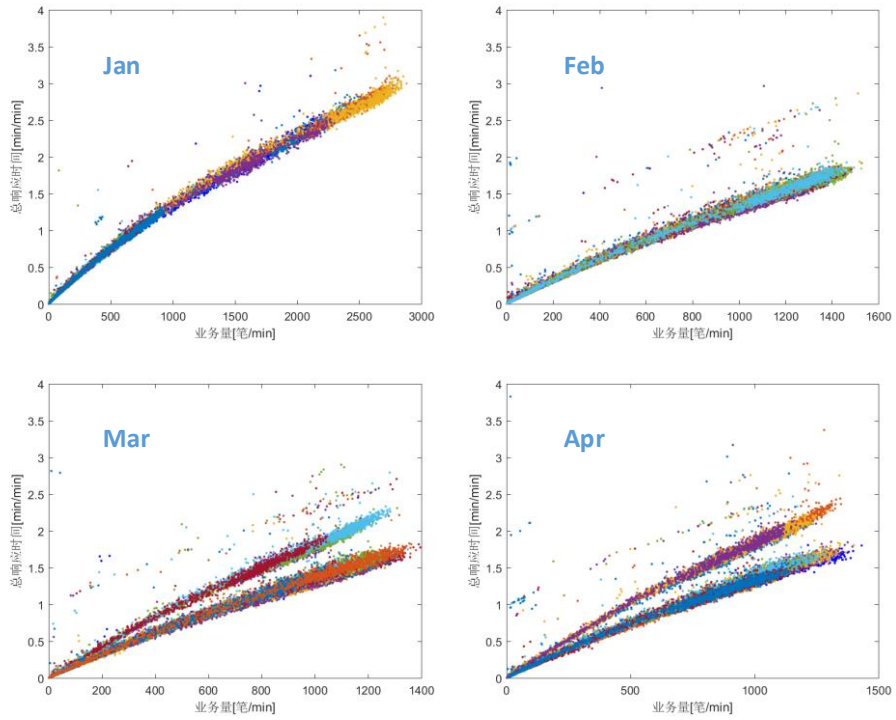


图 12 一至四月份总响应时间与业务量的关系

通过对 3 月份和 4 月份的数据进行进一步分析，如图 13，可知交易总响应时间的斜率（平均交易响应时间）有两个值，3 月 19-22 日和 4 月 16-19 日为高值，其他时间斜率为低值。由后文的分析（图 14）可知，交易响应时间的斜率只有这两个值。

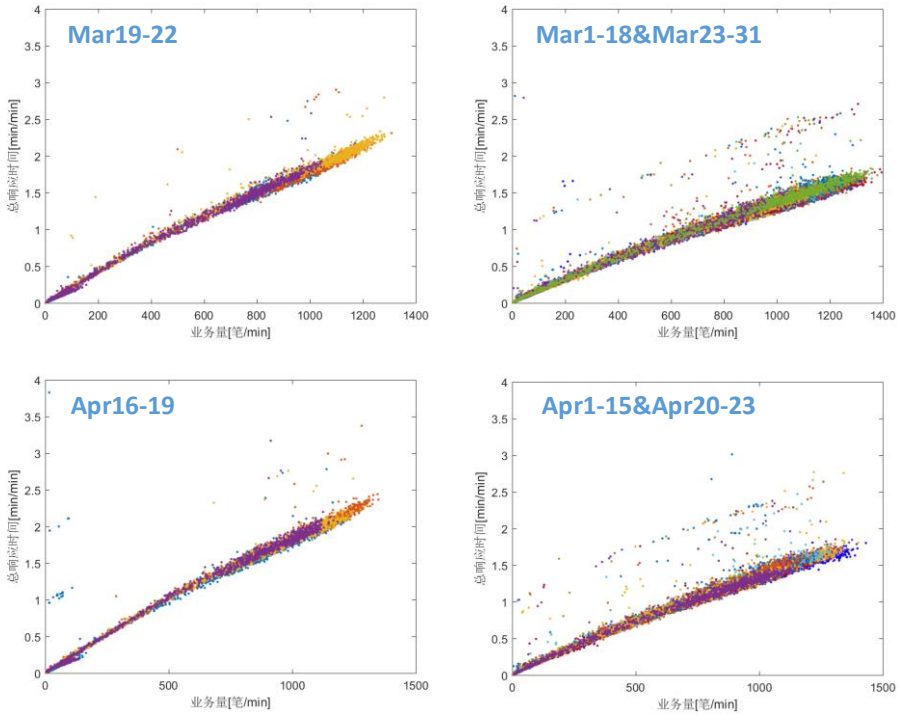


图 13 3 月份和 4 月份的两种总响应时间和业务量的关系

接下来重点分析业务量越高，交易响应时间反而会越短的特殊特征。这似乎是一个悖论，因为业务量升高，即使是并发处理，不同的交易之间因为存在冲突和等待，服务器监控到的交易响应时间会增大，这里却观察到平均交易响应时间随业务量的增大而降低。对此，我们提出三种假设：

假设 1：夜间和白天的处理方式不同，夜间为串行处理，白天为并行处理。由图 14 可知，这种特征不明显，ATM 系统在白天和夜间的处理能力是基本相同的。

假设 2：业务量越高，由于系统的并发调度算法，响应时间会缩短。由上述分析知，由于并发交易之间不可避免要出现冲突和等待，单个业务的响应时间会变长。

假设 3：业务量越高，响应时间短的业务占比越大，响应时间越短。由图 3(a)可知当业务量接近于零时，交易响应时间分布范围很大，说明不同类型的交易响应时间不同，这解释了夜间响应时间波动剧烈的现象。此外，随着业务量的增大，由于取平均效应，响应时间的范围会缩短。

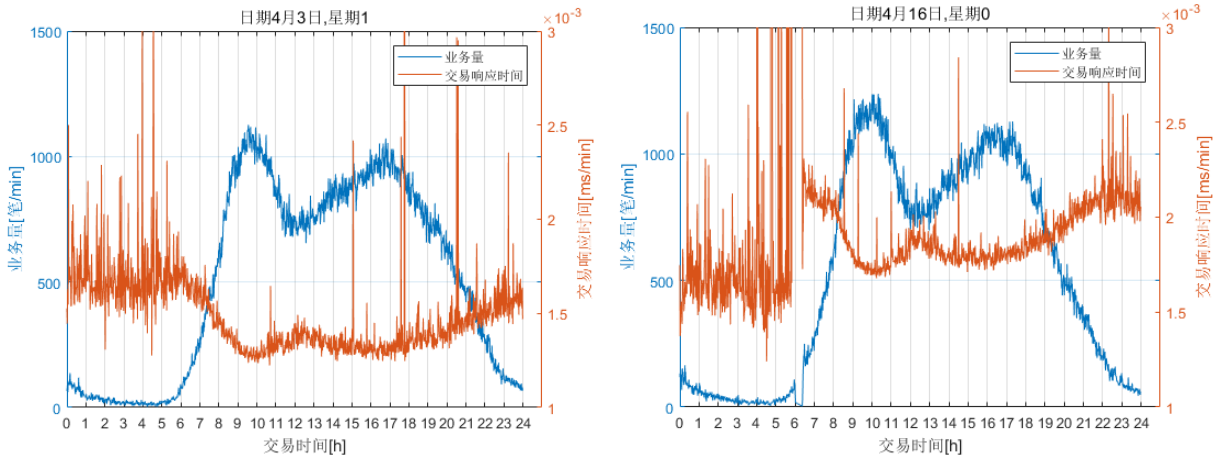


图 14 4 月 3 日和 4 月 16 日的业务量和交易响应时间的变化

为了进一步检验假设 3 的正确性，接下来我们建立后端服务器的排队模型。ATM 交易系统的业务量，其实等价于后端服务器的吞吐量，而吞吐量-响应时间是通信及网络工程中常见的变量，因此可利用通信及网络工程中常用的研究方法——排队论来进行研究。这里每分钟的业务量相当于队列的到达率，响应时间相当于队列的平均等待时间。

这里为了简化模型求解，构建了两窗口服务能力不同的 M/M/2 排队模型[12]，模型的状态转移图如图 15 所示，该模型假设交易到达后端的过程为准稳态的泊松过程，到达率即为业务量；为简单起见，只考虑两种不同类型的交易，交易响应速率不同，分别为μ_1和μ_2；到达后端的交易以确定的概率φ和$1-\varphi$分为两种类型。

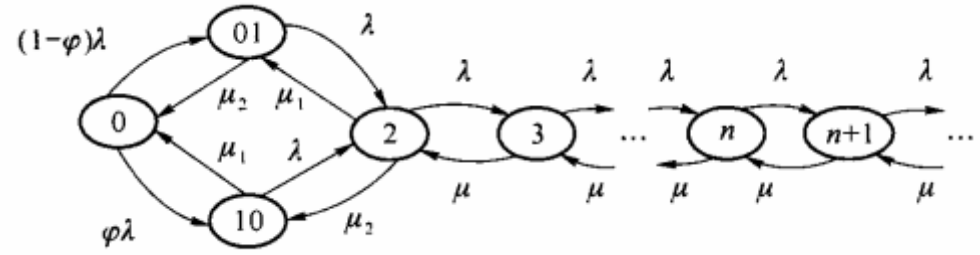


图 15 两窗口服务能力不同的 M/M/2 排队模型的状态转移图[12]

定义$\rho=\frac{\lambda}{\mu}$，$\alpha=\frac{\mu_2}{\mu_1}$，可得平均队长

$$L=\frac{\rho_1(1+\alpha)}{1-\rho_1}\frac{1+(1+\alpha)\rho_1-(1-\alpha)\varphi}{\alpha(1+2\rho_1)+\rho_1[1+(1+\alpha^2)\rho_1-(1-\alpha^2)\varphi]}$$

平均响应时间

$$\mathrm{T}=\mathrm{W}=\mathrm{L}/\lambda$$

将μ_1、μ_2和φ取不同的值，可得到平均交易响应时间随业务量λ的变化情况。如图 16 所示。从图 16 可知，

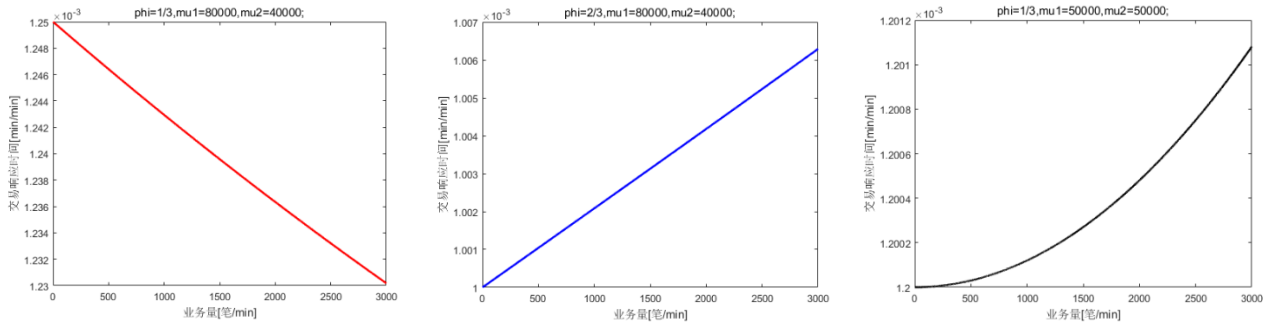


图 16 μ_1、μ_2和φ取不同的值时平均交易响应时间随业务量λ的变化

图 16 表明，μ_1、μ_2和φ取不同的值时，平均交易响应时间随业务量的增大而增大或减小，说明不同的交易分流比例和不同类型的响应时间等会造成不同的响应时间，通过合理设置不同的交易的并发处理能力等可在一定范围内是构造交易响应时间随业务量增大而降低的模型。交易响应时间随业务量增大的一种解释是，业务量越高，响应时间低的交易类型完成越快，占比越大，使得平均响应时间越短。如能设置更多的服务类型，应能得到更加接近实际的响应时间-业务量关系模型，但高阶排队模型求解复杂。

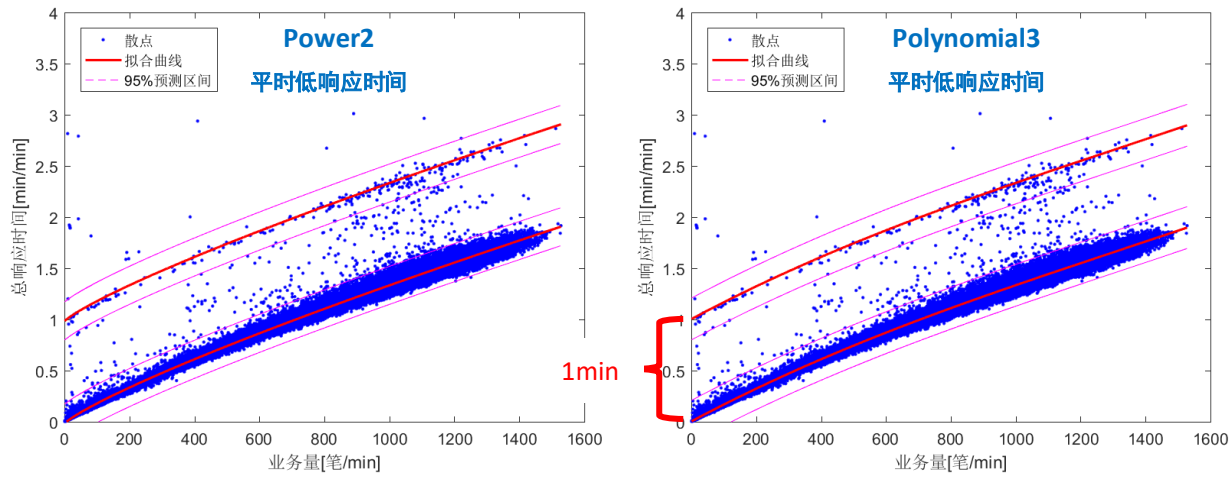
这里利用回归的方法，对交易总响应时间与业务量的函数关系进行拟合。由平均交易响应时间随业务量的增大而降低，可知二者近似与反比关系，则可用幂指数函数或者对数函数来拟合总响应时间和业务量的关系，当然，也可以使用万能的多项式函数。

将总响应时间随业务量的变化关系分成平时低响应时间（包括 1 月春节后、2 月、3 月 1-18 日、3 月 23-31 日、4 月 1-15 日、4 月 20-23 日）、平时高响应时间（包括 3 月 19-22 日、4 月 16-19 日）和春节前（1 月 23-27 日）三类，分别用幂函数（Power2）、三次多项式（Polynomial3）进行拟合，拟合结果如图 17 和表 2 所示。

图 17 中每张图中的拟合曲线有两条，上方的拟合曲线由下方的拟合曲线向上平移 1min 得到。图 17 中同时给出了用拟合曲线进行预测的置信水平$\mathbf{1}-\boldsymbol{\psi}$为 95%预测区间。

幂函数（Power2）、三次多项式（Polynomial3）的拟合效果均很好，均可采用。这里的 1min 平移发生的可能原因是某些交易等待接受响应的时间过长超过了 1min，系统认为该数据为异常数据，自动终止交易。因此，这种类型的数据也可认为是异常数据。异常数据也包括在交易等待时间远超过 1min 的情况。

可据此进行异常点分离。假设总响应时间水平的高低由银行根据 ATM 交易系统的运行状态事先确定，首先确定日期是否为如春节一样的重大节日、以及该日的响应时间水平高低，以确定应该使用的函数参数。将每张图下方拟合曲线的预测区间上界$\left(\overline{\boldsymbol{\Theta}_{\boldsymbol{T}_{\boldsymbol{total}}}}\big|_{\mathbf{1}-\boldsymbol{\psi}}\right)_{\mathbf{1}}$作为正常点与异常点的分界线，在该曲线上方的点即为异常点。当然，也可以定义最上方的预测区间上界上方的点为第一类异常点，上下方两条拟合曲线预测区间上界之间的部分为第二类异常点，分别提取。



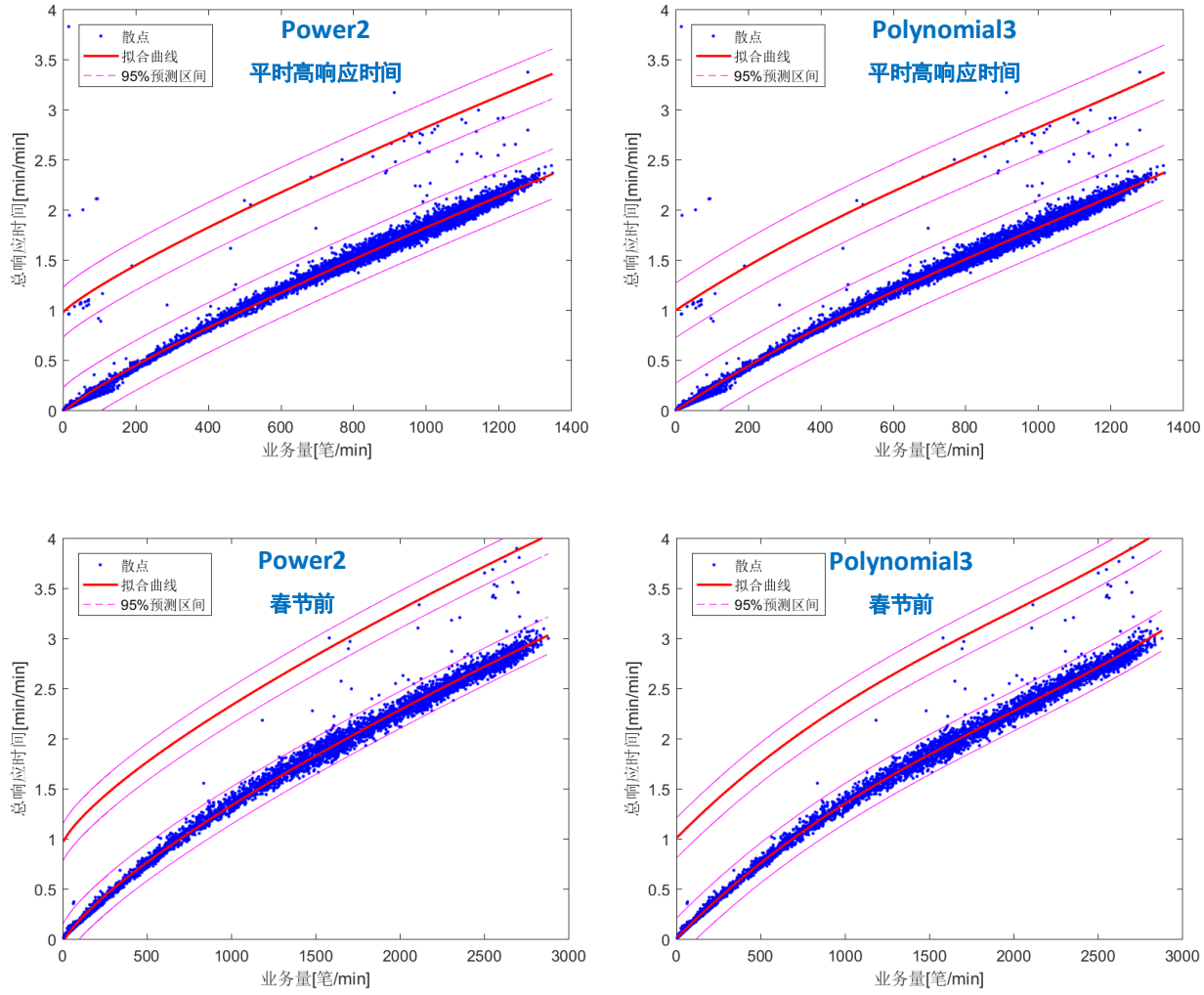


图 17 三种时段下总响应时间与业务量的关系拟合结果

表 2 总响应时间与业务量关系拟合结果

时段	$T_{total}(x) = a_T x^{b_T} + c_T$	$T_{total}(x) = p_1 x^3 + p_2 x^2 + p_3 x + p_4$
平时低	$a_T = 0.00427, (0.004211, 0.00433)$ $b_T = 0.8336, (0.8317, 0.8356)$ $c_T = -0.02054, (-0.02161, -0.01947)$	$p_1 = 1.223 \times 10^{-10}, (1.14 \times 10^{-10}, 1.305 \times 10^{-10})$ $p_2 = -4.822 \times 10^{-7}, (-4.983 \times 10^{-7}, -4.661 \times 10^{-7})$ $p_3 = 0.001691, (0.001683, 0.0017)$ $p_4 = 0.002798, (0.001953, 0.003643)$
平时高	$a_T = 0.005271, (0.005029, 0.005514)$ $b_T = 0.8484, (0.842, 0.8548)$ $c_T = -0.02795, (-0.03256, -0.02333)$	$p_1 = 2.807 \times 10^{-10}, (2.375 \times 10^{-10}, 3.24 \times 10^{-10})$ $p_2 = -8.345 \times 10^{-7}, (-9.15 \times 10^{-7}, -7.54 \times 10^{-7})$ $p_3 = 0.002377, (0.002337, 0.002417)$ $p_4 = -0.004942, (-0.008666, -0.001218)$
春节前	$a_T = 0.00739, (0.007162, 0.007618)$ $b_T = 0.7575, (0.7537, 0.7614)$ $c_T = -0.0555, (-0.05988, -0.05111)$	$p_1 = 7.126 \times 10^{-11}, (6.759 \times 10^{-11}, 7.492 \times 10^{-11})$ $p_2 = -4.253 \times 10^{-7}, (-4.401 \times 10^{-7}, -4.106 \times 10^{-7})$ $p_3 = 0.0017, (0.001685, 0.001716)$ $p_4 = 0.004476, (0.001329, 0.007622)$

3.4 基于 Gauss 拟合的业务量时间序列聚类分析

业务量时间序列如图 18 所示。由图 18 可知春节前后的业务量峰值与平时存在明显差异，工作日和非工作日在日业务总量和峰值上看不出明显的周期性。

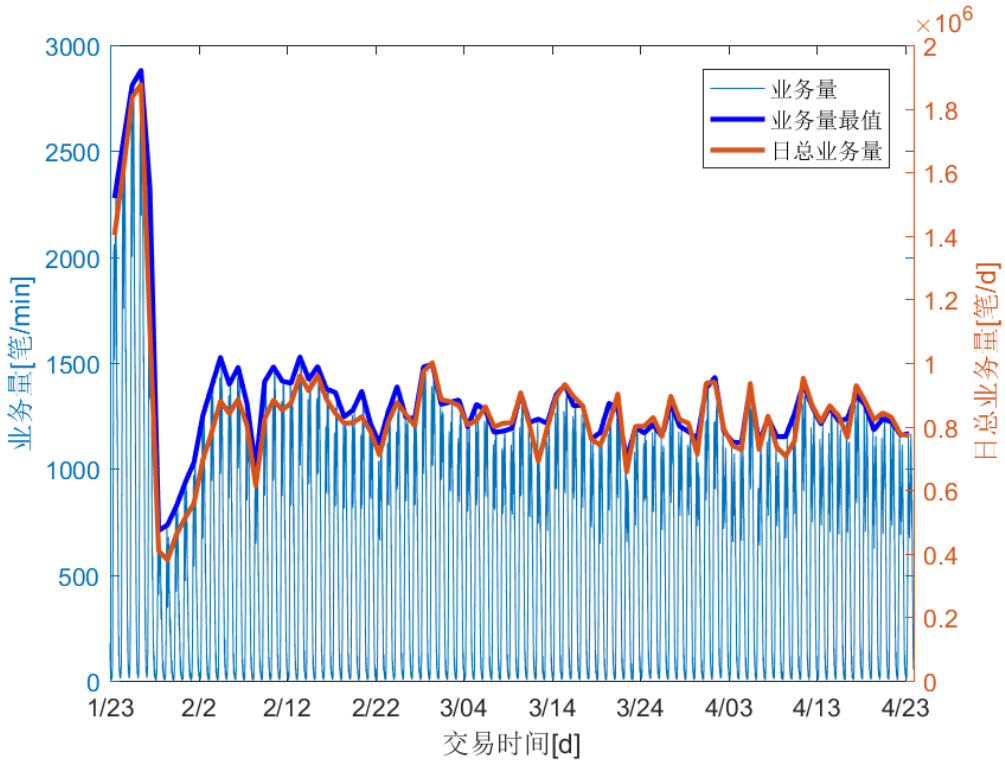


图 18 业务量时间序列

将 1 月 23 日至 4 月 23 日所有天的业务量时间序列按月汇总，如图 19 所示。从图中可以看出个别天的数据很异常，不同于正常的 Pattern，如 1 月 27 日（除夕）。且不同日期的业务量峰值变化很大，尤其是 1 月 27 日前业务量的峰值明显高于其他日期的业务量峰值。1 月 27 日后几天，业务量峰值又明显小于其他日期。除夕当天业务量从上午到下午有一个缓慢降低的过程。这一点可从春节前人们大量购买年货，春节后进入假期，人们很少购物来解释。

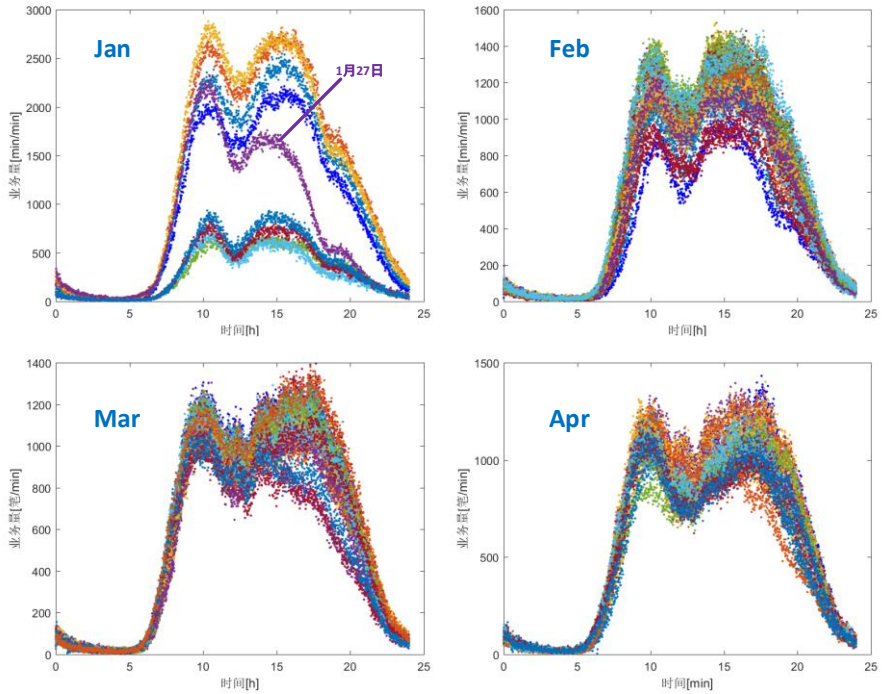


图 19 1 月份至 4 月份每日业务量随时间变化

3.4.1 数据预处理：确定聚类参数

按照设计思路，我们首先需要对每天的数据进行粗略地拟合，并得到相关的参数，拟合的具体代码实现参见文件“Model1_1.m”，思路就是用两个 Gauss 分布的叠加去近似，即

$$M(t) = m_1 \cdot \exp(\sigma_1 (t-\mu_1)^2) + m_2 \cdot \exp(\sigma_2 (t-\mu_2)^2)$$

其中 $m_1, \sigma_1, \mu_1, m_2, \sigma_2, \mu_2$ 为模型参数。

图 20 是对 4 月 6 日的拟合结果。其中，横轴表示时间，1 表示 24 点，其他地方均匀划分，例如 0.75 就表示下午 18 时；而纵轴表示交易量。

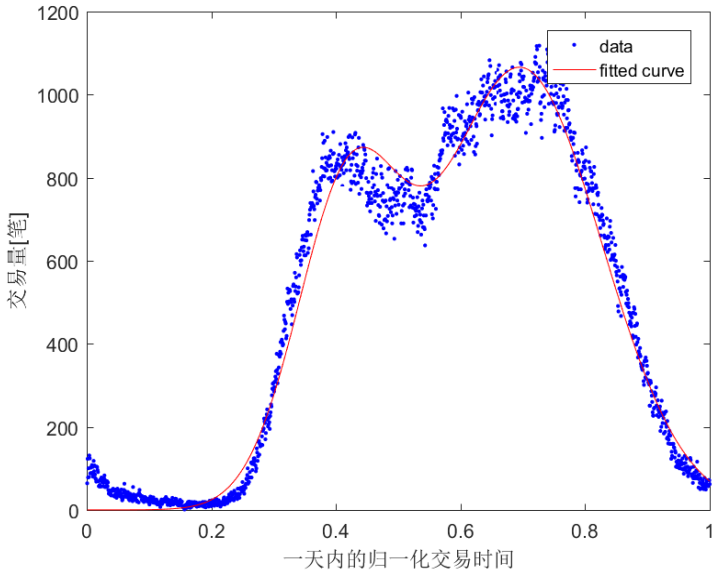


图 20 4 月 6 日的拟合结果

通过图像可以看出，在 95%的置信水平下，$m_1 = 742.4\,(731.6, 753.3)$，$\sigma_1 = -71.16\,(-73.4, -68.92)$，$m_2 = 1063(1055, 1071)$，$\sigma_2 = -29.67\,(-30.29, -29.05)$ 后面所对应的区间为置信区间，另外两个参数 $\mu_1 = 0.4201, \mu_2 = 0.6965$，为手动确定，以使函数收敛于期望中的形式，这里采取的方法是在左右两个峰值处分别选取一个极值区间，作为对 μ_1, μ_2 的近似（具体方法详见代码）。

通过这种方法，我们就可以将 91 天的参数都确定下来，具体参数表请查看附件“paras.xls”。

3.4.2 聚类参数的提取：聚类分析

在确定了这些参数的基础上，我们就可以开始对日期数据点进行聚类分析了。（聚类分析的具体实现参见附件“Model1_2.m”）考虑到表征曲线的参数有 6 个，在聚类之前需要对参数进行降维，降维的思路是忽略 σ_1, σ_2 的模型的影响，将 $m_1 + m_2$ 和 $\mu_2 - \mu_1$ 作为两个参数，考虑所有的日期数据点在其中的分布情况，然后根据分布进行分类。其中，$m_1 + m_2$ 表示两个峰的峰值之和，基本上可以表征日总交易量，而 $\mu_2 - \mu_1$ 表示两个峰的峰间距，即交易高潮间的时间差，这两个参数能够很好地表征这个曲线的特征。关于这两个参数的平面散点图分布如下图所示。

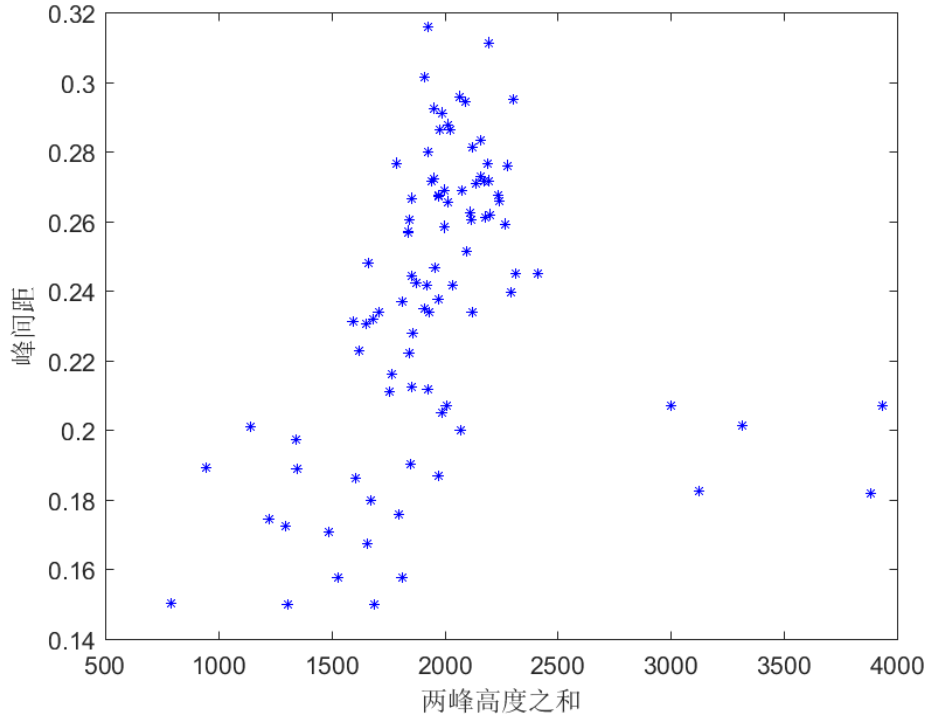


图 21 以日期作为散点的交易量特征参数的散点分布图

图中横轴表示 m_1+m_2，纵轴表示 $\mu_2-\mu_1$，图中横轴表示 m_1+m_2，纵轴表示 $\mu_2-\mu_1$，我们采用的聚类分析的方法是 k-Means 算法(取 k=3).分类的效果图如下图所示：

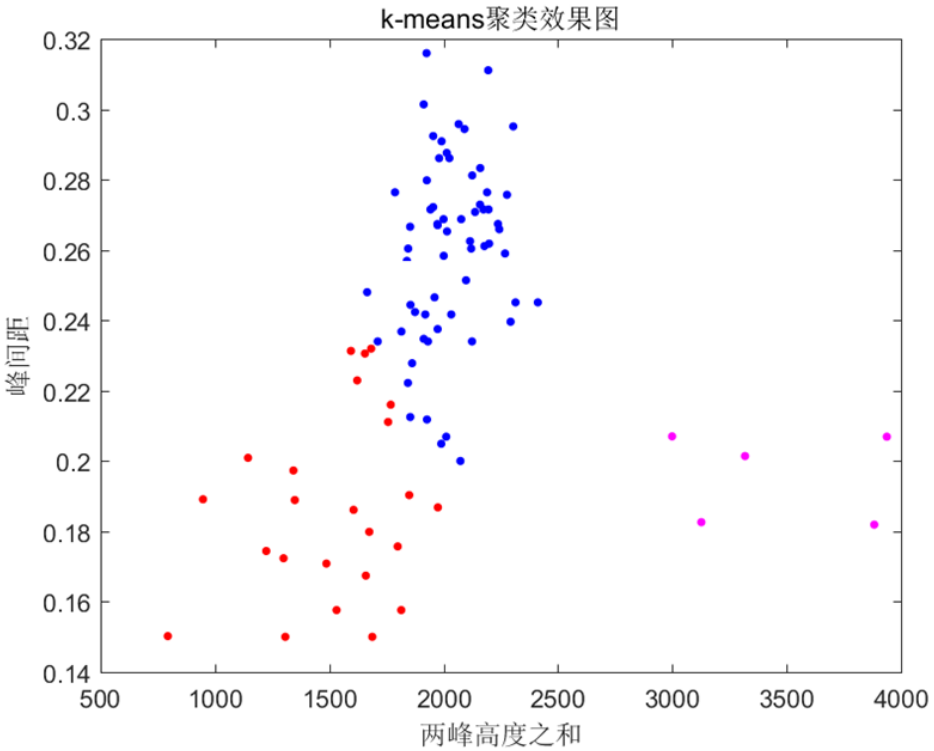


图 22 k-means 聚类效果图

表 3 散点图左下角所分布的日期节点

编号	月份	日期	编号	月份	日期
6	1	27	26	2	17
7	1	28	35	2	26
8	1	29	38	3	1
9	1	30	46	3	9
10	2	1	49	3	12
12	2	3	55	3	18
13	2	4	59	3	22
14	2	5	61	3	24
16	2	7	63	3	26
21	2	12	67	3	30
22	2	13	78	4	10
25	2	16	88	4	20

而右边所包括的日期为：

表 4 散点图右方所分布的日期节点

编号	月份	日期	编号	月份	日期
1	1	22	4	1	25
2	1	23	5	1	26
3	1	24			

其中第一列为该日的编号，后两列为对应的月份和日期，可以看出，第一类集中表征春节之后的一段时间，即 1 月 28 日到 2 月 7 日（1 月 28 日是春节），而第二类集中表征春节之前的一段时间，即 1 月 19 日到 1 月 27 日。在已知这两类的基础上（暂且统称为春节类），结合题目中所给的建议：工作日和非工作日存在差别，于是我们就绘出了针对以上三类的参数散点图：

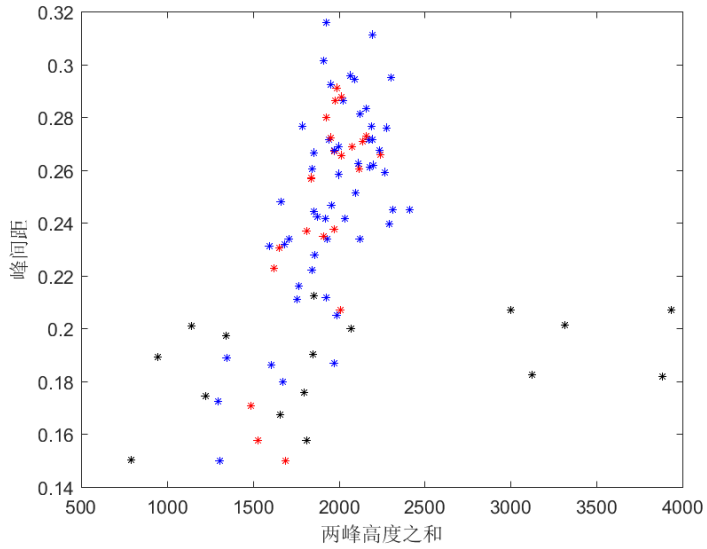


图 23 聚类后的以日期作为散点的交易量特征参数的散点图分布
（黑色表征春节类，红色表征非工作日，蓝色表征工作日）

可以看出，春节类和非春节类之间还是可以看出明显的差异的，而工作日与非工作日之间并不能提取出

明显的特征区分。

为了进一步确定工作日和非工作日之间的区别，我们还做了进一步的检验。即基于原始数据计算出每一天的总交易量，然后绘出这几类数据点关于日总交易量的散点图分布（具体实现见附件中的代码“Model1_3.m”），如下图所示：

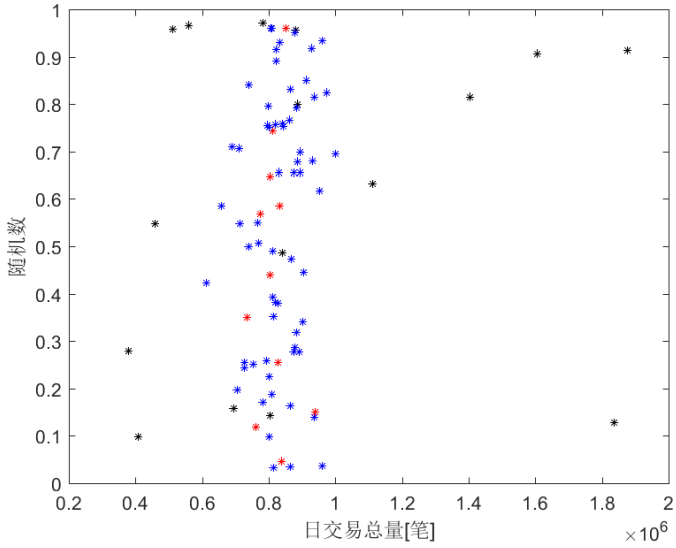


图 24 以交易量为参数的日期节点的二维展开散点图分布
（黑、蓝、红点分别表征春节、工作日、非工作日）

其中横轴表征日总交易量，纵轴为所取的 0~1 间的随机数，没有实质含义，只是为了避免所有的数据点全堆积在一条直线上难以区分辨认。图中，黑、蓝、红点分别表征春节、工作日、非工作日，从图中可以比较明显的看出，春节类可以明显区分开，而工作日和非工作日仍是“杂糅”在了一起，无法区分。

3.4.3 日交易 pattern 的分析

刚才的分析集中在关于交易量的分析，发现工作日和非工作日之间没有明显的区别。接下来，我们从另一个维度，进一步论证这个问题，即基于日交易量 pattern 的分析。我们将去除春节前后的所有的点按照工作日和非工作日进行分类，工作日着蓝色，非工作日着红色，将它们全画在如下图所示的这张散点图上。可以发现，从 pattern 上来看，两者之间没有一个明显的区别或者说倾向性。

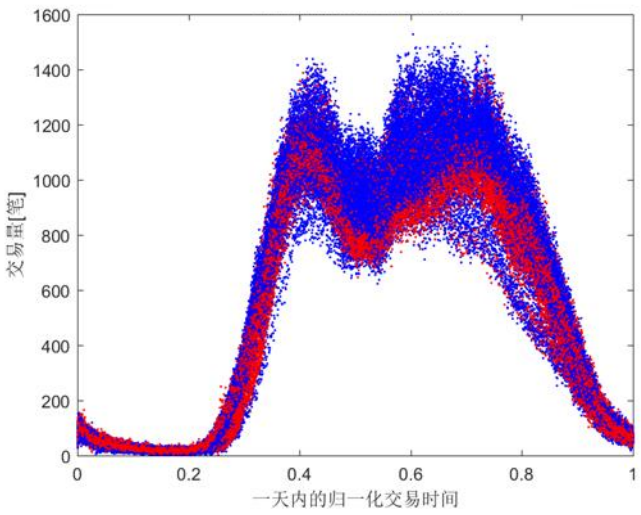


图 25 非春节前后的所有数据点

因此，最终我们就得到了一个看似不合理而又合理的结论：工作日和非工作日的用户使用 ATM 机的取款情况没有太大的差别。说它看似不合理是因为不太符合人的直观感觉，而又说它合理，是因为 ATM 机取款这种特殊形式所致。众所周知，ATM 机取款与传统的柜台办公相比，在时间、空间以及效率上都有很大的优势，时间上，基本上都是支持 24 小时服务的，而空间上，由于成本相对较低，因此城市中架设的 ATM 机数量远远大于商业银行柜台的数量，而在效率上，省去了大量等待排队的时间。对于忙碌中的上班族，在上下班途中，顺路使用 ATM 机取款也是一种非常常见的生活姿态了。因此，从这个角度来说，这样的结果也在预期之内。

3.4.4 基于分类的模型拟合：原点位移->双 Guass->三 Gauss

依据上述聚类分析的结果，将日期分成三类（春节前、春节后、平时）进行建模分析。正式建模时，为了使得毛刺点尽可能得少，图像更加平滑，针对某个时间点的前后五分钟取平均值作为该时刻的标准值，并由标准值出发绘制出来各自的标准曲线（具体实现见附件代码“Model1_4.m”），如下图所示。

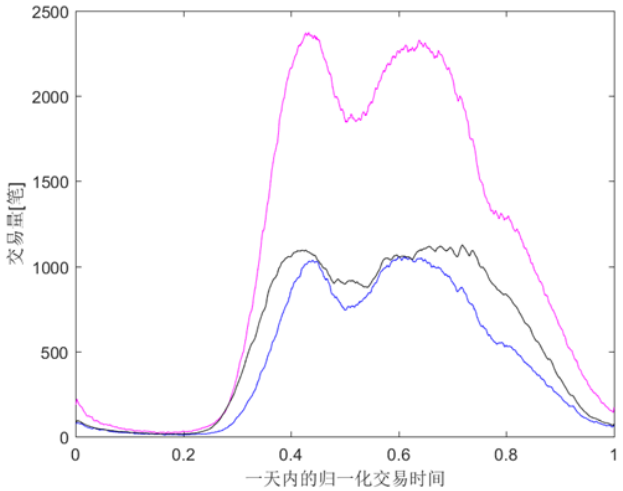


图 26 各类日期的交易量随时间的变化关系
（红线为春节前，黑线为平时，蓝线为春节后）

其中横轴代表某天中时间，纵轴表示交易量，图中红线为春节前，黑线为平时，蓝线为春节后。注意到最左边图像都有一个明显的凸起的部分，因此将这部分嫁接到图像最右端（这里共取了 216 个数据点，即 3.6 个小时，这里的实际意义是我们发现凌晨 3：36 分左右的交易量是最少的），再来建模就相对而言更为合理一些，毕竟一天的划分也是人为的，这样的处理并没有造成质上的区别，只是为了使得模型的拟合效果更好一些。

同时，在真正建模时，我们发现，采用两个 guass 分布建模时会导致中间凹陷下去那一部分略微抬起。下图展示的就是一张采用双 Guass 拟合的效果图，可以看出，这个凹陷现象还是挺明显的。

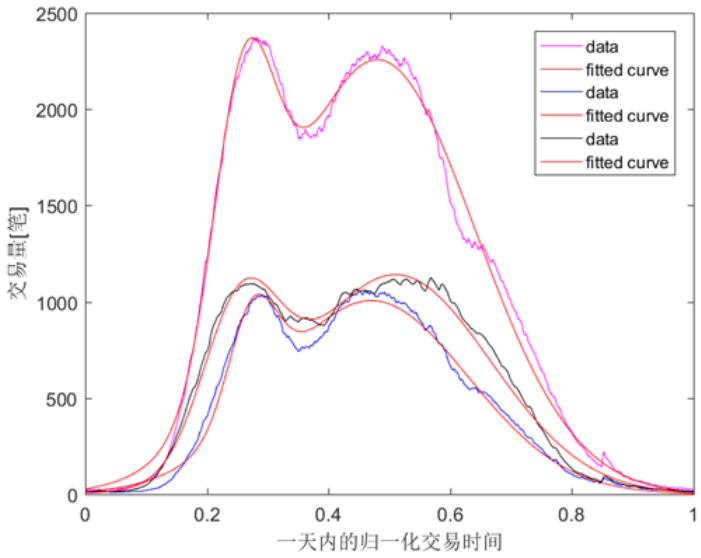


图 27 各类日期的业务量随时间变化关系的双 Gauss“标准模型”近似
（红线为春节前，黑线为平时，蓝线为春节后）

因此，在这个地方我们加了一个反 gauss 分布，将这部分凸出来的部分抵消掉。当然，这个反高斯有没有实际意义，还需要进一步的研究工作才能确定。以下是我们拟合的结果以及相应的图像（平移后的）：

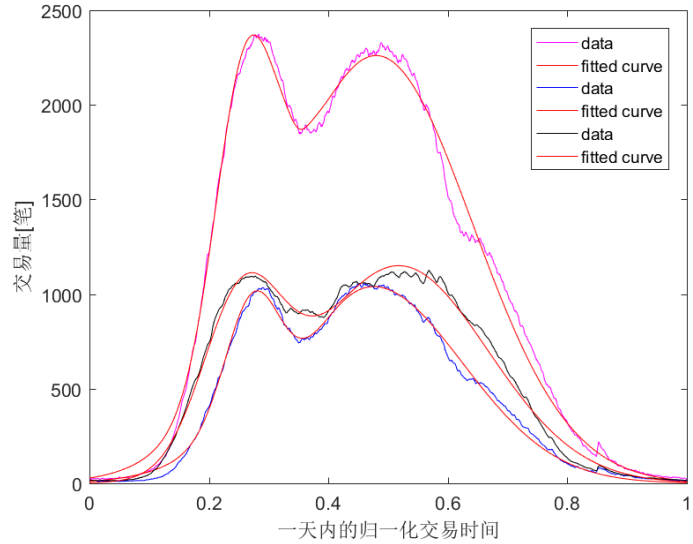


图 28 各类日期的交易量随时间变化关系的“标准模型”近似

所用基本模型：

$$M(t)=m_1\cdot\exp(\sigma_1(t-\mu_1)^2)+m_2\cdot\exp(\sigma_2(t-\mu_2)^2)-m_3\cdot\exp(\sigma_3(t-\mu_3)^2)$$

各模型参数及 95%置信水平下的置信区间：

紫红色（年前模型）

$$\begin{gathered}\mathrm{m}_1=1420(1405,1436)\\ \sigma_1=-195.9,(-201.3,-190.4)\\ \mathrm{m}_2=2261,(2253,2269)\\ \sigma_2=-19.17,(-19.36,-18.98)\\ \sigma_3=-2577,(-5758,603)\end{gathered}$$

其中，取定 $\mu_1=0.26,\mu_2=0.48,m_3=50,\mu_3=0.35$

蓝色（年后模型）

$$\begin{gathered}\mathrm{m}_1=653.1,(645.9,660.4)\\ \sigma_1=-236.3,(-241.6,-231.1)\\ \mathrm{m}_2=1046,(1042,1050)\\ \sigma_2=-20.82,(-21,-20.65)\\ \sigma_3=-165.6,(-179.5,-151.7)\end{gathered}$$

其中，取定 $\mu_1=0.275,\mu_2=0.47,m_3=190,\mu_3=0.33.$

黑色（平时模型）

$$\begin{gathered}\mathrm{m}_1=869.7,(862.4,876.9)\\ \sigma_1=-112.4,(-114.4,-110.3)\\ \mathrm{m}_2=1185,(1179,1190)\\ \sigma_2=-21.08,(-21.3,-20.86)\\ \sigma_3=-32.74,(-37.21,-28.28)\end{gathered}$$

其中，取定 $\mu_1=0.255,\mu_2=0.51,m_3=100,\mu_3=0.335.$

可以看出，最后的拟合效果很好。

这里引入了 9 个参数（业务量 Gauss 分布的波峰值 m_1,m_2,m_3，表征业务量 Gauss 分布的标准差的参数 $\sigma_1,\sigma_2,\sigma_3$，业务量 Gauss 分布的波峰对应的时间 μ_1,μ_2,μ_3）对业务量的时间变化进行表征。下节将引入业务量的评分α，以克服 3σ准则判断业务量异常的不足，更直观而准确地判断业务量是否异常。

3.5 小结

本章首先将数据分为交易成功量与业务量、交易总响应时间与业务量和业务量时间序列三部分进行分析，发现了交易成功量与业务量的线性关系，确定了直线的斜率即平均交易成功率和给定置信水平下的交易成功量的预测区间，为交易成功率异常检测提供了依据。发现了交易总响应时间与业务量的两种不同时段下确定的函数关系，建立了排队模型，详细分析了响应时间随业务量升高而降低的原因和交易响应时间可能出现的异常类型。将交易总响应时间分为春节前、平时高值、平时低值三个时段进行拟合，确定了该函数关系和利用该函数关系进行异常检测的给定置信水平下的预测区间。

基于 Gauss 拟合对该业务量时间序列进行聚类分析，利用聚类分析的结果将数据分为春节前、春节后和

平时三类。分别对每类下的日交易量 Pattern 进行了分析，确定了更加精确的 3Gauss 拟合模型。

3.6.1 参数选择汇总

3.6.1.1 交易成功量m_s与业务量m的关系相关参数

（1）平均交易成功率：$\bar{n} = 95.61\%$；
（2）交易成功量预测的置信水平：$1-\varphi$（须由银行按照用户体验、工作效率等的要求确定）；
（3）交易成功量置信水平为$(1-\varphi)$的预测区间下界：$\underline{\Theta_{m_s}}\Big|_{1-\varphi}$。

3.6.1.2 交易总响应时间T_{total}与业务量m的关系相关参数

（1）三种时段下总响应时间的函数关系，即拟合参数：$(a_T、b_T、c_T)_{1,2}$（或$(p_1、p_2、p_3、p_4)_{1,2}$）；
（2）总响应时间预测的置信水平：$1-\psi$（须由银行按照用户体验、工作效率等的要求确定）；
（3）三种时段下总响应时间置信水平为$(1-\psi)$的下方拟合曲线的预测区间上界：$\left(\overline{\Theta_{T_{total}}}\Big|_{1-\psi}\right)_1$。

3.6.1.3 业务量m时间序列相关参数

（1）业务量高斯分布的峰值m_1, m_2, m_3，标准差$\sigma_1, \sigma_2, \sigma_3$和归一化时间$\mu_1, \mu_2, \mu_3$
（2）业务量聚类参数$m_1 + m_2$和$\mu_2 - \mu_1$；

4 符号说明汇总

交易时间t[min]
业务量m[笔/min]
交易成功率n[%/min]
平均交易成功率$\bar{n}$[%/min]
交易响应时间T[min/min]
交易总响应时间T_{total}[min/min]
交易成功量与业务量线性关系截距b_s[笔/min]
交易成功量预测的置信水平$1-\varphi$[100%]
交易成功量置信水平为$(1-\varphi)$的预测区间下界$\underline{\Theta_{m_s}}\Big|_{1-\varphi}$[笔/min]
三种时段下总响应时间与业务量的幂指数函数关系的拟合参数$(a_T、b_T、c_T)_{1,2}$
三种时段下总响应时间与业务量的 3 次多项式函数关系的拟合参数 $(p_1、p_2、p_3、p_4)_{1,2}$
总响应时间预测的置信水平$1-\psi$[100%]

三种时段下总响应时间的置信水平为$(1-\psi)$的最高预测区间上界$\left(\overline{\Theta_{T_{total}}}\big|_{1-\psi}\right)_{1,2}$ [min/min]

三种时段下总响应时间的置信水平为$(1-\psi)$的预测区间$\left(\underline{\Theta_{T_{total}}}\big|_{1-\psi}, \overline{\Theta_{T_{total}}}\big|_{1-\psi}\right)_{1,2}$ [min/min]

业务量 Gauss 分布的峰值m_1, m_2, m_3[笔/min]

业务量 Gauss 分布的标准差的参数$\sigma_1, \sigma_2, \sigma_3$[笔/min]

业务量 Gauss 分布的峰值对应的归一化时间μ_1, μ_2, μ_3[1]

业务量评分时段Δt_1[min]

业务量随时间变化的评分α[分]

业务量评分前后相邻时间段Δt_1的差值$\Delta\alpha$[分]

业务量随时间变化预测的置信水平η；

业务量随时间变化的置信水平为η的评分阈值α_{max}[分]和$\Delta\alpha_{max}$[分]

5 任务 2：ATM 交易状态异常检测方案

将交易状态异常分为交易成功率异常、交易响应时间异常和业务量异常三部分分别进行检测。

5.1 业务量异常检测

5.1.1 基于评分机制的业务量异常检测

针对任务 2，一个很朴素的想法就是针对每个小的时间区间，由已知的这些数据的值，给出它的标准差 σ 值，然后采用 3σ 检验的方法，判断所给的交易量数据是否异常，即是否在所给的一个“标准”区间内。这种方法存在两个潜在的问题：

（1）在用 3σ 检验时，仅给出一个点是或不是离群点的信息，针对临界情况无法做到很好的平滑过渡；

（2）没有考虑到历史数据的影响，无法监测到短期内交易量骤降，但仍在“标准”区间内的情况。如下图所示的就是平时（非春节）的所有数据点所展示的散点图（运行附件“Model1_4.m”）：

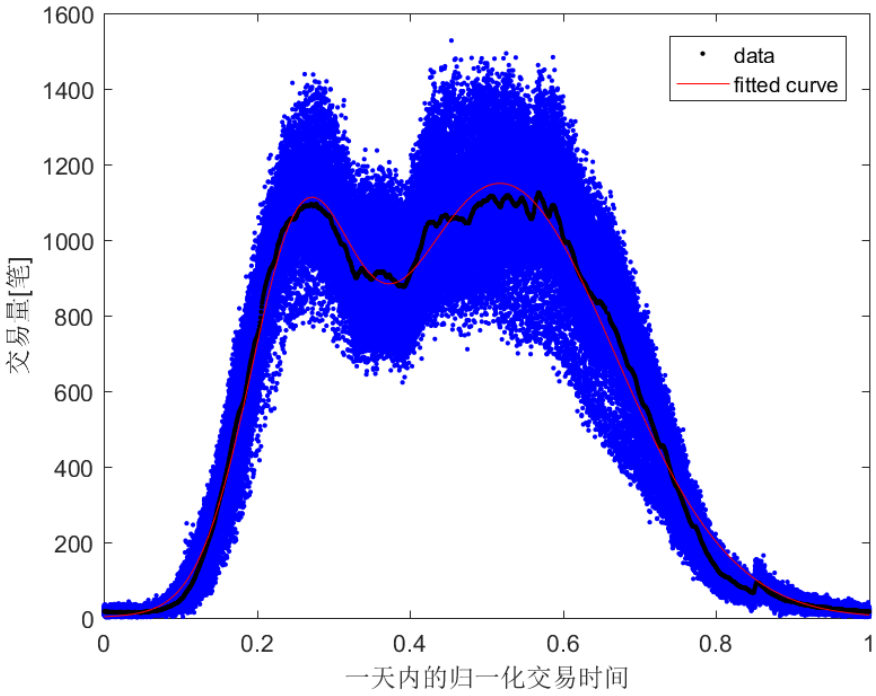


图 31 平时所有日期数据的散点图分布

可以看出，尽管我们模型拟合的效果还算良好，但如果采用3σ检验的方法，无法有效的解决解决上述问题。因此，这里引入评分机制来解决这个问题。

评分机制，顾名思义，就是根据某时间的交易量给出评分α，一般我们理解在当交易量相对较大时，就交易量这项指标出错的概率应该是越低的（不考虑进一步的高阶影响），因此我们定义标准值及标准值以上的区域评分为 10 分，标准值以下的地方按照偏移量给出相应的评分，而在3σ以外给 0 分。中间部分的给分函数可以根据实际需求取，两种比较简单的取法，一个是用线性函数，另一个就是根据 Guass 分布的概率给出。这样，针对每一个时刻，我们都能给出针对其交易量的评分α。当出现以下两种情况之一时系统就会报出预警：

（1）某一段时间区间Δt_1内的平均得分低于设定阈值α_{min}；

（2）某一段时间区间内的平均得分与之前相邻一段时间区间相比，分差低于设定阈值$\Delta\alpha_{min}$。

当然，这两种报警机制的阈值以及时间区间的大小需要结合实际情况，先测试一些数据，然后给出合适的参数。而这些参数本身的变数性也使得模型本身具有了更高的自由度，可以根据实际的需求给出合适“松紧”的预警机制。

同时，我们注意到，上述所给的两种预警方案，每个时间节点上的计算量都是 O(1)的，同时所需要的数据量也只是之前一小段时间区间 cache 中的评分值，因此所占用的空间和计算资源对系统性能的影响都是可以忽略的。

最后采用这种做法，我们通过算法将其实现了出来，如图 32 和 33 所示。当然这个地方由于是个非监督性的学习，因此需要给定一些参数，诸如在目前这些数据中正常点的比例η有多少。这里的阈值和比例可以理解为在置信水平为η时的预测区间下界α_{min}和$\Delta\alpha_{min}$。

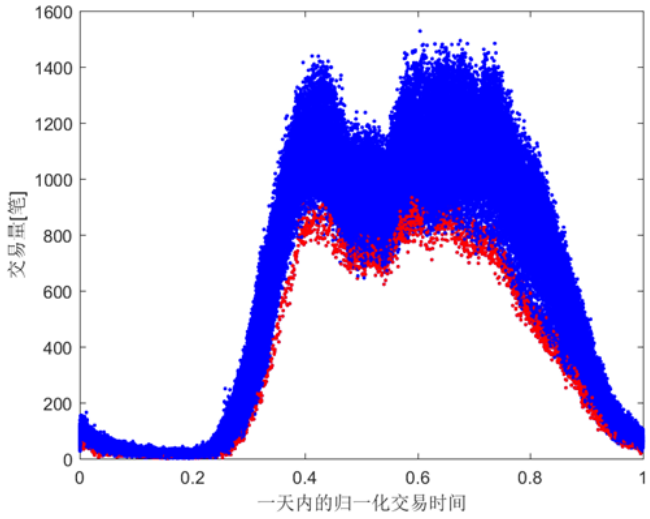


图 32 置信水平为 97%时的异常点分布（$\boldsymbol{\alpha_{min} = 11, \Delta\alpha_{min} = 74, \eta = 97\%}$）

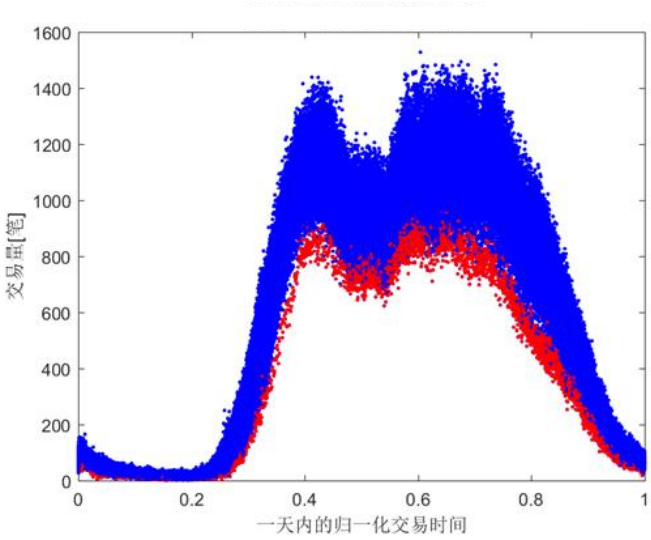


图 33 置信水平为 95%时的异常点分布（$\boldsymbol{\alpha_{min} = 22, \Delta\alpha_{min} = 65, \eta = 95\%}$）

可以看出，通过我们的检测方法，不仅可以把一些交易量很低的异常点检测出来；还可以把一些基于历史数据所展现出的 pattern 的异常点给检测出来，这个也是我们在方案中已经展现出来的，最后的实现结果与我们方案的设计时自洽的。

5.1.2 基于向量空间和可靠性的贡献值分配模型

参考信息检索领域将文档排序的思路，我们又提出了基于向量空间和可靠性的贡献值分配模型。

在信息检索领域中，对于用户所提交的一个 query，针对 query 的词向量与 docs 的词向量进行匹配，计算它们的“相似程度”，进而得到量化的 similarity 值。而另一方面，对于万维网上的亿万个网页以及其间复杂的链接结构关系，我们还需要去评估网页的 authority，一个典型的方法就是 Google 创始人 Larry Page 提出来的 PageRank 算法。最后，我们综合 similarity 信息和 PageRank 值，得到网页总体的 score，针对 score

对网页进行排名，排名高的展现位置就越靠前。

迁移到我们这个问题上来，首先我们做出下面这个假设：

$$m_{t,\tau} = m(HM, TM, HS)$$

其中，HM 表示历史交易量集，TM 表示交易量训练集，HS 表示历史交易量打分集。我们假设认为交易量仅与这三个因素有关，即认为交易量变化满足 Markov 性。如下图我们给出了这种 Markov 性的简单验证，即针对每相邻两天的数据点，观测其相关性，最终计算得到 r=0.2205,相关性很弱。

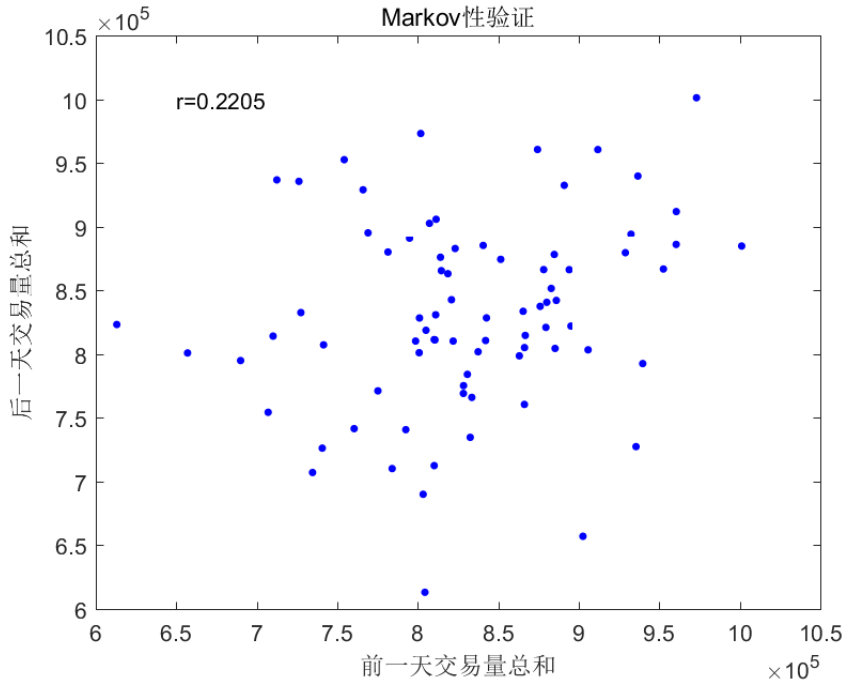


图 34 业务量时间序列的 Markov 性验证

于是，针对这种假设的 Markov 性，我们提出了一个很朴素却很实用的交易量线性预测公式：

$$m_{t,\tau} = \sum_{t' \neq t} w(similarity, score) \cdot m(t', \tau)$$

其中要求 $t' \neq t$ 是出于交叉验证的考虑，即要将测试的数据集和训练的数据集区分开，不然就容易造成过学习的情况。而每天该时刻的 $m(t', \tau)$ 前面需要匹配一个对应的权重因子 w，来决定它的重要性。而这个权重因子主要取决于两个因素：similarity 和 score，两者在 IR 中分别对应于 Doc Index 和 PageRank。

在这里，我们简单地论述一下线性公式的合理性。一方面，由于我们认为交易量分布满足 Markov 性，因此针对于当前研究的这一天而言，与历史上其他天的数据之间是相互独立的，而历史上其他天之间亦是独立的。因此，任何一个高维关系都可以“干干净净”地拆分成若干个低次幂的乘积之和。但根据齐次性，我们知道，这些高次项是不存在的。另一方面，有人认为，与神经网络的架构类比的话，应该还需要添加一个偏移量 bias。这个想法其实也是合理的，但却同样也是没有必要的。这是因为在这个问题中，我们的所有有效信息全部来自于 HM，TM 和 HS，而并没有已标记的 Label 可供我们作为学习的参考，因此，即使存在 bias,最终也能化为 $m(t', \tau)$ 加权求和的形式。因此，bias 的存在亦是没有必要的。

基于上述的交易量线性预测公式，我们可以看出实质上是一个组合加权的效果，其中最重要的就是权重因子 w,它主要取决于两个因素：similarity 和 score。其中，similarity 表征历史上某天的 pattern 与测试天已

知部分 pattern 的相似程度，这里我们采用它们在向量空间中的夹角来表示，亦即其曼哈顿距离，公式如下所示：

$$similarity = \sum_{\tau<\tau_0} \left| m(t_1,\tau) - m(t_2,\tau) \right| .$$

其中，$\tau < \tau_0$ 体现了 Markov 性，即某天一定时刻的交易量仅与该日之前的交易量有关。而 score 所表征的是历史上某天 pattern 的可靠程度，这与该日异常检测中的正常率息息相关。因此，我们建立在人工打分的基础上，得到每天相应的正常率，据此作为下一次的 score 值，每轮根据已知的 score。经过几轮迭代，可以得到一组收敛的 score 值，即作为一个合理的 score，当然，由于评分标准之间参数可能存在着细微的差异，对最后的打分可能也会产生细微的影响。因此，在实际应用过程中，也可以将参数的选取和打分的确定两者作为一个整体，再进行更多次地迭代，产生更为精确的标准。考虑到由此产生的效益甚微，因此在我们的模型中没有做这方面的尝试。这就相当于在用共轭梯度法解线性方程组时，尽管能够最终得到精确解，但考虑到计算量、实际计算精度和需求等多方面因素，我们也会在迭代若干步后停止迭代。

有了 similarity 和 score 之后，我们就可以设计我们的权重因子函数了。为此，我们就需要先确定我们的评价标准，使得在最终参数确定时能够起到“效能函数”的作用。为此，我们选取 3 月 12 日、3 月 22 日、3 月 30 日、4 月 10 日四天作为评价数据，以它们的后半天（720 个数据点）的异常点个数减去前半天的异常点个数作为效能函数。最终调整参数使得效能函数取得最大值。这是因为这几天一个显著的特点是 pattern 在后半天出现较为严重的问题。尽管我们还没有给“异常”下一个明确的定义，但这种在 pattern 上与标准（均值）相比，在某一段有一个骤低，我们基本就可以归结为异常了。而采取等量的正例样本和反例样本是为了避免过学习的情形。为了在参数上可以调整，我所选取的含参权重模型为

$$w(similarity, score) = \frac{e^{-\lambda \cdot similarity} score^p}{\sum e^{-\lambda \cdot similarity} score^p} .$$

其中，分母起到了归一化的效果。

先取定 p=1 时，我们来研究随着参数 λ 的变化，效能函数值的变化规律，得到的变化曲线图如下图所示：

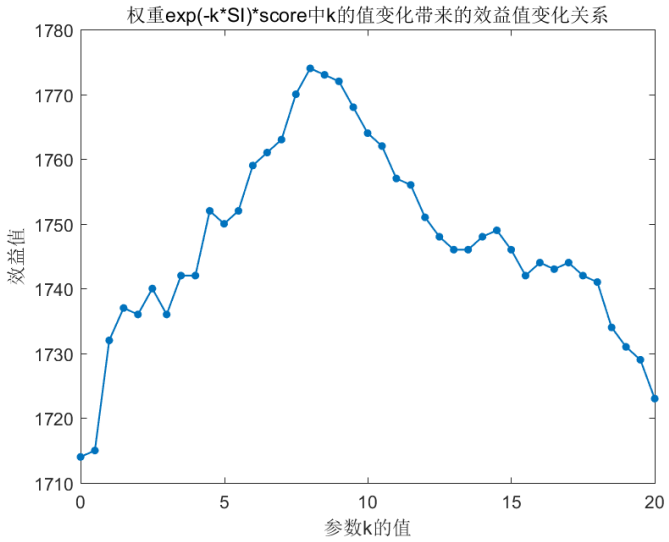


图 35 权重 exp(-k*SI)*score 的值变化带来的效益值变化关系

可以发现，大致在 $\lambda = 8$ 时，效能函数取得最大值。接着，我们取定 $\lambda = 8$，研究随参数 p 的变化，效

能函数的变化趋势，得到的曲线图如下图所示：

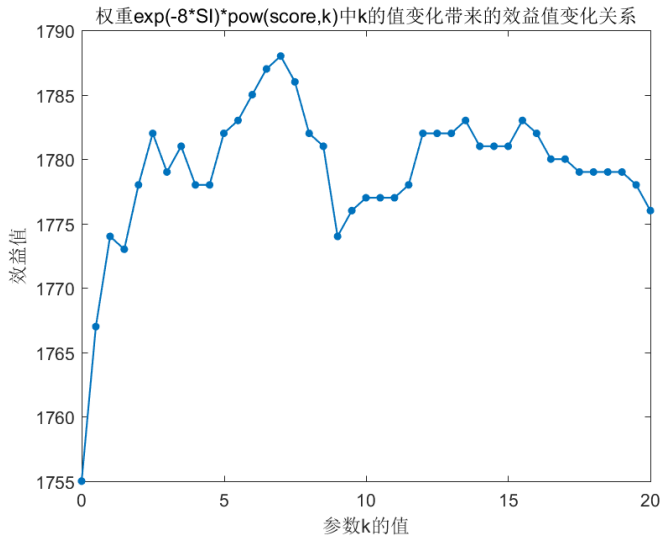


图 36 权重 exp(-8*SI)*pow(score,k)中 k 值变化带来的效益值变化关系

可以发现，在[1,20]区间内，效益值随参数 p 的变化不是很明显，我们在其中取 p=7 即可。因此，最终我们得到了一个不错的权重函数：

$$w(similarity, score) = \frac{e^{-8 \cdot similarity} score^7}{\sum e^{-8 \cdot similarity} score^7}.$$

据此，我们就能够拿这个权重函数，以及已知的数据集进行预测。下图展示的就是对原有平时数据的检测：

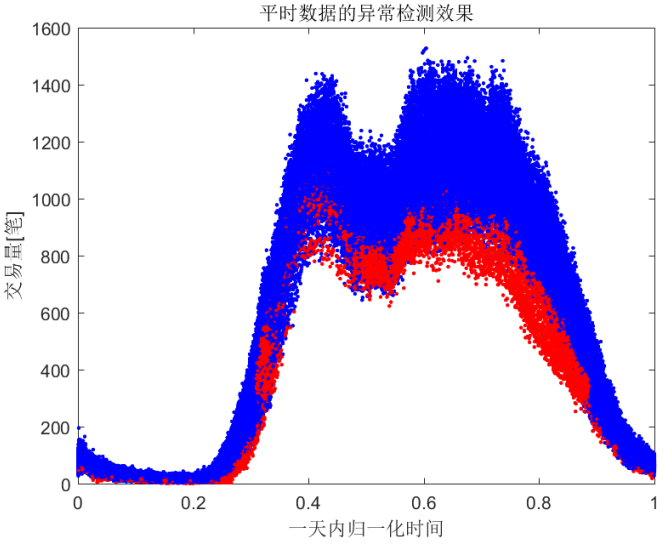


图 37 对原有平时数据的检测

说几个具体代码实现时的细节：

一个是平滑处理，由于原数据中数据点的波动比较大，因此，采用这个检测时，如果在一个孤立点上检测到“异常”，很有可能也是在正常的数据波动范围之内的。因此，我们需要设置一个阈值 k，使得当有至少连续 k 的点是异常点时，判定这一段的点均为异常点。最终，通过实际的尝试和比较，得到当 k=3 时效果最佳。

另一个是空缺点填充，在对原始数据进行观察的时候，我们发现有些数据点是缺失的，我所采取的解决方案是，对于前一个有数据点的时刻，将数据进行比例 95%~105%之间的随机扰动。这样一方面能够很好的

填补空白数据点的空缺，又同时避免了一个数据值连续出现多次的问题。

还有一个就是 pattern 异常与交易量大小异常的平衡。这个在版本 1 的检测方法里也遇到了同样的问题，当时没有很好地解决这个问题。在这里我们做法是在对标准值(standard)取作预测值(prediction)与平均值(average)的线性组合，亦即：

$$standard = w \cdot prediction + (1-w) \cdot average .$$

其中，prediction 主要监测 pattern 异常，而 average 主要监测交易量大小异常。最终，通过对整体性能的评价，我们得到取 $w = 0.7$ 为宜。

5.1.3 基于统计指导的二维贡献值分配模型

版本 2 的方案借鉴了信息检索领域的思路，但一个不足之处就是在监测上缺乏统计知识的支撑。为此，基于刚才的版本 2，我们又提出了基于统计知道的二维贡献值分配模型。

在这个模型中，我们的一个假设就是任意时刻的交易量 m 是由当天总交易量 M 与该时刻交易量占当天总交易量的比例 η 两个随机变量的乘积组成，且两个随机变量之间是独立的。

于是，我们就可以导出任意时刻交易量 m 的期望值和方差，具体计算公式如下所示：

$$Em = E(\eta M) = E\eta \cdot EM .$$

$$Dm = D(\eta M) = D\eta \cdot \left(EM\right)^2 + DM \cdot \left(E\eta\right)^2 + D\eta \cdot DM .$$

其中，

$$EM_{t,\tau} = \sum_{t' \neq t} w(similarity, score) M(t', \tau) .$$

$$E\eta_{t,\tau} = \sum_{t' \neq t} w(similarity, score) \eta(t', \tau) .$$

$$DM_{t,\tau} = \sum_{t' \neq t} w(score) \left(M_{t',\tau} - M(t', \tau)\right)^2 .$$

$$D\eta_{t,\tau} = \sum_{t' \neq t} w(score) \left(\eta_{t',\tau} - \eta(t', \tau)\right)^2 .$$

在具体实现的时候很多地方与版本 2 很类似，较大的不同，一个就是从一个一维贡献分配模型，变成了二维的，所对应的，计算量也翻了一倍；另一个就是最后检测异常的方法，从开始的分权比例法，变成了传统的统计回归，这样说服力更强一些。

在具体实现的时候，也有几个点需要注意一下，一个就是针对 M 和 η 的期望值计算中的权重，其计算方法与版本 2 相同，但这两者之间也存在区别：一是两者 similarity 的标准不一样，其中是考查交易量大小的 similarity，后者则是针对归一化的 pattern 的相似度；另一个就是两者的 score 也不同，score 取决于具体的检测标准，采用交易量大小和 pattern 判据下的 score 必然会有所区别，这也就导致了它们的 score 集也是

存在区别的。另外，在这个设计过程中，依然沿袭了版本 2 中的平滑处理和空缺点填充机制。

下图展示的就是使用了基于统计指导的二维贡献值分配模型，针对已有的数据点中平时点的异常检测结果：(取的是向下1.65σ作为区分限)

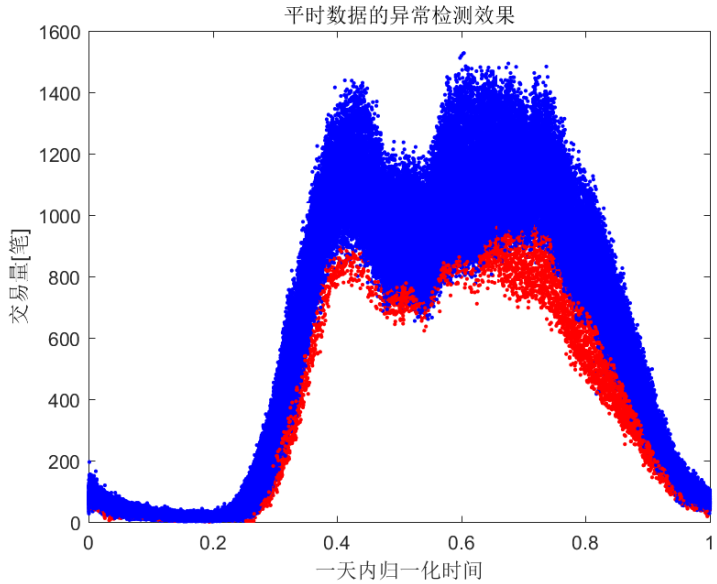


图 38 基于统计指导的二维贡献值分配模型，针对已有的数据点中平时点的异常检测结果

可以看出，检测效果还是很不错的。

5.2 交易成功率异常检测

对于交易成功率，首先根据交易成功率和业务量确定（交易成功量m_s，业务量m），根据交易成功量与业务量的线性关系$m_s = \overline{n}m + b_s$，确定该交易成功量是否低于在预先给定的置信水平$1-\varphi$下的交易成功量预测区间下界$\underline{\theta_{m_s}}\Big|_{1-\varphi}$。若交易成功量不低于预测区间下界，则交易成功率正常，否则交易成功率异常。

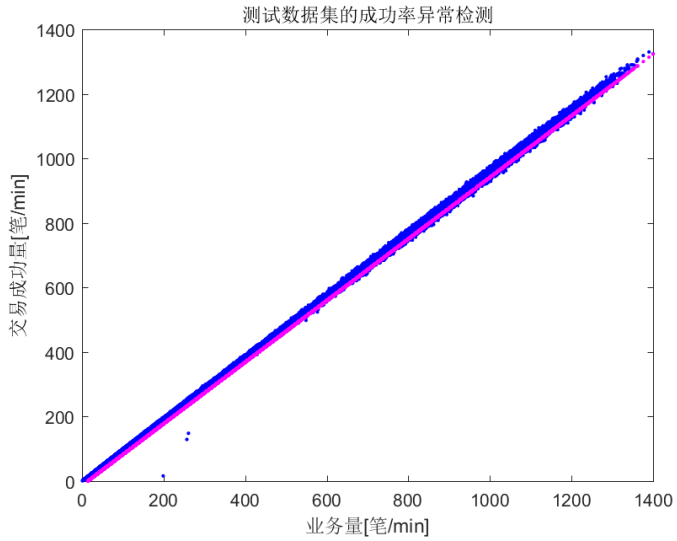


图 39 测试数据集的成功率异常检测

5.3 交易响应时间异常检测

对于交易响应时间，首先根据交易响应时间和业务量确定总响应时间T_{total}，并判断该日期是否是重大节日、以及该日的响应时间高低，据此确定即将使用的函数参数$T_{total}(x) = a_T x^{b_T} + c_T$（或$T_{total}(x) = p_1 x^3 + p_2 x^2 + p_3 x + p_4$），设定总响应时间高于上方拟合曲线预测区间上界和处于两预测区间上界之间的部分均为异常点，判断此时的总响应时间T_{total}是否超出最高预测区间上界$\left(1+\overline{\Theta_{T_{total}}}\big|_{1-\psi}\right)_{1,2,3}$或者位于预测区间带$\left(1+\overline{\Theta_{T_{total}}}\big|_{1-\psi},\ \overline{\Theta_{T_{total}}}\big|_{1-\psi}\right)_{1,2,3}$之间。若否则交易响应时间正常，若是则交易响应时间异常。

这是一种区分不同日期和不同时刻的分层交易响应时间异常检测方法，具体实现方法如下：

（1）首先确定交易响应时间高值日期和低值日期。去掉每天交易响应时间的若干高值和低值，对剩余时间间隔的交易响应时间取平均，求取交易响应时间的日均值，利用 k-means 方法，对日均交易响应时间进行聚类，得到交易响应时间高值日期和低值日期。

（2）选择幂指数函数或者多项式，分别带入高值日期和低值日期不同的拟合参数，确定高低值日期的预测区间边界利用该边界进行异常值提取。

对测试数据检测的部分结果如下，图中已区分了交易响应时间的高低值日期，并分别汇总在两张图中。两张图中的两条线为响应时间是否异常的边界，这里将两条线之间的部分视为一类异常值，红色线上方的值视为另一类异常值。交易响应时间高值的日期是 6 月 11-14 日。

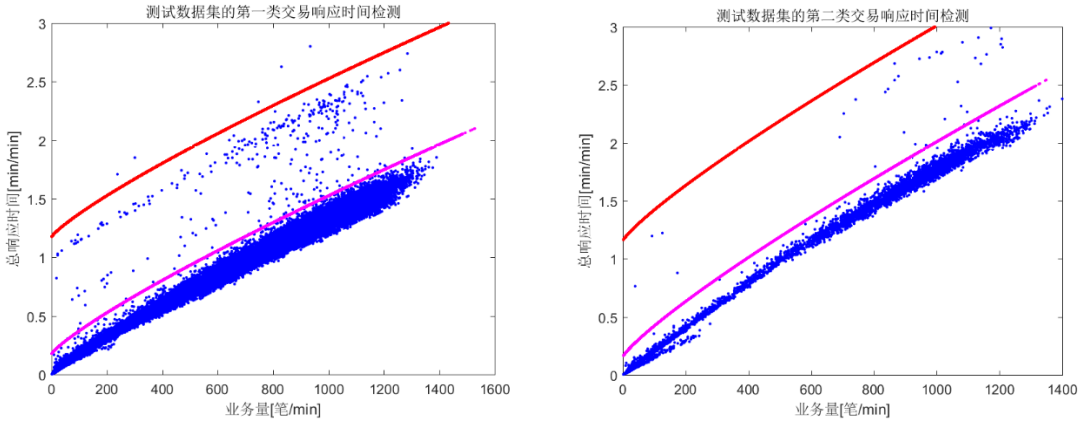


图 40 响应时间测试数据集异常检测示意图

根据任务 1 中确定的参数和分析，拟定了如下的 ATM 交易状态异常检测方案，其中业务量预测以基于评分机制的业务量异常检测模型为例，如图 39 所示。

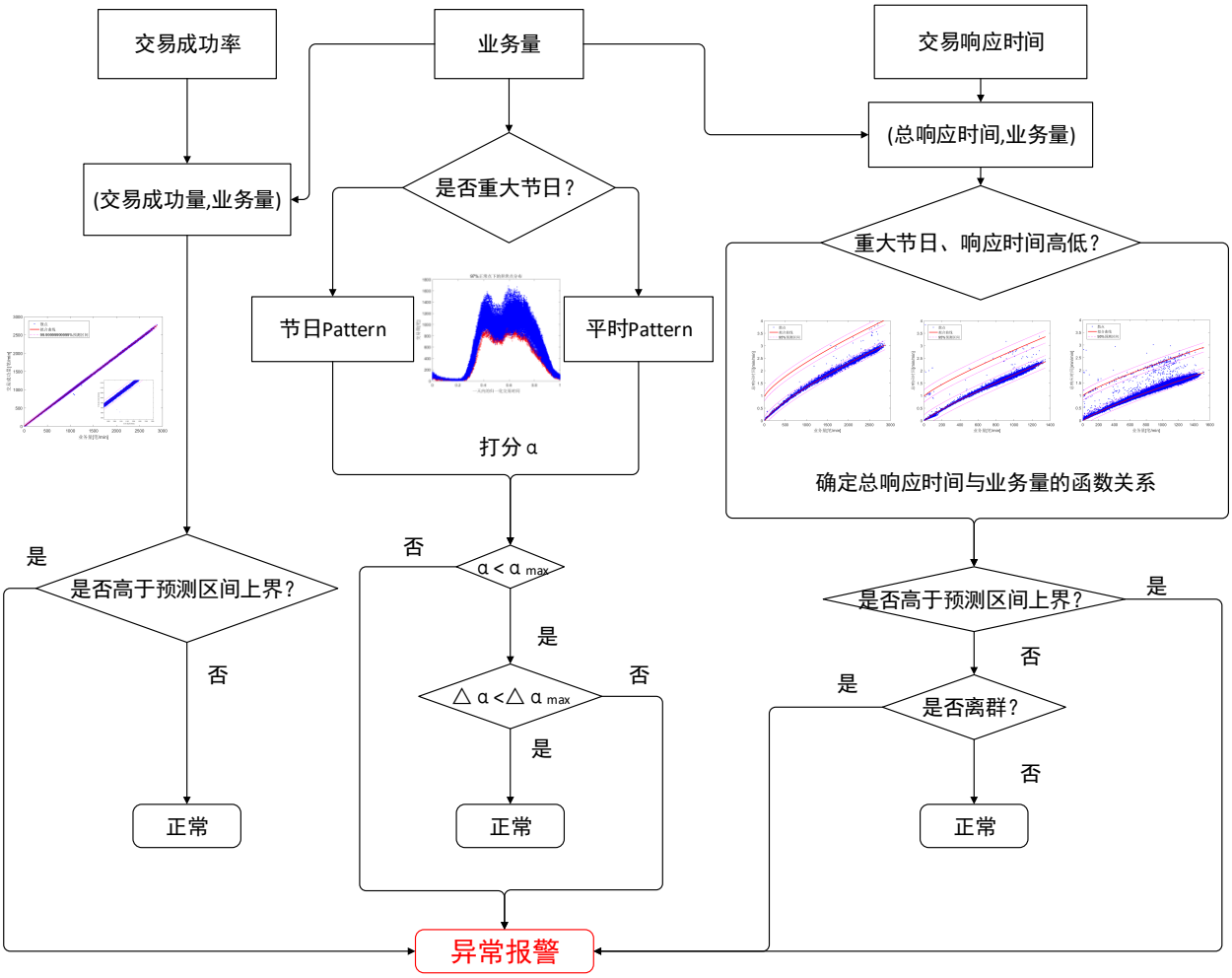


图 41 ATM 交易状态异常检测方案

5.4 减少虚警误报的措施

（1）主动监控与被动监控结合。采用交易仿真和交易模拟进行主动监控，模拟各应用系统关键或典型交易，定期探测交易是否能正常处理，以判断应用系统是否正在提供正常服务。[13]如，可以考虑在所有一级分行抽取部分重要网点部署探测脚本，定时发起模拟用户行为的仿真交易，记录整个交易流程（例如 ATM→综合前置→通用网关→主机)的响应时间，与相关交易的平均响应时间进行比较，如果超过平均交易响应时间，则进行报警，从而为关键业务交易的可用性问题提供优先的早期预警。[14]记录业务量、交易响应时间、交易成功率对系统状况进行监测属于被动监控行为。将主动监控和被动监控相结合，将能得到更好的检测效果。

（2）充分利用数据的特征对疑似异常进行排除。如，固定的平均交易成功率；分立的交易响应时间与业务量的反比关系等。

（3）利用更多的数据，结合银行对于用户体验和工作效率等方面的要求，给定合理的预测区间置信水平。

（4）综合考虑各种因素。如，春节前后；重大节日；气象条件等，结合已知数据和经验进行判断。

（5）充分利用已建立的模型进行合理的检测。

6 任务 3：基于扩展数据的方案优化

如可增加采集数据，可从客户行为特征即业务量时间序列，和 ATM 交易系统特征，包括交易成功量随业务量的变化，交易响应时间随业务量的变化等两个方面对检测方案进行优化。

6.1 客户行为特征

全面考虑异常情况。一年或几年的数据可用来分析各种节假日、突发事件、重大活动等情况下的客户通过 ATM 交易系统进行交易的行为特征，以更好进行判断。

6.2 ATM 交易系统特征

6.2.1 交易成功量随业务量的变化

利用更多的数据，结合银行对于用户体验和工作效率等方面的要求，确定平均交易成功率随业务量变化的更合理的置信水平和预测区间。

6.2.2 交易响应时间随业务量的变化

结合 ATM 系统实际运行的信息，包括各类业务的响应时间分布，客户发起各种类型交易的概率等，建立交易响应时间随业务量的变化更符合实际的模型。

利用更多的数据，结合银行对于用户体验和工作效率等方面的要求，确定平均交易响应时间随业务量变化更合理的置信水平和预测区间。

7 模型评价与展望

7.1 创新点和特色

（1）采用分而治之的方法对商业银行总行数据中心监控系统的 ATM 交易信息进行分析与建模，利用已知的业务量、交易成功率、交易响应时间三维时间序列数据，将 ATM 交易系统分为用户行为（业务量时间序列）和系统特性（交易响应时间、交易成功率）两部分分别建立模型进行分析，利用分析得到的结论进行 ATM 交易系统异常检测。

（2）利用排队模型对业务量和交易响应时间的关系进行了分析，建立了 ATM 交易系统多线程服务时间不等的 M/M/k 排队模型，解释了交易响应时间随业务量增大反而降低的原因，确定了交易响应时间异常的类型。

（3）引入新变量交易总响应时间和交易成功量，还原原始数据，降低数据分析的难度。

（4）提出了对业务量时间序列、交易成功率、交易响应时间进行异常检测的方法。

（5）提出了减少虚警误报的措施和基于扩展数据的方案优化方法。

8 参考文献

[1] 前瞻网，2015 年 Q3 我国 ATM 机数量为 84.08 万台，http://www.qianzhan.com/qzdata/detail/149/151221-dd169fac.html，2015,12,21.

[2] 王正友, 刘斯明. ATM 现金流量动态分析 [J]. 计算机辅助工程, 2006, 02): 71-3.

[3] 郭二凤. ATM 现金流预测的研究 [D]; 辽宁科技大学, 2015.

[4] 张明慧, 张尧禹. 人脸检测技术在 ATM 自动识别功能拓展系统中应用 [J]. 微型电脑应用, 2010, 04): 24-5+68.

[5] 朱志磊. 关于 ATM 机隔间内尾随检测算法的研究 [D]; 浙江工业大学, 2015.

[6] 封刚. 基于视频图像的银行 ATM 智能视频检测系统的研究和实现 [D]; 华南理工大学, 2012.

[7] 陈琼. 基于计算机视觉的 ATM 操作的异常行为检测 [D]; 西安电子科技大学, 2012.

[8] 王文龙. 应用于 ATM 监控的异常人脸检测方法研究 [D]; 沈阳航空航天大学, 2013.

[9] 郭思郁. 用于 ATM 机的遮挡人脸检测算法研究 [D]; 南京理工大学, 2013.

[10] 黄征宇. 用于 ATM 机遮挡人脸检测的模糊级联分类器和 ORB 算法的研究 [D]; 中南大学, 2013.

[11] 危定邦. 面向 ATM 机取款人的脸部异常事件检测系统设计 [J]. 计算机与网络, 2013, 07): 72-4.

[12] 陆传赉编著. 排队论 第 2 版 [M]. 北京：北京邮电大学出版社, 2009.

[13] 徐泽中. 数据集中模式下应用监控通用指标探析与实现 [J]. 中国金融电脑, 2009, 11): 61-5.

[14] 许彦青. 数据大集中模式下的应用监控分析 [J]. 中国金融电脑, 2011, 4): 61-3.

9 附录

附件 1：Model1_1.m

```
% minnum = 14400;
% daynum = 10;
minnum = 33102;
daynum = 23;
[num,txt]=xlsread('Apr.xls');
i = 1;
start = 1;
a=[];
twomaxes = [];
while (i <= minnum+1)
    if (i==minnum)
        a=[a;start,i];
        break;
    end
    if (~isequal(txt(i+1,1),txt(start+1,1)))
%           如果不符合就这一天统计完,并 do something
        a=[a;start,i-1];
        start = i;
    end
```

```
    i = i+1;
end
for i = 1:daynum
    maxsign1 = 432;
    maxvalue1 = num(a(i,1)+maxsign1-2,4)+num(a(i,1)+maxsign1-1,4)+num(a(i,1)+maxsign1,4)+num(a(i,1)+maxsign1+1,4)+num(a(i,1)+maxsign1+2,4);
    maxstart1 = 432;
    for j = 1:288
        tmpvalue = num(a(i,1)+maxstart1+j-2,4)+num(a(i,1)+maxstart1+j-1,4)+num(a(i,1)+maxstart1+j,4)+num(a(i,1)+maxstart1+1+j,4)+num(a(i,1)+maxstart1+2+j,4);
        if (tmpvalue > maxvalue1)
            maxsign1 = maxstart1+j;
            maxvalue1 = tmpvalue;
        end
    end
    maxsign2 = 720;
    maxvalue2 = num(a(i,1)+maxsign2-2,4)+num(a(i,1)+maxsign2-1,4)+num(a(i,1)+maxsign2,4)+num(a(i,1)+maxsign2+1,4)+num(a(i,1)+maxsign2+2,4);
    maxstart2 = 720;
    for j = 1:288
        tmpvalue = num(a(i,1)+maxstart2+j-2,4)+num(a(i,1)+maxstart2+j-1,4)+num(a(i,1)+maxstart2+j,4)+num(a(i,1)+maxstart2+1+j,4)+num(a(i,1)+maxstart2+2+j,4);
        if (tmpvalue > maxvalue2)
            maxsign2 = maxstart2+j;
            maxvalue2 = tmpvalue;
        end
    end
    twomaxes = [twomaxes;maxsign1/(a(i,2)-a(i,1)+1),maxsign2/(a(i,2)-a(i,1)+1)];
end
for i = 21:23
    left = twomaxes(i,1);
    right = twomaxes(i,2);
    x=[1:(a(i,2)-a(i,1)+1)]'/(a(i,2)-a(i,1)+1);
    y=num(a(i,1):a(i,2),4);
%     p=fittype('e*exp(f*(x-0.4)*(x-0.4))+g*exp(h*(x-0.63)*(x-0.63))','independent','x');
    p = fittype('e*exp(f*(t-left)*(t-left))+g*exp(h*(t-right)*(t-right))','independent','t','coefficients',{'e','f','g','h'},'problem',{'left','right'});
    f=fit(x,y,p,'problem',{left, right},'StartPoint',[1000,-200,1000,-20])
    plot(f,x,y);
%     scatter(x,y,25,'blue.')
%     xlabel('一天内的归一化交易时间')
%     ylabel('交易量[笔]')
%     title('4 月 6 日的交易量拟合结果')
%     box on
```

```
%       saveas(gcf,'1-2.png')
%       beta0 = [1000,0.2,1000,0.2];
%       fun = inline(betas(1)*exp(betas(2)*(x-0.4)*(x-0.4))+betas(3)*exp(betas(4)*(x-0.63)*(x-0.63)),'betas','x')
%       [betas,R,J]=nlinfit(x,y,fun,beta0);
end
```

附件 2：Model1_2.m

```
[num] = xlsread('paras.xls');
a(1:91,1:3) = 0.0;
for i = 1:91
a(i,1) = num(i,1)+num(i,3);
a(i,2) = num(i,2)+num(i,4);
a(i,3) = num(i,6)-num(i,5);
end
% scatter3(a(:,1),a(:,2),a(:,3));
% scatter(a(:,1),a(:,3),15,'blue*');
% box on
% xlabel('两峰高度之和')
% ylabel('峰间距')
% saveas(gcf,'2-6.png')
%
% for i = 1:91
%       if (i >= 17)
%             if (mod(i,7) == 6 || mod(i,7)==0)
%                   scatter(a(i,1),a(i,3),15,'red*');
%                   hold on
%             else
%                   scatter(a(i,1),a(i,3),15,'blue*');
%                   hold on
%             end
%       else
%             scatter(a(i,1),a(i,3),15, 'black*');
%             hold on
%       end
% end
% box on
% xlabel('两峰高度之和')
% ylabel('峰间距')
% saveas(gcf,'2-7.png')
firsttype = [];
for i = 1:90
    if(a(i,1)< 2400 && a(i,3) < 0.2)
          if (i<=9)
```

```
            firsttype = [firsttype;i,1,i+21];
        else
            if (i<=37)
                firsttype = [firsttype;i,2,i-9];
            else
                if (i <= 68)
                    firsttype = [firsttype;i,3,i-37];
                else
                    firsttype = [firsttype;i,4,i-68];
                end
            end
        end
    end
end
secondtype = [];
for i = 1:90
    if(a(i,1)> 2400)
        if (i<=9)
            secondtype = [secondtype;i,1,i+21];
        else
            if (i<=37)
                secondtype = [secondtype;i,2,i-9];
            else
                if (i <= 68)
                    secondtype = [secondtype;i,3,i-37];
                else
                    secondtype= [secondtype;i,4,i-68];
                end
            end
        end
    end
end
 firsttype(:,3)
 secondtype(:,3)
% %去掉 1 月 19 到 2 月 7 号这一段过年的点，重新做聚类，看看能不能得到更加细致的分类信息
% aplus = [];
% for i = 1:90
%         if (i <= 8 || i >= 29)
%             aplus = [aplus;a(i,:)];
%             if (mod(i,7) == 4 || mod(i,7)==5)
%                 scatter(a(i,1),a(i,3),15,'red*');
%                 hold on
%             else
%                 scatter(a(i,1),a(i,3),15,'blue*');
```

```
%                     hold on
%                 end
%             else
%                 scatter(a(i,1),a(i,3),15,'black*');
%             end
% end
% xlabel('两峰高度之和')
% ylabel('峰间距')
% title('聚类后的以日期作为散点的交易量特征参数的散点图分布')
% box on
% saveas(gcf,'3.png')
%
% % scatter3(aplus(:,1),aplus(:,2),aplus(:,3));
% % scatter(aplus(:,1),aplus(:,3),15,'red');
%
% % zero(1:70,1:1) = 0.0;
% % scatter(aplus(:,1),zero(:,1));
```

附件 3：Model1_3.m

```
minnum = 12954;
daynum = 9;
[num,txt]=xlsread('Jan.xls');
start = 1;
i = 1;
count = 0;
a=[];
while (i <= minnum+1)
    if (i==minnum)
        count = count + num(i,1);
        a=[a;count];
        count = 0;
        break;
    end
    if (~isequal(txt(i+1,1),txt(start+1,1)))
%           如果不符合就这一天统计完,并 do something
        a=[a;count];
        count = num(i,1);
        start = i;
    else
        count = count + num(i,1);
    end
    i = i+1;
end
```

```
minnum = 40320;
daynum = 28;
[num,txt]=xlsread('Feb.xls');
start = 1;
i = 1;
count = 0;
while (i <= minnum+1)
    if (i==minnum)
        count = count + num(i,1);
        a=[a;count];
        count = 0;
        break;
    end
    if (~isequal(txt(i+1,1),txt(start+1,1)))
%           如果不符合就这一天统计完,并 do something
        a=[a;count];
        count = num(i,1);
        start = i;
    else
        count = count + num(i,1);
    end
    i = i+1;
end
minnum = 44637;
daynum = 31;
[num,txt]=xlsread('Mar.xls');
start = 1;
i = 1;
count = 0;
while (i <= minnum+1)
    if (i==minnum)
        count = count + num(i,1);
        a=[a;count];
        count = 0;
        break;
    end
    if (~isequal(txt(i+1,1),txt(start+1,1)))
%           如果不符合就这一天统计完,并 do something
        a=[a;count];
        count = num(i,1);
        start = i;
    else
        count = count + num(i,1);
    end
```

```
    i = i+1;
end
minnum = 33102;
daynum = 23;
[num,txt]=xlsread('Apr.xls');
start = 1;
i = 1;
count = 0;
while (i <= minnum+1)
    if (i==minnum)
        count = count + num(i,1);
        a=[a;count];
        count = 0;
        break;
    end
    if (~isequal(txt(i+1,1),txt(start+1,1)))
%           如果不符合就这一天统计完,并 do something
        a=[a;count];
        count = num(i,1);
        start = i;
    else
        count = count + num(i,1);
    end
    i = i+1;
end
for i = 1:90
    if (i >= 17)
        if (mod(i,7) == 6 || mod(i,7)==7)
            scatter(a(i,1),rand(),15,'red*');
            hold on
        else
            scatter(a(i,1),rand(),15,'blue*');
            hold on
        end
    else
        scatter(a(i,1),rand(),15,'black*');
        hold on
    end
end
xlabel('日交易总量[笔]')
ylabel('随机数')
box on
saveas(gcf,'4-2.png')
% plot(a(:,1))
```

附件 4：Model1_4.m

```
a(1:91,1:1440) = 0.0;
num = [];
txt = [];
pred = 1; %当前这一天开始的位置
start = 1; %下一天开始的位置
sign = 1;   %现在读到的位置
for i = 1:91
    if (i == 1)
        [num,txt]=xlsread('Jan.xls');
        minnum = 12954;
        start = 1;
        sign = 1;
    end
    if (i == 10)
        minnum = 40320;
        [num,txt]=xlsread('Feb.xls');
        start = 1;
        sign = 1;
    end
    if (i == 38)
        minnum = 44637;
        [num,txt]=xlsread('Mar.xls');
        start = 1;
        sign = 1;
    end
    if (i == 69)
        minnum = 33102;
        [num,txt]=xlsread('Apr.xls');
        start = 1;
        sign = 1;
    end
    sign = start;
    pred = start;
    for j = 1:1440
        if (start <= minnum && isequal(txt(pred+1,1),txt(start+1,1))) %还在当前这一天则需要向前推
            start = start + 1;
        end
        if (sign > minnum)
            a(i,j) = num(minnum,1);
        else
            a(i,j) = num(sign,1);
```

```
        end
        sign = sign+1;
    end
end
pingshi(1:1440,1) = 0.0; %平时的数据
nianqian(1:1440,1) = 0.0; %年前的数据（1 月 19 到 1 月 27）
nianhou(1:1440,1) = 0.0; %年后的数据（1 月 28 到 2 月 7）
for i = 1:90
    if (i <= 5)
        for j = 1:1440
            nianqian(j,1) = nianqian(j,1) + [a(i,1+mod(j-3+1440,1440))+a(i,1+mod(j-2+1440,1440))+a(i,1+mod(j-1+1440,1440))+a(i,1+mod(j,1440))+a(i,1+mod(j+1,1440))]/5;
        end
    else
        if (i >= 6 && i <= 16)
            for j = 1:1440
                nianhou(j,1) = nianhou(j,1) + [a(i,1+mod(j-3+1440,1440))+a(i,1+mod(j-2+1440,1440))+a(i,1+mod(j-1+1440,1440))+a(i,1+mod(j,1440))+a(i,1+mod(j+1,1440))]/5;
            end
        else
            for j = 1:1440
                pingshi(j,1) = pingshi(j,1) + [a(i,1+mod(j-3+1440,1440))+a(i,1+mod(j-2+1440,1440))+a(i,1+mod(j-1+1440,1440))+a(i,1+mod(j,1440))+a(i,1+mod(j+1,1440))]/5;
            end
        end
    end
end
x = [1:1440]'/1440;
nianqian = nianqian/5;
nianhou = nianhou/11;
pingshi = pingshi/75;
apingshi = [a(17:91,:)];
apingshiba(1:75,1:1440) = 0.0;
for i = 1:75
    for j = 1:1440
        apingshiba(i,j) = apingshiba(i,j) + [apingshi(i,1+mod(j-3+1440,1440))+apingshi(i,1+mod(j-2+1440,1440))+apingshi(i,1+mod(j-1+1440,1440))+apingshi(i,1+mod(j,1440))+apingshi(i,1+mod(j+1,1440))]/5;

    end
end
% for i = 1:75
%     plot(apingshiba(i,:))
%     hold on
% end
```

```
pingshimin(1:1440,1)=0.0;
for i = 1:1440
      pingshimin(i,1) = min(apingshiba(:,i));
end
pingshiscore(1:75,1:1440) = 0.0;
for i = 1:75
    for j = 1:1440
        if (apingshiba(i,j) >= pingshi(j,1))
            pingshiscore(i,j) = 100;
        else
            pingshiscore(i,j) = (apingshiba(i,j)-pingshimin(j,1))*100/(pingshi(j,1)-pingshimin(j,1));
        end
    end
end

% delnondel(1:99,1:99) = 0.0;
% for k = 1:99
%     for l = 1:99
%         deltascore = k;
%         score = l;
%         count = 0;
%         for i = 1:75
%             for j = 1:1440
%                 if (pingshiscore(i,j)<score || pingshiscore(i,j)-pingshiscore(i,1+mod(j+1339-5,1440)) < -
deltascore)
%                     count = count+1;
%                 end
%             end
%         end
%         delnondel(k,l) = count / 75/1440;
%     end
% end

deltascore = 65;
score = 22;
 count = 0;
 for i = 1:75
     scatter([1:1440]'/1440,apingshi(i,:)','blue.')
     hold on
 end
 for i = 1:75
     for j = 1:1440
         if (pingshiscore(i,j)<score || pingshiscore(i,j)-pingshiscore(i,1+mod(j+1339-5,1440)) < -deltascore)
             count = count+1;
```

```
                scatter(j/1440,apingshi(i,j),'red.')
                hold on
            else
                scatter(j/1440,apingshi(i,j),'blue.')
                hold on
             end
        end
 end
 xlabel('一天内的归一化交易时间')
ylabel('交易量[笔]')
title('95%正常点下的异常点分布')
box on
saveas(gcf,'8-2.png')
ita = count/75/1440


% plot(x,nianqian,'m');
% hold on
% plot(x,nianhou,'blue');
% hold on
% plot(x,pingshi,'black');
% xlabel('一天内的归一化交易时间')
% ylabel('交易量[笔]')
% box on
% title('各类日期的交易量随时间的变化关系')
% saveas(gcf,'5.png')

nianqianbian = [nianqian(217:1440,1);nianqian(1:216)];
y = nianqianbian;
% p = fittype ('e*exp(f*(t-0.26)*(t-0.26))+g*exp(h*(t-0.48)*(t-0.48))','independent','t','coefficients',{'e','f','g','h'});
% f=fit(x,y,p,'StartPoint',[1000,-200,1000,-20])
% plot(f,x,y,'m');
% hold on
% nianhoubian = [nianhou(217:1440,1);nianhou(1:216)];
% y = nianhoubian;
% p = fittype ('e*exp(f*(t-0.275)*(t-0.275))+g*exp(h*(t-0.47)*(t-0.47))','independent','t','coefficients',{'e','f','g','h'});
% f=fit(x,y,p,'StartPoint',[1000,-200,1000,-20])
% plot(f,x,y,'blue');
% hold on
% pingshibian = [pingshi(217:1440,1);pingshi(1:216)];
% y = pingshibian;
% p = fittype ('e*exp(f*(t-0.255)*(t-0.255))+g*exp(h*(t-0.51)*(t-0.51))','independent','t','coefficients',{'e','f','g','h'});
% f=fit(x,y,p,'StartPoint',[1000,-200,1000,-20])
% plot(f,x,y,'black');
```

```
% hold on
% xlabel('一天内的归一化交易时间')
% ylabel('交易量[笔]')
% title('各类日期的交易量随时间变化关系的双 Guass“标准模型”近似')
% box on
% saveas(gcf,'6-1.png')




%
% for i = 1:91
%     if (i >= 17)
%         tmpbian = [a(i,217:1440),a(i,1:216)];
%        tmpbian = [a(i,1:1440)];
%         if (mod(i,7) == 6 || mod(i,7)==0)
%             scatter(x, tmpbian,15,'red.');
%         else
%             scatter(x, tmpbian,15,'blue.');
%         end
%         hold on
%     end
% end
% xlabel('一天内的归一化交易时间')
% ylabel('交易量[笔]')
% title('非春节前后的所有数据点')
% box on
% saveas(gcf,'7-1.png')

% for i = 1:91
%     if (i >= 17)
%         tmpbian = [a(i,217:1440),a(i,1:216)];
%         scatter(x, tmpbian,'blue.');
%         hold on
%     end
% end
%
% pingshibian = [pingshi(217:1440,1);pingshi(1:216)];
% y = pingshibian;
%     p     =     fittype     ('e*exp(f*(t-0.255)*(t-0.255))+g*exp(h*(t-0.51)*(t-0.51))-100*exp(k*(t-0.335)*(t-
0.335))','independent','t','coefficients',{'e','f','g','h','k'});
% f=fit(x,y,p,'StartPoint',[1000,-200,1000,-20,-20])
% plot(f,x,y,'black.');
% hold on
```

```
% xlabel('一天内的归一化交易时间')
% ylabel('交易量[笔]')
% % title('平时所有日期数据的散点图分布')
% box on
% saveas(gcf,'7-2.png')
```

附件 5：paras.xls

date	m_1	σ_1	m_2	σ_2	μ_1	μ_2
0123	1087	-81.36	1912	-19.8	0.4493	0.6563
0124	1131	-99.87	2187	-17.5	0.4472	0.6486
0125	1361	-193.7	2577	-16.62	0.4306	0.6375
0126	1264	-231.9	2619	-15.48	0.4299	0.6118
0127	1561	-182	1566	-26.02	0.4153	0.5979
0128	177.5	-775.4	616.5	-15.67	0.4339	0.5841
0129	398.5	-311.3	549.4	-19.41	0.4249	0.614
0130	462.1	-240.6	682.7	-21.5	0.4343	0.6352
0131	423.3	-514.4	801.1	-19.28	0.4281	0.6025
0201	480.3	-208.4	862.8	-22.48	0.4451	0.6424
0202	775.1	-185.1	1079	-26.99	0.4382	0.6507
0203	688.6	-117.6	1161	-23.55	0.4507	0.641
0204	465.8	-393.6	1349	-16.89	0.4431	0.6007
0205	524.3	-451.2	1275	-16.89	0.4382	0.6139
0206	769.5	-138.1	1304	-21.85	0.4389	0.6389
0207	447.9	-371.7	1212	-16.48	0.4319	0.5993
0208	746.7	-88.11	919	-34.09	0.4333	0.6813
0209	696.2	-104.3	1231	-24.07	0.4403	0.6521
0210	660.7	-163.7	1329	-20.45	0.4368	0.6417
0211	734.3	-244.7	1277	-19.9	0.4264	0.6333
0212	450	-362.5	1238	-14.09	0.4313	0.5813
0213	596	-281.2	1379	-15.47	0.4222	0.609
0214	833.2	-154.4	1291	-19.57	0.4104	0.6444
0215	955.4	-88.62	1359	-25.22	0.4306	0.6757
0216	527.3	-98.84	1230	-17.07	0.4382	0.6493
0217	648.9	-58.87	1120	-21.18	0.4444	0.6604
0218	969.6	-81.21	1151	-30.1	0.4313	0.6917
0219	822.9	-111.7	1151	-23.51	0.4333	0.6708
0220	634.9	-151	1209	-19.4	0.4222	0.6444
0221	835.5	-99.29	1096	-24.79	0.4188	0.6528
0222	807	-75.14	1038	-32.68	0.4278	0.6882
0223	683.1	-154.7	1172	-19.35	0.4042	0.6486
0224	628.3	-123.7	1234	-18.29	0.4236	0.6514

0225	968.5	-76.17	1169	-28.46	0.4278	0.6986
0226	384.9	-505.8	1147	-13.95	0.4292	0.5868
0227	1032	-95.68	1380	-25.71	0.4264	0.6715
0228	939.8	-76.85	1353	-23.3	0.4368	0.6764
0301	449.9702338	-246.2016027	1224.886388	-13.81430754	0.417361111	0.597222222
0302	944.5232878	-87.27138013	1245.640632	-25.47120891	0.408333333	0.684722222
0303	952.6701588	-70.38063909	1221.44542	-29.29564819	0.422916667	0.694444444
0304	732.7720853	-128.2435178	1082.782747	-18.60887568	0.422916667	0.659722222
0305	766.3369333	-180.6332964	1146.624425	-18.62977141	0.414583333	0.649305556
0306	761.6178306	-136.9806151	1198.705222	-18.69871813	0.403472222	0.65
0307	905.7984665	-81.37338586	1119.352475	-26.50320725	0.406944444	0.693055556
0308	801.0461014	-83.74664161	1124.429969	-21.218768	0.379166667	0.695138889
0309	614.2170386	-122.737325	1068.876293	-15.99063223	0.420138889	0.652083333
0310	909.3063989	-100.2534649	1287.24472	-23.52687324	0.409027778	0.680555556
0311	985.9814915	-82.9752738	1090.889847	-28.62230685	0.425	0.69375
0312	565.9421655	-242.9680534	921.4848819	-14.88243891	0.420138889	0.590972222
0313	878.4618359	-106.0252082	1122.101813	-25.38426604	0.414583333	0.672916667
0314	1025.711073	-71.40502607	1251.433656	-29.75725537	0.420833333	0.696527778
0315	790.7950691	-78.93870194	1242.569958	-20.65161427	0.432638889	0.674305556
0316	992.6096283	-65.79377638	1206.276262	-29.17109723	0.429861111	0.691666667
0317	757.8903912	-101.3180982	1161.900403	-19.47839302	0.41875	0.660416667
0318	604.3162324	-173.4059474	1017.840668	-15.71126568	0.419444444	0.642361111
0319	910.6699748	-74.37063005	1063.18343	-33.65336162	0.42767733	0.694714882
0320	722.437432	-126.9475274	1152.572674	-20.9737989	0.415277778	0.657638889
0321	878.3963439	-96.20178021	1219.611696	-21.484094	0.415972222	0.667361111
0322	474.8817705	-239.6500021	874.3545245	-14.1152006	0.396527778	0.585416667
0323	822.4397672	-90.48904434	1150.284408	-25.53391206	0.411111111	0.678472222
0324	214.8660422	-416.1137833	1092.881255	-11.95500917	0.420833333	0.570833333
0325	896.9603849	-75.36162457	1117.996592	-26.41570377	0.429166667	0.694444444
0326	650.5909758	-116.9024769	1005.423252	-17.28030489	0.429861111	0.660416667
0327	951.1853069	-94.21056109	1227.095942	-23.88881828	0.416666667	0.677777778
0328	877.1863061	-106.4660068	1122.443074	-22.26619618	0.402777778	0.671527778
0329	827.7211628	-95.4586013	1114.761878	-22.68697483	0.406944444	0.678472222
0330	413.9617519	-105.5308087	886.0726077	-12.70815703	0.433634468	0.605976372
0331	977.9961372	-82.85847846	1326.461511	-24.74140726	0.402777778	0.697916667
0401	963.4	-65.27	1280	-27.67	0.4306	0.6965
0402	948.5	-84.51	1032	-25.1	0.4104	0.6965
0403	924.9	-99.19	988	-24.14	0.3979	0.6993
0404	807.5	-119.6	1046	-24.85	0.4042	0.6708
0405	958.6	-101.7	1310	-23.2	0.4139	0.6729
0406	724.4	-71.16	1063	-29.67	0.4201	0.6965
0407	591.9	-155.6	1120	-15.47	0.4118	0.6458
0408	940.2	-81.58	986.3	-28.65	0.4153	0.6951
0409	811.4	-87.29	1028	-30.17	0.4278	0.6847

0410	556.2	-267.9	1050	-15.49	0.3868	0.5729
0411	815.6	-149	1300	-17.09	0.3889	0.6514
0412	964.1	-82.54	1196	-26.16	0.4104	0.6937
0413	844.5	-106.2	1110	-20.9	0.3882	0.6806
0414	909.1	-86.33	1216	-25.87	0.4125	0.6937
0415	921.6	-126.3	1092	-19.51	0.3937	0.6813
0416	1084	-85.07	1075	-35.72	0.4128	0.6857
0417	943.8	-119.2	1294	-21.42	0.409	0.6764
0418	1011	-101.6	1184	-22.05	0.384	0.6951
0419	978.2	-88.29	1113	-26.01	0.4014	0.6958
0420	538.7	-128.3	1056	-12.96	0.4236	0.6549
0421	949.6	-94.72	1116	-23.32	0.3917	0.6875
0422	942.3	-79.13	1012	-28.07	0.4236	0.6958