Dydy 打车派单模式探究

小组作业报告

陈凯昊 黄筝 朱彦頔

薛修梅 高鸣 杨文龙

2017 年 11 月 25 日

目录

一、 问题重述

Dydy 是一家涵盖出租车和网约车的出行平台，出租车（A 类）与网约车（B 类）的数量比例为 3:1。该平台运用的派单模式为：以订单呼叫地 1 公里为半径圈定派单司机圈，如果圈内只有一名等待司机，则将此单派给他；如果有多名司机，B 类司机优先被派单，并按照距离最短的原则派车。当订单派给 A 类司机时，派单时间计入 TA；派给 B 类司机时，派单时间计入 TB。连续 22 个工作日 12:00-13:00 观察 5 分钟订单到达数量（Order），TA.sp 为 A 类司机成功接通时间的样本，TB.sp 为 B 类司机成功接通时间的样本。希望通过数据分析理解，如果一位司机最长忍耐时间是 10 分钟，前面有 30 张单子在等待接受服务：

（1）那么他需要等待 10 分钟以上的可能性有多大？

（2）如果 10 分钟是一名司机的合理等待时间，而合理等待时间远小于实际等待时间，司机则会出现抵触情绪， 抵触情绪导致司机拒绝使用派单模式， 而这类的问题则应当受到经营管理方面的高度重视， 请问在以下 4 种原因（单选题）分析中，合理等待时间与等待时间之间的逆差这个原因是否应当受到重视？

用什么方法，你的结论基于怎样的假设？

原因序号	司机对软件服务平台不满意的主要原因	非常不满意	不满意	一般	满意	非常满意
1	合理等待时间与感知等待时间之间的逆差	984	143	49	16	8
2	派车模式的激励政策	113	53	30	20	10
3	消费者刷卡便利性	320	250	33	18	6
4	派车地点合理性	274	90	84	4	1

二、 问题假设及模型假设

（1）每名司机的合理等待派单时间是固定的；

（2）每张订单 1 公里内没有空闲司机的可能性忽略不计；

（3）假设出租车司机可以实时了解到目前前面排队的订单数量来决定是否进入派单模式，已经进入派单模式排队的司机不会选择放弃排队而离开；

（4）假设区域形状为矩形，A 类司机与 B 类司机在区域中均匀分布，每一个订单呼叫地在区域中是随机出现的；

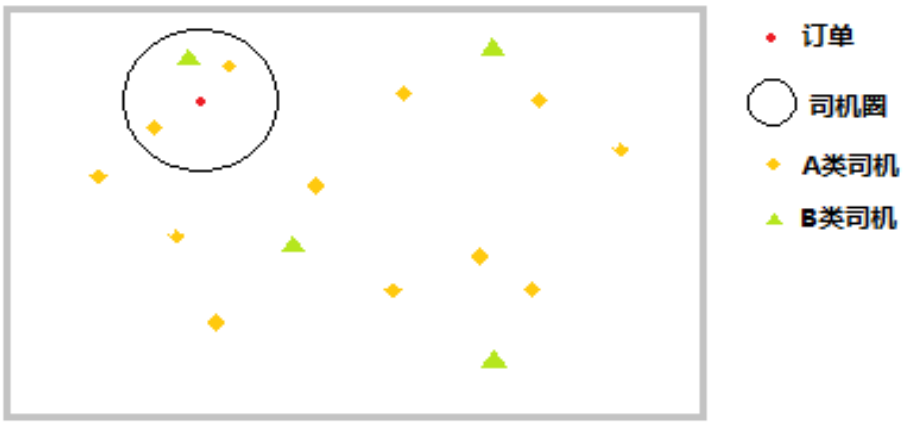
（5）每类司机派单成功时间的分布只与司机类型（出租车或网约车）有关，不考虑信号传输所需时间；

（6）订单的发出是源源不断的，即不会出现在某一时间全区域内没有订单的情况，通过拟合的分布进行模拟，可以在大概率的情况下保证这一假设条件。

三、 解题思路

1、问题一：建立模型

（1）在 a x a 区域上建立坐标系： 对于每一次模拟，通过生成随机数确定每一订单出现的坐标（x，y），从而可以确定以其为圆心，1 公里为半径的司机圈；



同样地，可以通过生成随机数确定 n 名司机所在的位置，其中 3/4 为 A 类司机，1/4 为 B 类司机。此时对于每一订单所确定的司机圈，圈内的司机种类和个数都是确定、可知的。

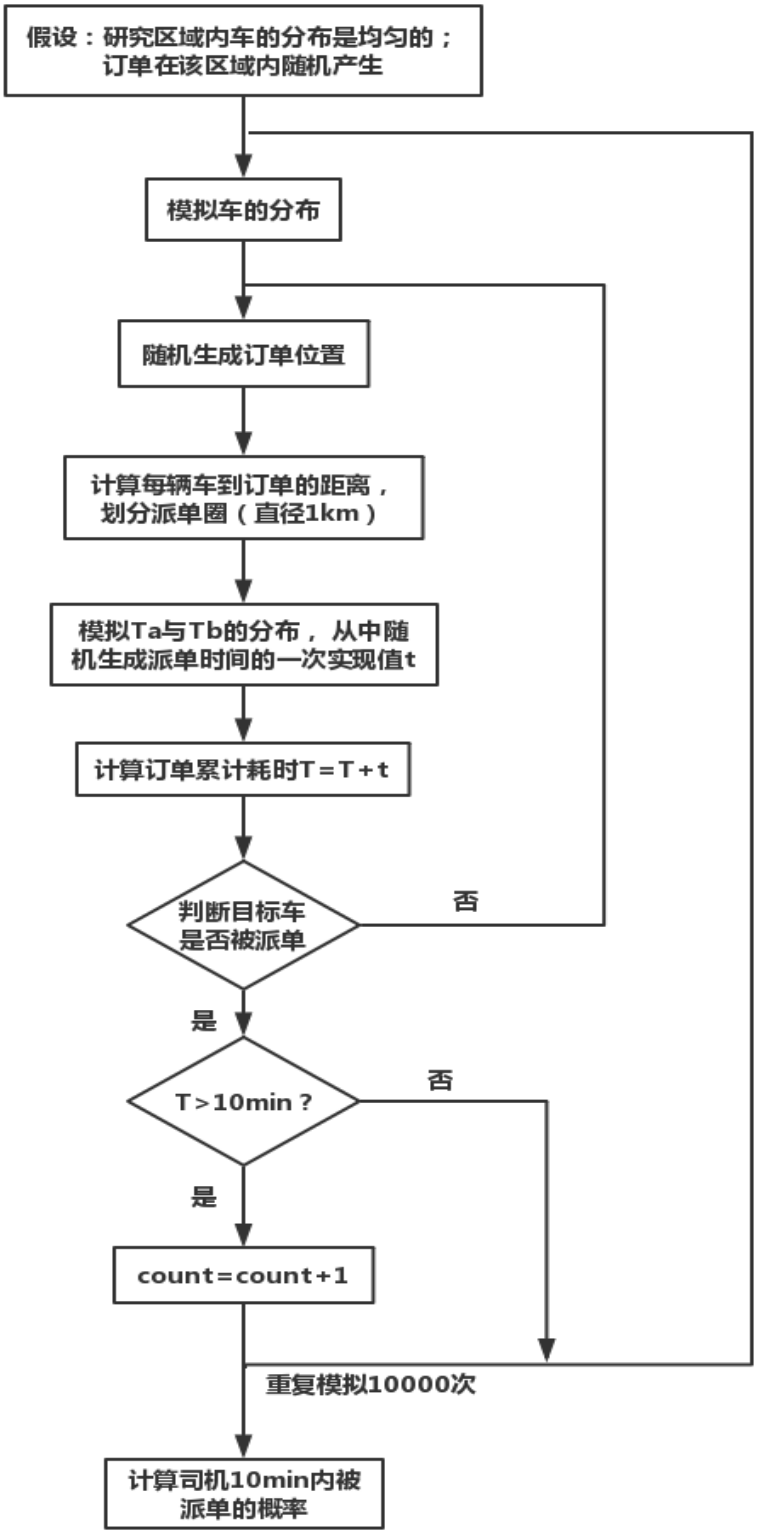
（2）通过 TA、TB 与 Order 的样本，拟合其分布，通过分布进行模拟，证明在系统内已有 30 张订单的情况下，10min 之内出现系统内订单已经被处理完的情况概率极小，可以忽略不计，即认为订单是源源不断的。

（3）记当前时刻为 0，对于每一订单，都可以通过（1）的方式确定司机圈内的司机情况。进一步，由于处理订单的时间只与该司机圈内的司机情况有关，可以从拟合的 TA 和 TB 的分布中生成并确定每一订单的时间。在这里我们编写函数，对于每一订单，通过输入该订单的坐标，可以输出处理该订单所需的时间和对于感兴趣司机是否获得该订单。

（4）由于司机在该区域服从均匀分布，我们在 n 名司机中随机选取一名作为目标司机，确定其坐标。对于每一订单，如果该司机在司机圈内且恰好被选中，就停止循坏，不再考虑新的订单；否则继续下一订单，直到该司机被选中为止。

（5）至此，对于每一次模拟，我们都可以算出目标司机总的等待时间，如果该时间>10 分钟，则记为 1；否则记为零。模拟 N 次以后，用记为 1 的模拟总次数 M 除以模拟总次数 N，便可记 M/N 是司机需要等待 10 分钟以上的概率。

（6）流程图

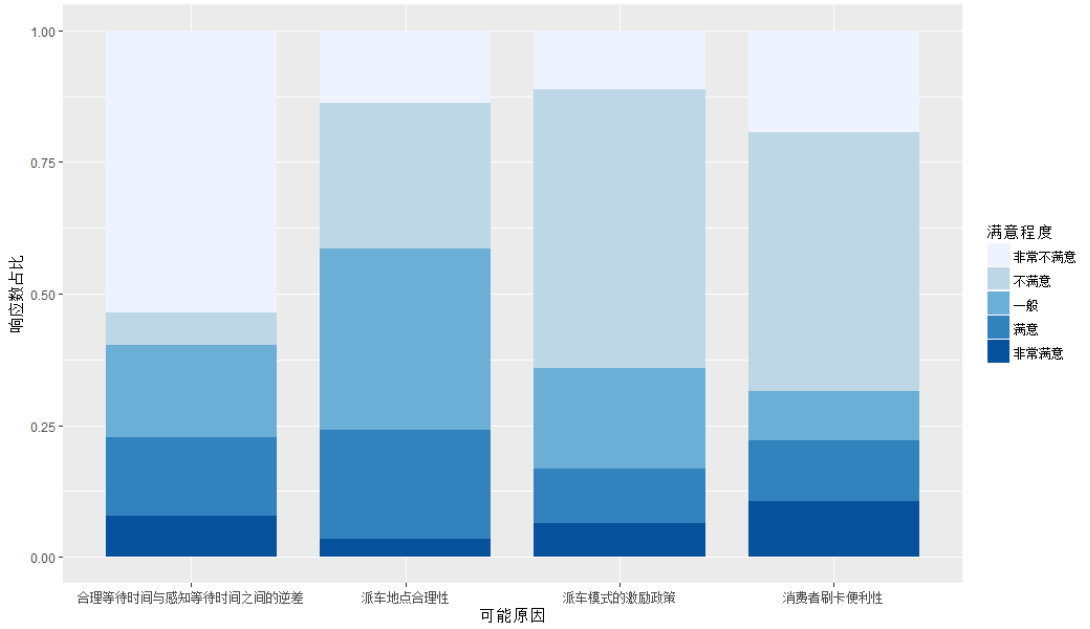


2、问题二：Ridit 检验

根据题意，司机对 Dydy 平台不满意主要有 4 个原因：合理等待时间与感知

等待时间之间的逆差、派车模式的激励政策、消费者刷卡便利性、派车地点合理性。对每一方面采用 5 级量表形式调查满意度。要寻找司机抵触情绪的主要来源，也就是从上述 4 个方面中寻找司机整体满意度最低的一个。

为此，将 4 个问题视为 4 个处理，而顺序尺度变量按满意度从低到高依次为非常不满意、不满意、一般、满意、非常满意 5 个顺序类。首先作出各可能原因中不同满意程度的百分比堆积柱状图（如下图），可见司机对于“合理等待时间与感知等待时间之间的逆差”选择“非常不满意”的比例远高于其他三项，可以初步判断这个原因引起了大多数司机的抵触情绪。但若考察（非常不满意+不满意）的比例，可见“合理等待时间与感知等待时间之间的逆差”对应比例在四者中并不是最高的，因此不能简单做出该可能原因最值得重视的结论。



为了做出更严谨的推断并能够量化推断犯错误的概率，以下对“司机对四个可能原因的感知（即满意度）是否相同”进行假设检验。该研究的问题可以转化为比较 4 个可能原因（视作 4 个不同处理）满意度的平均水平是否存在差异。并且不同满意度的响应数差别太大，不适合认为指定等距得分进行计算因此可以选择 Ridit 检验方法。

Ridit 检验的过程如下：

（1）将总体 2506 名司机的满意度作为参照组，计算各顺序类别的 Ridit 得分；

（2）按照参照组顺序类别的 Ridit 得分结构，将每种可能原因的满意度分组响应数转换为 Ridit 得分，计算平均 Ridit 及 95%的可信限；

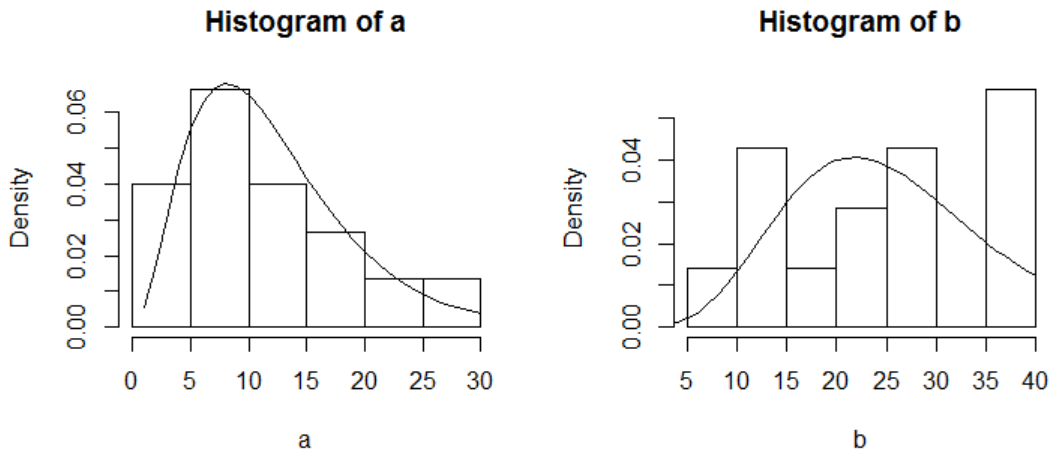
（3）参照组的平均 Ridit 得分为 0.5，利用变换后的 Ridit 得分进行各处理之间强弱的比较。

四、 建立模型及代码实现

1、问题一

（1）观察 Order、TA、TB 的数据分布情况。Order 数据为一个计数过程，分布呈现右偏趋势，尝试用 possion 分布、negative binomial 分布进行拟合，发现负二项分布拟合情况良好。TA 数据分布同样呈现右偏趋势，拟合为 Gamma 分布；TB 数据分布偏态不明显，尝试多种分布进行拟合后，最终选用负二项分布。最后对拟合的分布进行 KS 检验，检验的 p 值分别达到 0.8047、0.9754、0.915，拟合效果较好。

```
library(MASS)
fitdistr(a,densfun = 'gamma')
hist(a,freq = F)
lines(dgamma(x=1:30,shape = 3.07,rate = 0.2559))
fitdistr(b,densfun = 'negative binomial')
hist(b,freq = F,breaks = 5)
lines(dnbinom(x=1:40,size = 7.982,mu = 25.57))
ks.test(b,'pnbinom',size = 7.982,mu = 25.57)
ks.test(a,'pgamma',shape = 3.07,rate = 0.2559)

fitdistr(order,densfun = 'negative binomial')
ks.test(order,'pnbinom',size = 5.141744,mu = 79.636364)
hist(order,freq = F)
lines(dnbinom(x=1:200,size = 5.141744,mu = 79.636364))
```

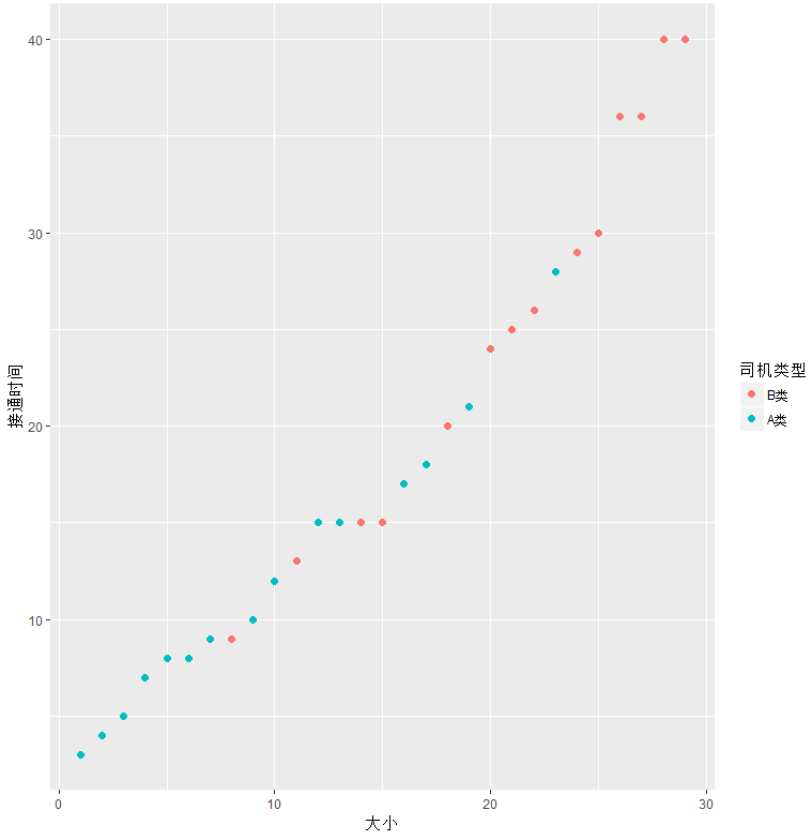



```
> ks.test(a,'pgamma',shape = 3.07,rate = 0.2559)

        One-sample Kolmogorov-Smirnov test

data:  a
D = 0.12391, p-value = 0.9754
alternative hypothesis: two-sided
```

```
> ks.test(b,'pnbinom',size = 7.982,mu = 25.57)

        One-sample Kolmogorov-Smirnov test

data:  b
D = 0.14901, p-value = 0.915
alternative hypothesis: two-sided
```

```
> ks.test(order,'pnbinom',size = 5.141744,mu = 79.636364)

        One-sample Kolmogorov-Smirnov test

data:  order
D = 0.13682, p-value = 0.8047
alternative hypothesis: two-sided
```

✧ **对假设(6)进行简单验证：**



检验思路：由拟合的 TA 与 TB 分布，我们可以随机获得其实现值，同时我们可以发现 TB>TA，故我们简化地认为订单都是与 A 型车成功连接。只需要证明，在处理完 30 个订单的时间内，新产生的订单数将大概率大于 30 个即可。

```
#证明订单源源不断
for(i in 1:10000){
  T=0
  neworder=0
  T=sum(rgamma(30,shape = 3.07,rate = 0.2559))
  neworder=T/300*rnbinom(1,size = 5.141744,mu = 79.636364)
  if(neworder>30){count1=count1+1}
  if(neworder>60){count2=count2+1}
}
p1=count1/10000
p2=count2/10000
```

拟合 10000 次，可以发现，在处理 30 单的时间中，新生成的订单数高于 30 个的概率为 0.9683，而订单高于 60 个的概率为 0.7718。而且，在拟合中，我们仅仅使用了 TA 的分布，假设所有订单都是由 A 类车成功接通。而实际中还会有一部分为 B 类车接通，这部分的时间还应大于 A 类车。由此，我们有充分的理由假设：在前面有 30 个订单待处理的情况下，订单几乎不可能被处理完。

（2）网上搜索打车平台相关资料：


网易财经 网易首页 应用 薄荷直播 网易考拉 严选

北京市滴滴数据首次披露：网约车非拥堵直接原因

2016-11-03 15:10:00 来源: 21世纪经济报道(广州) 举报

4 易信 微信 QQ空间 微博 更多

（原标题：北京市滴滴数据首次披露：网约车非拥堵直接原因）

近日，北京交通大学交通系统科学与工程研究院闫学东教授课题组发布了一份完成于10月31日的报告，报告利用滴滴出行大数据，得出结论：北京的道路拥堵主要因为出行需求导致，网约车不是造成北京道路拥堵的直接原因。

北京市注册司机超150万人

滴滴出行在北京市的运营数据首次详细披露。报告称，目前，滴滴出行在北京市域的日均订单量已过100万单，总注册司机数已超过150万人。同时，滴滴司机中，每日在线时长不超过2小时的占比接近50%。

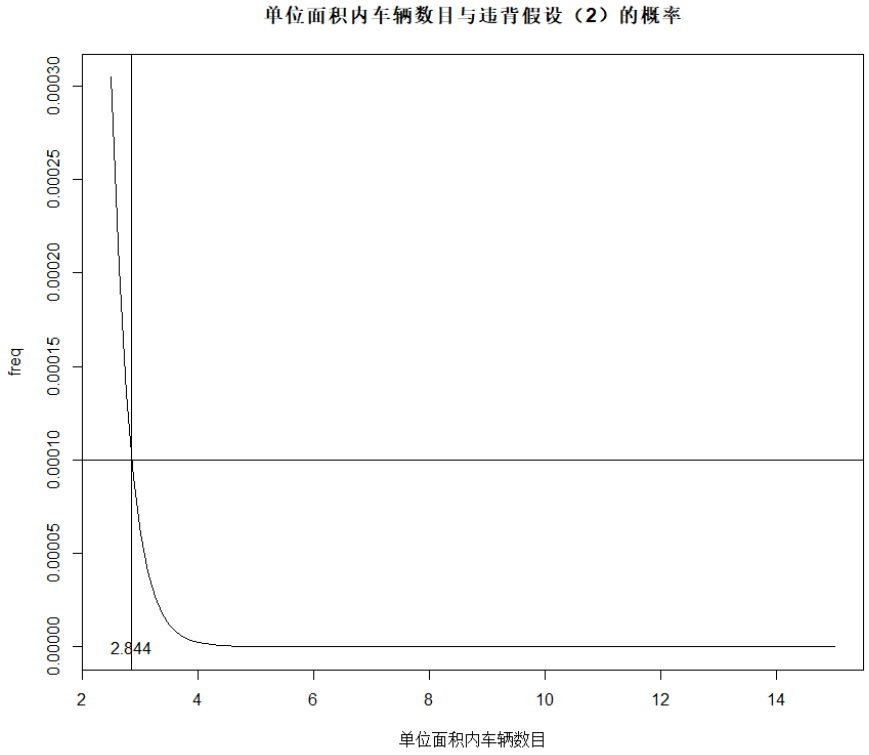
从今年1月以来，两大打车软件烧钱换市场带来了惊人的扩张效益，打车软件市场目前已被两大公司垄断，快的打车和嘀嘀打车占据累计用户市场份额的97%以上，其中快的打车排第一位，市场份额为57.6%，嘀嘀打车用户排在第二位，市场份额为39.8%，排在第三的为大黄蜂，市场份额为1.9%，其他打车软件仅占市场份额的0.7%。目前，快的打车已覆盖全国261个城市，每天完成623万个订单。嘀嘀打车覆盖178座城市，每天完成521万次订单出行服务。

Source:

网易财经，http://money.163.com/16/1103/16/C4V820HA002580S6.html

http://sh.qq.com/a/20140519/005713.htm

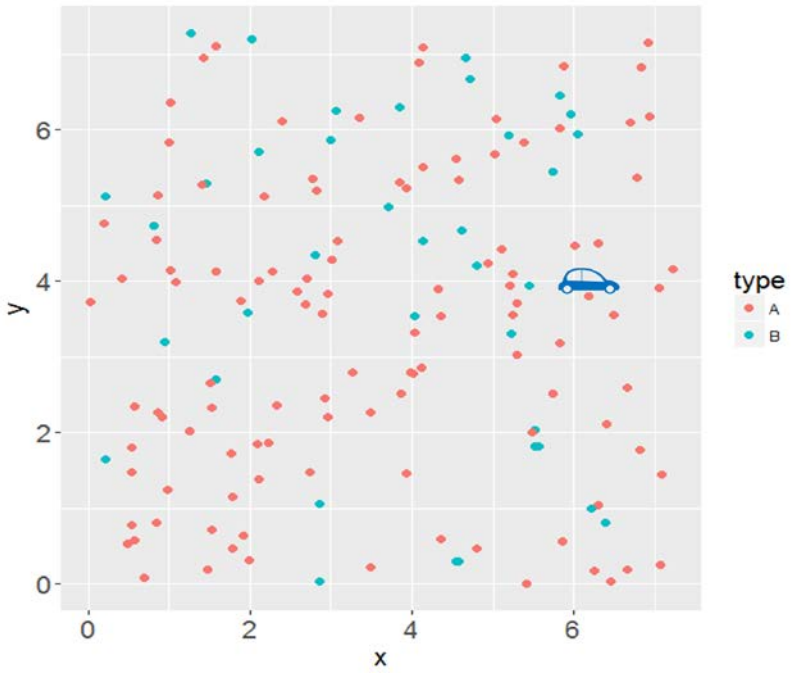
根据五分钟内订单情况推算区域总面积为 53 平方千米。根据题目假设（2），订单周围 1 公里内没有车辆的概率足够小，在本题中将临界值定为 1e-04.根据临界值计算每公里内有多少辆车，绘制图像得到结果为每平方公里平均有 2.844 辆车，因此总区域内车辆数约为 150 辆。



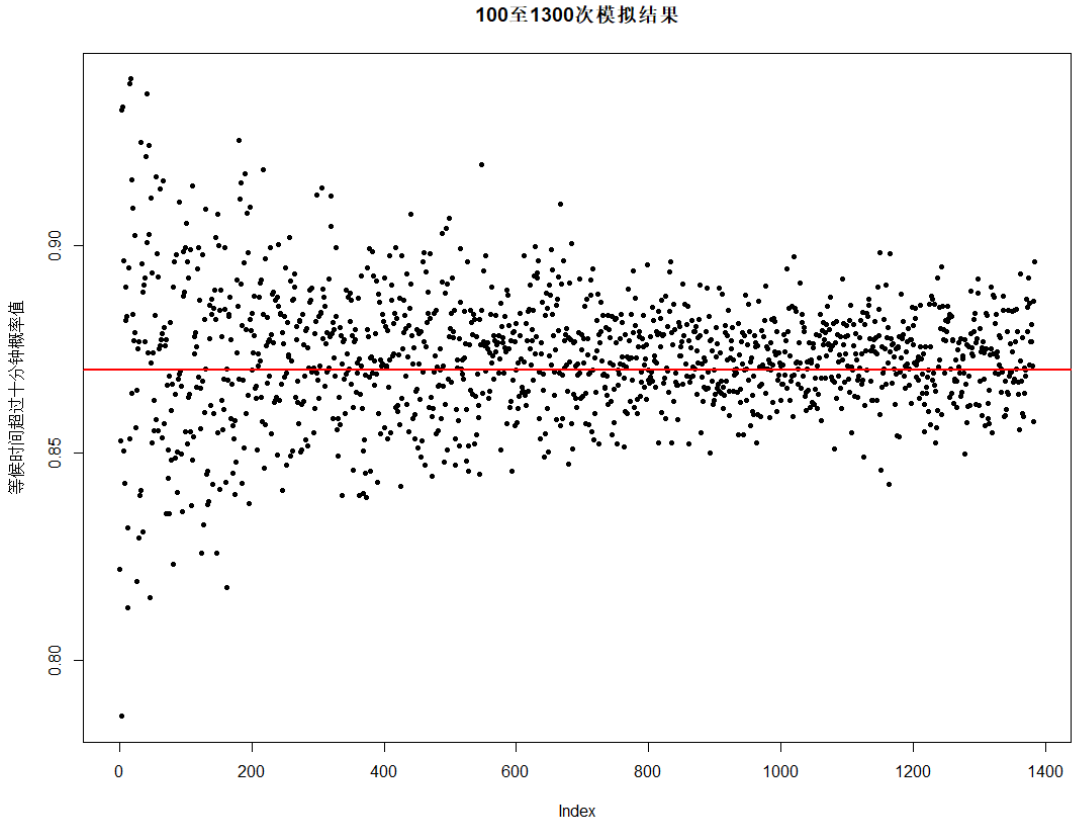
（3）进行 100 至 1400 次模拟，计算目标车辆等待超过 10 分钟接收不到订单的概率，并绘出散点图观察：

```r
time.istarget <- function(order,targetcar){
  # t表示订单接通服务时间；p表示是否目标车接到订单
  t <- 0
  p <- 0
  # 首先计算每一辆车到订单的distance
  car$distance <- sqrt((order[1]-car$x)**2+(order[2]-car$y)**2)
  # 目标车到订单的距离
  juli <- sqrt((order[1]-targetcar[1])**2+(order[2]-targetcar[2])**2)
  # 筛选出距离小于1km的车
  subcar <- car[car$distance<=1,]
  # 模拟生成ta与tb数值
  ta <- rgamma(1,shape=3.07,rate=0.2559)
  tb <- rnbinom(1,size=7.892,mu=25.57)
  # 判断订单类型（A/B）
  t <- ifelse(sum(subcar$type)==0,ta,tb)
  # 判断目标车是否接到该订单
  attach(subcar)
  if(sum(target)==1 & targetcar[3]==0 & sum(type)==0 & juli==min(subcar[,5]))p=1
  if(sum(target)==1 & targetcar[3]==1 & juli==min(subcar[type==1,5]))p=1

  detach(subcar)
  return(c(t,p))
}

#n辆车，a边长，下x、y是车坐标，type车类型（0是a，1是b），我们的车的标识target

set.seed(123)
a=sqrt(53)
n=150

ppp=vector(length = 1400)
for (j in 1:1400) {
  p=vector(length = j)
  for (i in 1:j) {
    x=runif(n,0,a)
    y=runif(n,0,a)
    target=rep(0,n)
    target[sample(1:n,1)]=1
    type=rep(0,n)
    type[sample(1:n,n/4,replace = F)]=1
    car=data.frame(x,y,type,target)
    rm(x,y,target,type)
    targetcar=as.vector(as.matrix(car[car$target==1,1:3]))
    gotten=0
    T=0
    while(gotten==0){
      order=c(runif(1,0,a),runif(1,0,a))
      res=time.istarget(order,targetcar)
      T=T+res[1]
      if(T>600) break
      if(res[2]==1)gotten=1
      }
    p[i]=ifelse(T>600,1,0)
  }
  ppp[j-100]=sum(p)/j
}
```

```
ggplot(car,aes(x=x,y=y))+geom_point(aes(colour=type),size=2)
```



所有车辆和目标车辆分布



观察图形可以看出，当模拟次数较少时，得到的概率波动较大，结果不稳定；而随着模拟次数增加，得到的概率不断收敛，波动减小，最终趋于稳定的值，

即目标司机等待时间超过十分钟概率大致在 0.87 附近。

2、问题二

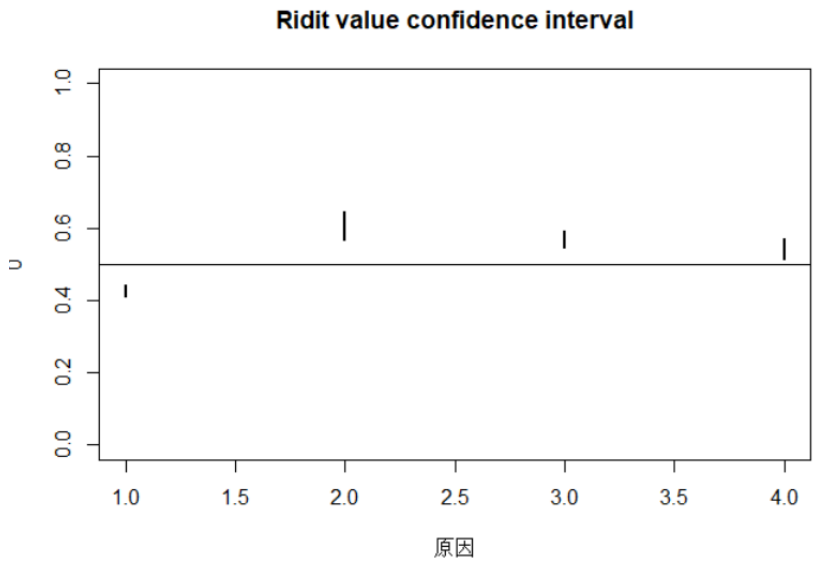
（1）导入数据，编写 Ridit 函数，计算每种原因对应的平均 Ridit 得分、95%可信限、统计量 W 值、Kruskal-Wallis 检验的 p 值；

```
a=c(984,113,320,274)
b=c(143,53,250,90)
c=c(49,30,33,84)
d=c(16,20,18,4)
e=c(8,10,6,1)
data=as.matrix(data.frame(a,b,c,d,e))

ridit_test=function(x)
{
  order.num=ncol(x)
  treat.num=nrow(x)
  rowsum=rowSums(x) #Oi.
  colsum=colSums(x) #O.i
  total=sum(rowsum)
  N=(colsum/2)[1:order.num]+c(0,(cumsum(colsum))[1:order.num-1])
  ri=N/total
  p_coni=x/outer(rowsum,rep(1,order.num),"*") #概率阵—i水平下属于第j顺序类的概率
  pi.=rowsum/total #属于第i水平的概率
  score=p_coni%*%ri #每个处理的得分
  confi_inter=matrix(c(score-1/sqrt(3*rowsum),score+1/sqrt(3*rowsum)),byrow = F,ncol = 2)
  if(length(rle(sort(ri))$lengths)==length(ri)) #不打结
  {w=(12*total/(total+1))*sum(rowsum*(score-0.5)^2)}
  if(length(rle(sort(ri))$lengths)<=length(ri)) #打结
  {
    tao=rle(sort(ri))$lengths
    T=1-sum(tao^3-tao)/(order.num^3-order.num)
    w=(12*total/(total+1)*T)*sum(rowsum*(score-0.5)^2)
  }
  pvalue=pchisq(w,treat.num-1,lower.tail = F)
  list(Score=score,confi_inter=confi_inter,W=w,Pvalue=pvalue)
}

ridit_test(data)
```

```
## $Score
##           [,1]
## [1,] 0.4273790
## [2,] 0.6057912
## [3,] 0.5703263
## [4,] 0.5422558
##
## $confi_inter
##           [,1]      [,2]
## [1,] 0.4107123 0.4440456
## [2,] 0.5673865 0.6441960
## [3,] 0.5472691 0.5933834
## [4,] 0.5151295 0.5693820
##
## $W
## [1] 153.1523
##
## $Pvalue
## [1] 5.504452e-33
```

（2）绘制每种原因对应的 Ridit 得分置信区间图，比较各组情况。

```
result=ridit_test(data)
graph=result$confi_inter
plot(0,0,ylim = c(0,1),xlim = c(1,4),xlab="原因",col="gray7",main="Ridit value confidence interval")
abline(h=0.5)
for(i in 1:nrow(graph))
{
  lines(c(i,i),graph[i,],lwd=2)
}
```



五、 结果分析与反思

1、问题一

通过模拟的方法我们可以得到等待 10 分钟以上的概率，当模拟 100 次时，得到的司机等待 10 分钟概率为 0.84；当模拟 1000 次时，得到的司机等待 10 分钟概率为 0.85；当模拟 10000 次时，得到的司机等待 10 分钟概率为 0.87，且模拟次数越多，得到的概率波动越小，最终的结果也趋于稳定。说明大概率情况下上等待时间会超过司机的忍耐程度，即 10 分钟。模型充分利用了题目所给出信息，在一系列合理假定的基础上得到了最多的 10000 次模拟中司机等待时间超过 10min 的次数，进一步计算得到司机等待时间超过 10min 的概率大致在 87%左右。

同时此模型也存在着一定的不足。使用基于北京市打车业务的情况来代表样本区域情况可能存在一定的误差，我们通过网上数据和 order 数据假定的场景区域面积 53km^2 不一定保证准确度，这在一定程度上可能影响了模型的准确性和稳健性。

2、问题二

由 Ridit 值置信区间图可知，“合理等待时间与感知等待时间的逆差”这一原因的 Ridit 得分显著低于平均 Ridit 得分，由于等级排序是从“非常不满意”到“非常满意”，所以可知，在此原因上，极大多数人都呈现不满意的情况，与总体情况存在显著差异，表明“合理等待时间与感知等待时间的逆差”这一原因应当受到重视。