**北京科技大学实验报告**

学院：计算机与通信工程学院 专业：通信工程 班级：通信1701

姓名：胡成成 学号：41724260 实验日期： 2020年 4月 7日

**实验名称：指数型随机变量相关分布函数验证**

**实验目的：**

1. 进一步加深对下列分布函数的理解。

Y = X1 + X2 + … + XN 符合参数为p的指数分布，

= X1 + X2 + … + 符合参数为(n, )的erlang分布，

其中X1，X2，…为独立同分布的参数为的指数型随机变量，N为参数为p（由1开始）的几何型随机变量。

1. 掌握基于累积分布函数生成随机数的方法。

**实验仪器：**

1. 操作系统：WIN10

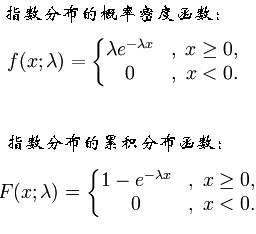
2. 操作软件：MATLAB R2018b

**实验原理：**

1. 指数分布

在概率理论和统计学中，指数分布是描述泊松过程中的事件之间的时间的概率分布，即事件以恒定平均速率连续且独立地发生的过程。指数函数的一个重要特征是无记忆性。这表示如果一个随机变量呈指数分布，当s，t>0时有P(T>t+s | T>t)=P(T>s)。

* 概率密度函数



* 分布函数



* 数学期望



* 方差



1. 几何分布

几何分布是离散型概率分布。定义为：在n次伯努利试验中，试验k次才得到第一次成功的机率。即：前k-1次皆失败，第k次成功的概率。在伯努利试验中，成功的概率为p，若ξ表示出现首次成功时的试验次数，则ξ是离散型随机变量，它只取正整数，且有P(ξ=k)=(1-p)的(k-1)次方乘以p (k=1，2，…，0<p<1)，此时称随机变量ξ服从几何分布。它的期望为1/p，方差为(1-p)/(p\*p)。

* 分布列

，

* 期望



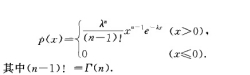
* 方差



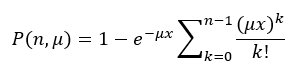
1. 爱尔朗分布

概率与统计相关学科中，爱尔朗分布（Erlang Distribution）是一种连续型概率分布。爱尔朗分布与指数分布一样多用来表示独立随机事件发生的时间间隔。相比于指数分布，爱尔朗分布能更好地对现实数据进行拟合（更适用于多个串行过程，或无记忆性假设不显著的情况下）。一般通过将爱尔朗过程分解为多个指数过程的技巧来对爱尔朗分布进行分析。

* 概率密度：



* 累积分布函数：



**实验内容与步骤：**

1. 输入：几何分布参数p，指数分布参数；
2. 过程：
   1. 设n = 向上取整，通过生成随机数N, X1, X2,…, 生成Y与，重复该操作1e6次；
   2. 随机生成N, X1, …, 时不允许直接调用自带指数/几何分布随机生成函数，需要用rand()函数间接生成；
   3. 对随机生成的X1，Y，进行数值统计，包括各自的累积分布（横轴100个点）、均值、方差；
3. 输出：显示数值统计得到的均值与方差；在一幅图中分别基于数值统计与理论公式描绘X1，Y，的累积分布（横轴100个点），验证统计值与理论值是否吻合。

**程序说明及流程图：**

1. 设置输入的几何分布与指数分布的参数，，p=0.3：

mu=0.2;

p=0.3;

n=ceil(1/p);

1. 初始化X，Y，并通过随机数生成N, X1, X2,…, 生成Y与：

y=zeros(1,1e6); %初始化Y

x\_sum=zeros(1,1e6); %初始化X

Erln=zeros(1,1e6); %初始化

for i=1:1e6

x=(-1/mu)\*log(1-rand());

x\_sum(1,i)=x;

end

for i=1:1e6

for j=1:ceil(log(1-rand())/log(1-p))

x=(-1/mu)\*log(1-rand());

y(1,i)=y(1,i)+x;

end

end

for i=1:1e6

for j=1:n

x=(-1/mu)\*log(1-rand());

Erln(1,i)=Erln(1,i)+x;

end

end

1. 计算数值统计的得到的均值与方差

mean(x\_sum)

var(x\_sum)

mean(y)

var(y)

mean(Erln)

var(Erln)

1. 计算数值统计与理论计算值并提取其中100对数值，为后续作图准备：

%计算x numerical部分

[range\_x,mid\_x]=hist(x\_sum,100);

num\_x=numel(x\_sum);

density\_x=cumsum(range\_x/num\_x);

%计算x 理论部分

x\_theory=1-exp(-mu\*t);

% 计算y numerical部分

[range\_y,mid\_y]=hist(y,100);

num\_y=numel(y);

density\_y=cumsum(range\_y/num\_y);

%计算y 理论部分

t=1:1:100;

y\_theory=1-exp(-mu\*p\*t);

%计算erln numerical部分

[range\_erln,mid\_erln]=hist(Erln,100);

num\_erln=numel(Erln);

density\_erln=cumsum(range\_erln/num\_erln);

%计算erln 理论部分

y\_temp=zeros(1,100);

for i=1:100

for k=0:n-1

y\_temp(1,i)=y\_temp(1,i)+(mu\*i)^k/factorial(k);

end

end

real\_y=zeros(1,100);

for i=1:100

real\_y(1,i)=1-exp(-mu\*i)\*y\_temp(1,i);

end

1. 对比数值统计与理论计算值做出图像：

%绘图

plot(mid\_x,density\_x,'sr')

axis([0 80,-inf inf])

hold on;

plot(mid\_y,density\_y,'om')

hold on;

plot(mid\_erln,density\_erln,'xb')

hold on;

plot(t,x\_theory,'--b')

hold on;

plot(t,y\_theory,':.m')

hold on;

plot(t,real\_y,'\*y')

legend('x-numerical','y-numerical','erln-numerical','x-theory','y-theory','erlang-theory')

**实验结果与分析：**

1. 数值统计计算的均值方差：

ans = 4.9872 %X1均值

ans =24.8066 %X1方差

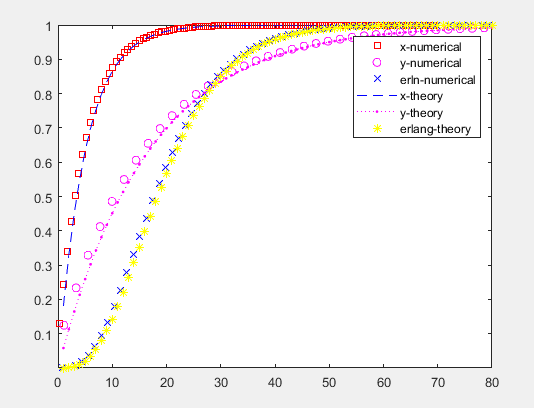
ans =16.6681 %Y均值

ans =277.7838 %Y方差

ans =19.9959 %均值

ans =100.0758 %方差

1. 验证统计值与理论值是否吻合的图像输出，其中，p=0.3：



根据输出的图像可以看出，数值统计与理论计算的吻合度很高，基本上差距不大。

**遇到的问题：**

1. 前期对概念理解不透彻，特别是分布的实际物理意义，难以下手去实践
2. 编程过程中矩阵维度的统一，容易出现“0：n”与“1：n”的矛盾
3. 区分“.\*”与“\*”在矩阵运算的区别

**实验总结：**

通过本次实验，将指数分布与几何分布从原理性角度通过编程实践去理解验证，并对爱尔兰分布一同实践验证，从本质上更深入的学习了这些分布的意义，对随机过程有了新的理解。此次实验也再次加固了随机过程的学习，有利于之后对排队论知识的学习与理解。

**附录（程序源代码）:**

**Verificationtest.m主程序**

mu=0.2;

p=0.3;

n=ceil(1/p);

y=zeros(1,1e6);

x\_sum=zeros(1,1e6);

Erln=zeros(1,1e6);

for i=1:1e6

for j=1:n

x=(-1/mu)\*log(1-rand());

Erln(1,i)=Erln(1,i)+x;

end

end

for i=1:1e6

for j=1:ceil(log(1-rand())/log(1-p))

x=(-1/mu)\*log(1-rand());

y(1,i)=y(1,i)+x;

end

end

for i=1:1e6

x=(-1/mu)\*log(1-rand());

x\_sum(1,i)=x;

end

mean(x\_sum);

var(x\_sum);

mean(y);

var(y);

mean(Erln);

var(Erln);

% 计算y numerical部分

[range\_y,mid\_y]=hist(y,100);

num\_y=numel(y);

density\_y=cumsum(range\_y/num\_y);

%计算y 理论部分

t=1:1:100;

y\_theory=1-exp(-mu\*p\*t);

%计算erln numerical部分

[range\_erln,mid\_erln]=hist(Erln,100);

num\_erln=numel(Erln);

density\_erln=cumsum(range\_erln/num\_erln);

%计算erln 理论部分

y\_temp=zeros(1,100);

for i=1:100

for k=0:n-1

y\_temp(1,i)=y\_temp(1,i)+(mu\*i)^k/factorial(k);

end

end

real\_y=zeros(1,100);

for i=1:100

real\_y(1,i)=1-exp(-mu\*i)\*y\_temp(1,i);

end

plot(real\_y)

%计算x numerical部分

[range\_x,mid\_x]=hist(x\_sum,100);

num\_x=numel(x\_sum);

density\_x=cumsum(range\_x/num\_x);

%计算x 理论部分

x\_theory=1-exp(-mu\*t);

%绘图

plot(mid\_x,density\_x,'sr')

axis([0 80,-inf inf])

hold on;

plot(mid\_y,density\_y,'om')

hold on;

plot(mid\_erln,density\_erln,'xb')

hold on;

plot(t,x\_theory,'--b')

hold on;

plot(t,y\_theory,':.m')

hold on;

plot(t,real\_y,'\*y')

legend('x-numerical','y-numerical','erln-numerical','x-theory','y-theory','erlang-theory')