**上机作业三**

1. 完成浦丰投针试验（实验一）的理论分析，用R编程实现仿真模拟分析。

理论分析：

设针与平行线的夹角（锐角）为，中心距离最近的一条直线为。则。

于是

buffon <- function(l){

  theta <- runif(1, max = pi/2)

  D <- runif(1, max = l)

  if(D <= l\*sin(theta)) return(T)

  else return(F)

}

N <- 10000;

l = 1;

sum <- 0;

for(i in  1:N){

  if(buffon(l)) sum = sum + 1;

}

p <- sum/N; p

[1] 0.6286

1. 完成电梯问题（实验二）的理论分析，用R编程实现仿真模拟分析。

理论分析：

设，则，即。

设，则。

当时，

# T2

elevator <- function(r, n){

  x <- sample(1:n, r, replace = T)

  return(length(unique(x)))

}

N <- 10000;

sum <- 0;

for(i in  1:N){

  sum = sum + elevator(10, 7);

}

E <- sum/N; E

[1] 5.4959

1. 完成掷骰子问题（实验三）的理论分析，用R编程实现仿真模拟分析。

理论分析：

dice <- function(n, t, m){ # n dices, throw t times, at least m 6s

  sum <- 0

  for(i in 1:t){

    x <- sample(1:6, n, replace = T)

    if(all(x == 6)) sum = sum + 1;

    if(sum == m) return(T)

  }

  return(F)

}

N <- 10000;

sum <- 0;

for(i in  1:N){

  if(dice(1, 4, 1)) sum = sum + 1;

}

p1 <- sum/N; p1

sum <- 0;

for(i in  1:N){

  if(dice(2, 24, 1)) sum = sum + 1;

}

p2 <- sum/N; p2

[1] 0.5102

[1] 0.4812

1. 完成报童问题（实验四）的理论分析，用R编程实现仿真模拟分析。

理论分析：

当报童买进份报，实际卖报数的分布为

当时，若，；若，。

fun <- function(a, b, lambda, n){

  sum1 <- 0

  for(i in 1:(n-1)){

    sum1 <- sum1 + lambda^i/factorial(i-1)\*exp(1)^(-lambda)

  }

  sum2 <- 0

  for(i in 0:(n-1)){

    sum2 <- sum2 + lambda^i/factorial(i)\*exp(1)^(-lambda)

  }

  return(a\*n + (a + b)\*(sum1 - n\*sum2))

}

fun(1.5, 0.6, 120, 100)

fun(1.5, 0.6, 120, 140)

[1] 149.7414

[1] 167.6451

newspaper <- function(a, b, lambda, n){

  x <- rpois(1, lambda)

  if(x > n) x <- n

  rt <- a\*x - b\*(n-x); rt

}

N <- 10000;

sum <- 0;

for(i in 1:N){

  sum <- sum + newspaper(1.5, 0.6, 120, 100);

}

E <- sum/N; E

sum <- 0;

for(i in 1:N){

  sum <- sum + newspaper(1.5, 0.6, 120, 140);

}

E <- sum/N; E

[1] 149.7344

[1] 167.7532

1. 完成12个乒乓球中有3个旧球问题（实验八）的理论分析，用R编程实现仿真模拟分析。

理论分析：

设：第一次取到的三个球中有个是新的，则

设：第二次取到的三个球全是新的，则

于是

pa <- rep(0,4);

pba <- rep(0,4);

pb <- 0;

for(i in 0:3){

  pa[i+1] <- choose(9,i)\*choose(3,3-i)/choose(12,3);

  pba[i+1] <- choose(9-i,3)/choose(12,3);

  pb <- pb + pa[i+1]\*pba[i+1];

}

pb

pa3b <- pba[4]\*pa[4]/pb; pa3b

[1] 0.1457851

[1] 0.2380952

firsttime <- function(){

  left <- 9

  x <- sample(1:12, 3, replace = F)

  for(i in 1:3){

    if(x[i] <= 9) left = left - 1

  }

  left

}# return number of left new balls

fun <- function(n){

  x <- sample(1:12, 3, replace = F)

  for(i in 1:3){

    if(x[i] > n) return(F)

  }

  return(T)

}

N <- 10000;

sum <- 0;

for(i in 1:N){

  if(fun(firsttime())) sum <- sum + 1;

}

p <- sum/N; p

n <- 0;

sum <- 0;

for(i in 1:N){

  t <- firsttime();

  if(fun(t)){

    n = n + 1;

    if(t == 6) sum = sum + 1;

  }

}

p <- sum/n; p

[1] 0.1429

[1] 0.2510431

1. 随机抛掷两枚均匀的骰子，求双骰子点数和5点出现在7点前的概率.

理论分析：

记：首次投到5时投了次；：前次都没有投到7。则

twodicesum <- function(){

  x <- sample(1:6, 2, replace = T)

  return(sum(x))

}

throw <- function(){

  repeat{

    x <- twodicesum()

    if(x == 5) return(T);

    if(x == 7) return(F);

  }

}

N <- 10000;

sum <- 0;

for(i in 1:N){

  if(throw()) sum = sum + 1;

}

p <- sum/N; p

[1] 0.3929