midterm1

张孟涵

10/18/2019

1

(a)

```
NB_by_geometric <- function(n,r,p){
  X <- rep(0, n)
  for(j in 1:n){
    for(i in 1:r){
      U <- runif(1, 0, 1)
      X[j]=X[j]+ceiling(log(U)/log(1-p))   #geometric
    }
  }
  return(X)
}
```

(b)

$$\frac{P_{j+1}}{P_j} = \frac{\frac{(j+1-1)!}{(j+1-r)!(r-1)!}\cancel{p}^r(1-p)^{j+1-r}}{\frac{(j-1)!}{(j-r)!(r-1)!}\cancel{p}^r(1-p)^{j-r}} = \frac{j}{j+1-r}(1-p)$$

$$\Rightarrow P_{j+1} = \frac{j(1-p)}{j+1-r}P_j$$

(c)

```r
NB<- function(n,r,p){
  X <- rep(NA, n)
  U=runif(n,0,1)
  for(i in 1:n){
    pr=p^r
    F=pr
    j=r   #注意 i=r 開始 不是從0
    while(F<=U[i]){
      pr=j/(j+1-r)*(1-p)*pr
      F=F+pr
      j=j+1
    }
    X[i]=j
  }
  return(X)
}
```

(d)

```r
NB_by_bernoulli<- function(n,r,p){
  success <- rep(0, n)
  X <- rep(NA, n)
  for(i in 1:n){
    j=0
    while(success[i]<5){
      U <- runif(1, 0, 1)
      success[i]=success[i]+ifelse(U<p,1,0)
      j=j+1
    }
    X[i]=j
  }
  return(X)
}
```

(e)

```r
r=5
p=0.3
#exact
r/p
```

```
## [1] 16.66667
```

```r
#simulation
nb1=NB_by_geometric(100000,r,p)
mean(nb1)
```

```
## [1] 16.65995
```

```
nb2=NB(100000,r,p)
mean(nb2)
```

```
## [1] 16.64808
```

```
nb3=NB_by_bernoulli(100000,r,p)
mean(nb3)
```

```
## [1] 16.68727
```

Yes,these expected values by the three simulation methods are closed.

2

```
#法一(較快)
rolls_times_1=function(n){
  Y=rep(NA,n)
  for(i in 1:n){
    X=rep(0,11)
    names(X)=seq(2,12)
    while(length(X[X==0])>0){
      U=runif(2,0,1)
      X[sum(ceiling(6*U))-1]=X[sum(ceiling(6*U))-1]+1
    }
    Y[i]=sum(X)
  }
  return(Y)
}
#法二(較慢)
rolls_times_2=function(n){
  Y=rep(NA,n)
  for(i in 1:n){
    X=c()
    while(sum(table(X)>0)<11){
      U=runif(2,0,1)
      X=c(X,sum(ceiling(6*U)))
    }
    Y[i]=length(X)
  }
  return(Y)
}

Y=rolls_times_1(10000)
mean(Y)
```

```
## [1] 60.94
```

3

(a)

```
inverse_transform=function(n,p){
  F=cumsum(p)
  U=runif(n,0,1)
  X=rep(NA,n)
  for(i in 1:n){
    for(j in 2:length(p)){
      if(F[j-1]<=U[i] && U[i]<F[j]){
        X[i]=j
        break
      }
    }
  }
  return(X)
}
```

(b)

```
AR=function(n,p){
  k=10
  c=max(p[5:14]/(1/k))
  X=rep(NA,n)
  iter=rep(NA,n)
  for(i in 1:n){
    DU_5_14=ceiling(10*runif(1,0,1))+4
    U=runif(1,0,1)
    iter[i]=1
    while(U>=p[DU_5_14]/(1/k)/c){
      DU_5_14=ceiling(10*runif(1,0,1))+4
      U=runif(1,0,1)
      iter[i]=iter[i]+1
    }
    X[i]=DU_5_14
  }

  res=list()
  res$X=X
  res$c=c
  res$iter=iter
  return(res)
}

p=c(0,0,0,0,rep(c(0.11,0.09),5))
ar=AR(10000,p)
c=ar$c
iter=mean(ar$iter)
c
```

```
## [1] 1.1
```

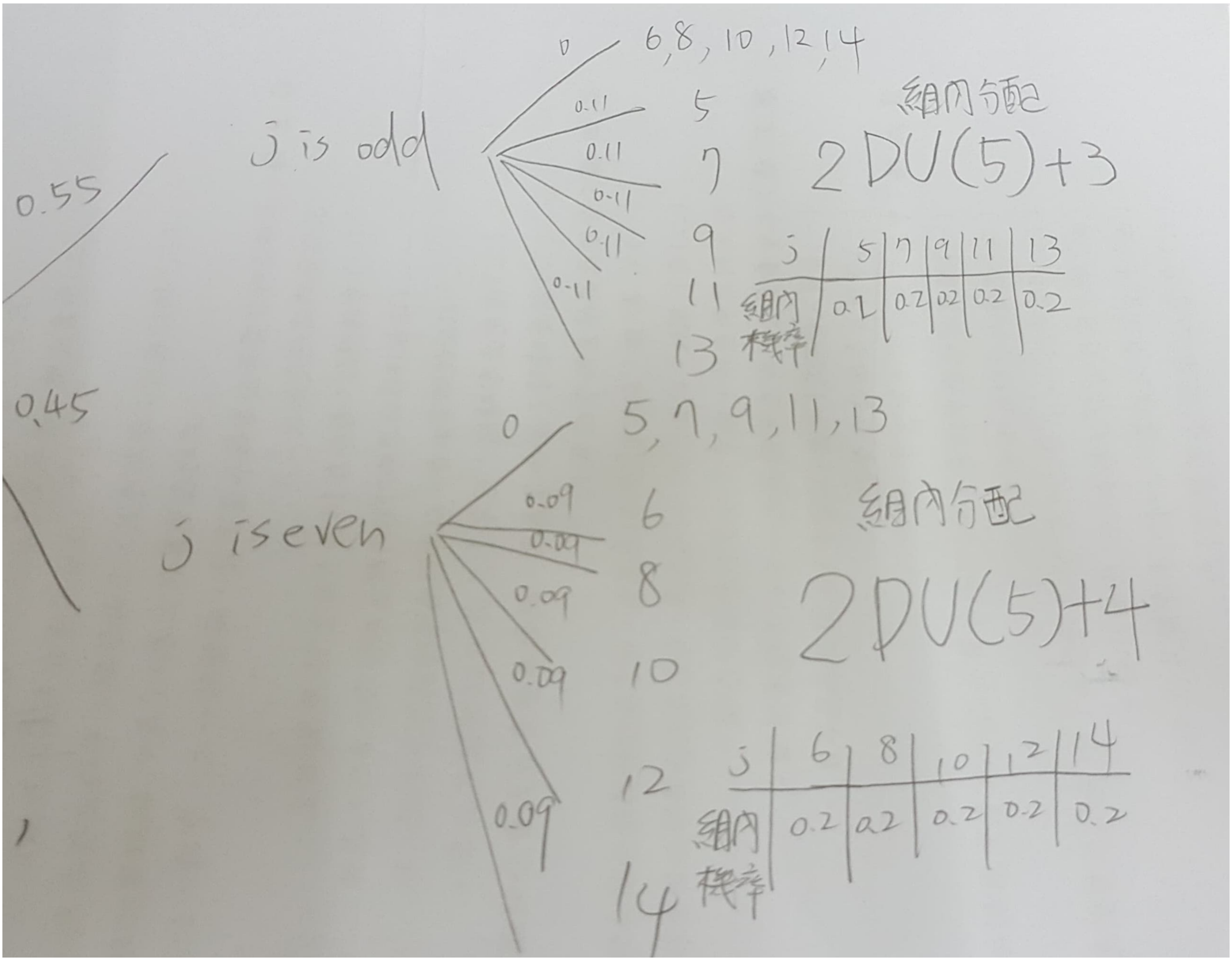
```
iter
```

```
## [1] 1.1006
```

c=1.1 計算方法:max[pi/(1/10)] for all i

average of iteration number of AR method =1.1006

(c)



```r
composite=function(n){
  X=rep(NA,n)
  U=runif(n,0,1)
  U2=runif(n,0,1)
  for(i in 1:n){
    if(U[i]<0.55){
      X[i]=ceiling(5*U2[i])*2+3
    }
    else{
      X[i]=ceiling(5*U2[i])*2+4
    }
  }
  return(X)
}
```

(d)

```r
#exact expected value
p=c(0,0,0,0,rep(c(0.11,0.09),5))
expected_value=0
for(i in 1:length(p)){
  expected_value=expected_value+i*p[i]
}
expected_value
```

```
## [1] 9.45
```

```r
#simulation
X_inverse_transform=inverse_transform(10000,p)
mean(X_inverse_transform)
```

```
## [1] 9.4333
```

```r
ar=AR(10000,p)
X_ar=ar$X
mean(X_ar)
```

```
## [1] 9.4501
```

```r
X_composite=composite(10000)
mean(X_composite)
```

```
## [1] 9.432
```

Yes,these expected values by the three simulation methods are closed.