# R作业2程序及结果(仅供参考)

March 17, 2016

1.读写数据文件

(1)

```
write.table(cars,"carsdata.txt")
read.table("carsdata.txt")

   speed dist
1      4    2
2      4   10
3      7    4
4      7   22
5      8   16
6      9   10
7     10   18
8     10   26
9     10   34
10    11   17
11    11   28
12    12   14
13    12   20
14    12   24
15    12   28
16    13   26
17    13   34
18    13   34
19    13   46
20    14   26
21    14   36
22    14   60
23    14   80
24    15   20
25    15   26
26    15   54
27    16   32
```

```
28     16   40
29     17   32
30     17   40
31     17   50
32     18   42
33     18   56
34     18   76
35     18   84
36     19   36
37     19   46
38     19   68
39     20   32
40     20   48
41     20   52
42     20   56
43     20   64
44     22   66
45     23   54
46     24   70
47     24   92
48     24   93
49     24  120
50     25   85
```

(2)

```
Air=AirPassengers;Titan=Titanic
save(Air,Titan,file="yourname.Rdata")
```

(3)

```
rm(Air,Titan)
```

(4)

```
load("yourname.Rdata")
Air;Titan

     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1949 112 118 132 129 121 135 148 148 136 119 104 118
1950 115 126 141 135 125 149 170 170 158 133 114 140
1951 145 150 178 163 172 178 199 199 184 162 146 166
1952 171 180 193 181 183 218 230 242 209 191 172 194
1953 196 196 236 235 229 243 264 272 237 211 180 201
1954 204 188 235 227 234 264 302 293 259 229 203 229
1955 242 233 267 269 270 315 364 347 312 274 237 278
```

```
1956 284 277 317 313 318 374 413 405 355 306 271 306
1957 315 301 356 348 355 422 465 467 404 347 305 336
1958 340 318 362 348 363 435 491 505 404 359 310 337
1959 360 342 406 396 420 472 548 559 463 407 362 405
1960 417 391 419 461 472 535 622 606 508 461 390 432
, , Age = Child, Survived = No

      Sex
Class  Male Female
  1st     0      0
  2nd     0      0
  3rd    35     17
  Crew    0      0

, , Age = Adult, Survived = No

      Sex
Class  Male Female
  1st   118      4
  2nd   154     13
  3rd   387     89
  Crew  670      3

, , Age = Child, Survived = Yes

      Sex
Class  Male Female
  1st     5      1
  2nd    11     13
  3rd    13     14
  Crew    0      0

, , Age = Adult, Survived = Yes

      Sex
Class  Male Female
  1st    57    140
  2nd    14     80
  3rd    75     76
  Crew  192     20
```

2.回归分析

```
attach(cars)
x=cbind(1,speed);y=dist
n=length(y);p=dim(x)[2]
```

```
reg.analysis=function(x,y,alpha){
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  #估计
  beta.hat=solve(t(x)%*%x,t(x)%*%y) #最小二乘估计
  beta.hat=as.vector(beta.hat)  #转化为向量
  sigma2.hat=sum((y-x%*%beta.hat)^2)/(n-p) #sigma^2的最小二乘估计
  sigma.hat=sqrt(sigma2.hat)
  cov.matrix=sigma2.hat*solve(t(x)%*%x)  #估计的协方差矩阵
  beta.sd=sqrt(diag(cov.matrix))   #估计的beta.hat的标准差
  beta.sd=as.vector(beta.sd)  #转化为向量
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  #置信区间
  beta.left=beta.hat-beta.sd*qt(1-alpha/2,n-p)  #区间左端点
  beta.right=beta.hat+beta.sd*qt(1-alpha/2,n-p) #区间右端点
  beta.interval=matrix(c(beta.left,beta.right),ncol=2,
                       dimnames=list(c("beta0","beta1"),c("left","right")))  #95%置
信区间
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  #残差
  y.hat=x%*%beta.hat
  res=y-y.hat                     #残差
  H=x%*%solve(t(x)%*%x)%*%t(x)    #帽子矩阵
  h=diag(H)
  s.res=res/(sigma.hat*sqrt(1-h)) #标准化残差
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  #检验
  beta1.hat=beta.hat[2]
  sxx=sum((x[,2]-mean(x[,2]))^2)
  test.stat=beta1.hat*sqrt(sxx)/sigma.hat  #检验统计量的观测值
  p_value=2*(1-pt(abs(test.stat),n-p))     #检验的p_value
  #~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  #输出结果
  list(beta.hat=beta.hat,
       beta.interval=beta.interval,
       res=res,
       s.res=s.res,
       test.stat=test.stat,
       p_value=p_value
       )
}
reg.analysis(x,y,alpha=0.05)


$beta.hat
[1] -17.579095   3.932409
```

```
$beta.interval
            left     right
beta0 -31.167850 -3.990340
beta1   3.096964  4.767853

$res
            [,1]
 [1,]   3.849460
 [2,]  11.849460
 [3,]  -5.947766
 [4,]  12.052234
 [5,]   2.119825
 [6,]  -7.812584
 [7,]  -3.744993
 [8,]   4.255007
 [9,]  12.255007
[10,]  -8.677401
[11,]   2.322599
[12,] -15.609810
[13,]  -9.609810
[14,]  -5.609810
[15,]  -1.609810
[16,]  -7.542219
[17,]   0.457781
[18,]   0.457781
[19,]  12.457781
[20,] -11.474628
[21,]  -1.474628
[22,]  22.525372
[23,]  42.525372
[24,] -21.407036
[25,] -15.407036
[26,]  12.592964
[27,] -13.339445
[28,]  -5.339445
[29,] -17.271854
[30,]  -9.271854
[31,]   0.728146
[32,] -11.204263
[33,]   2.795737
[34,]  22.795737
[35,]  30.795737
[36,] -21.136672
[37,] -11.136672
[38,]  10.863328
```

```
[39,] -29.069080
[40,] -13.069080
[41,]  -9.069080
[42,]  -5.069080
[43,]   2.930920
[44,]  -2.933898
[45,] -18.866307
[46,]  -6.798715
[47,]  15.201285
[48,]  16.201285
[49,]  43.201285
[50,]   4.268876

$s.res
             [,1]
 [1,]  0.26604155
 [2,]  0.81893273
 [3,] -0.40134618
 [4,]  0.81326629
 [5,]  0.14216236
 [6,] -0.52115255
 [7,] -0.24869180
 [8,]  0.28256008
 [9,]  0.81381197
[10,] -0.57409795
[11,]  0.15366341
[12,] -1.02971654
[13,] -0.63392061
[14,] -0.37005667
[15,] -0.10619272
[16,] -0.49644946
[17,]  0.03013240
[18,]  0.03013240
[19,]  0.82000518
[20,] -0.75422016
[21,] -0.09692637
[22,]  1.48057874
[23,]  2.79516632
[24,] -1.40612757
[25,] -1.01201579
[26,]  0.82717256
[27,] -0.87627072
[28,] -0.35074918
[29,] -1.13552237
[30,] -0.60956963
```

```
[31,]  0.04787130
[32,] -0.73777117
[33,]  0.18409193
[34,]  1.50103921
[35,]  2.02781813
[36,] -1.39503525
[37,] -0.73502818
[38,]  0.71698735
[39,] -1.92452335
[40,] -0.86524066
[41,] -0.60041999
[42,] -0.33559931
[43,]  0.19404203
[44,] -0.19590672
[45,] -1.26671228
[46,] -0.45938126
[47,]  1.02713306
[48,]  1.09470190
[49,]  2.91906038
[50,]  0.29053451

$test.stat
[1] 9.46399

$p_value
[1] 1.489919e-12
```

3.利用蒲丰投针试验估计圆周率

```
l=1  #针长
d=2  #两平行线间距(l<d)
n=1000000
set.seed(123)
x=runif(n,0,pi)
y=runif(n,0,d/2)
freq=mean(y<=l/2*sin(x)) #概率的估计
pi.hat=2*l/(d*freq)
pi.hat

[1] 3.153192
```

4.计算定积分的近似值

```
den=function(x) 1/sqrt(2*pi)*exp(-x^2/2) #定义被积函数，即dnorm
a=1;b=5
m=100000
```

(1)黎曼近似

```
d=(b-a)/m  #区间长度
x=seq(a-d,b+d,by=d)  #取点
I1=sum(den(x)*d)
I1

[1] 0.1586695
```

(2)蒲丰投针

```
x=runif(m,a,b)
y=runif(m,0,den(a))
freq=mean(y<=den(x))
I.hat2=(b-a)*den(a)*freq
I.hat2

[1] 0.1587328
```

(3)大数律

```
x=runif(m,a,b)
I.hat3=(b-a)*mean(den(x))
I.hat3

[1] 0.1588154
```

(4)用pnorm

```
I=pnorm(b)-pnorm(a)
I

[1] 0.158655
```

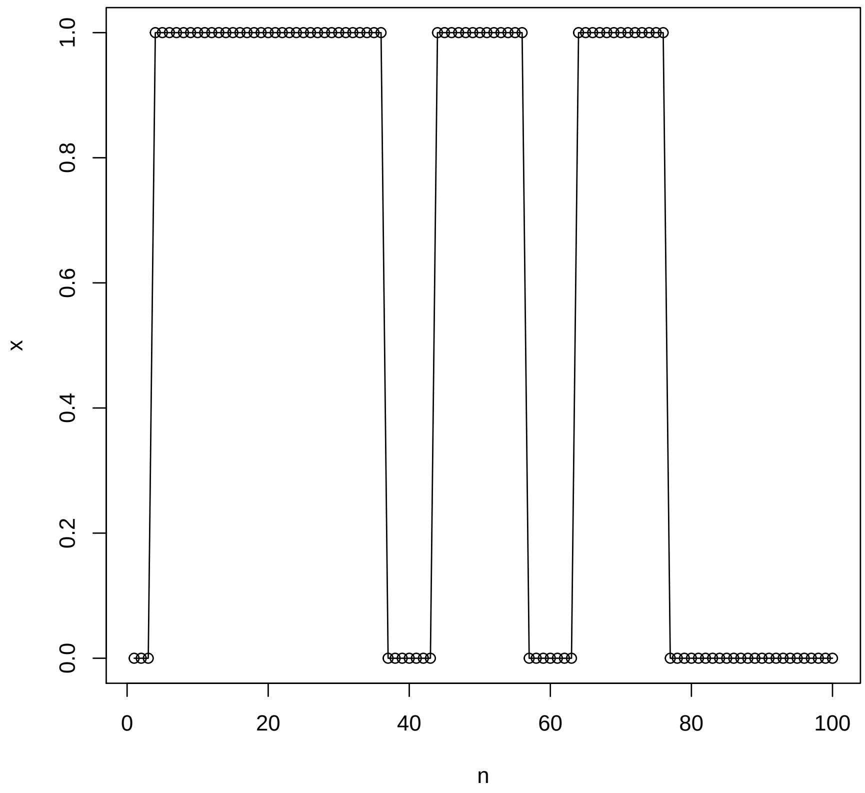
注：亦可将上述程序写成函数，参数为积分上下限、被积函数、模拟次数. 例：蒲丰投针

```
int=function(a,b,fun,m){
  x=runif(m,a,b)
  y=runif(m,0,den(a))
  freq=mean(y<=den(x))
  I.hat2=(b-a)*den(a)*freq
  I.hat2
}
int(a,b,den,m)

[1] 0.1598265
```

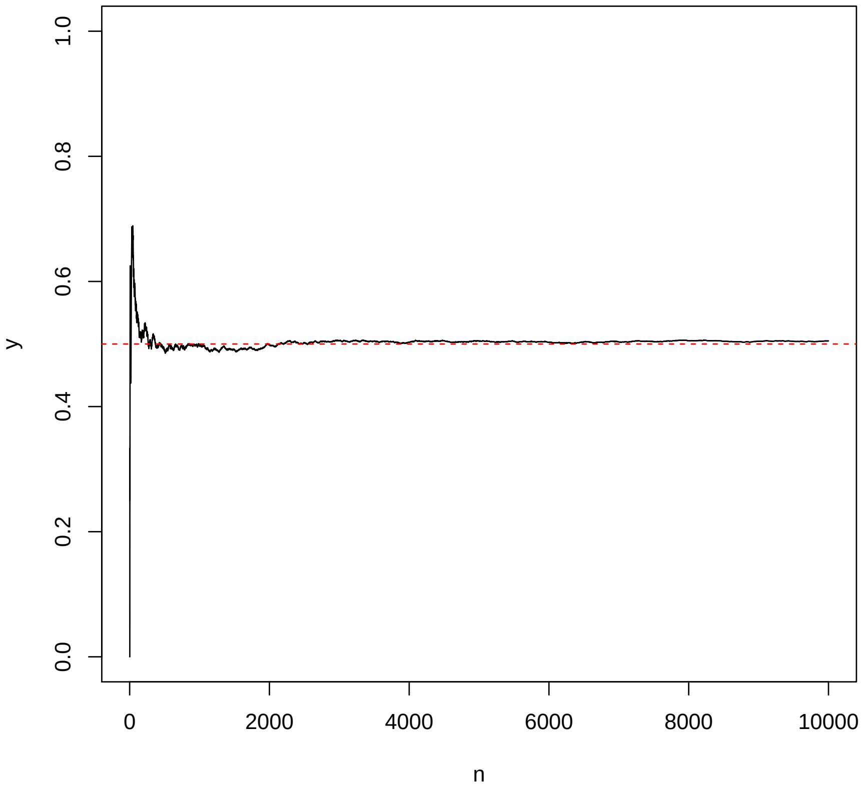
5.模拟马氏链

```
set.seed(12)
m=100
x=numeric(m)
x[1]=0
for (i in 2:m){
  if(x[i-1]==0)
    x[i]=rbinom(1,1,0.06)
  else
    x[i]=rbinom(1,1,0.9)
}
plot(1:m,x,xlab="n")
lines(1:m,x)
```



6.模拟抛硬币试验

```
m=10000
n=1:m
h=rbinom(m,1,1/2) #or h=sample(c(0,1),m,replace=T)
y=cumsum(h)/n
plot(n,y,type="l",ylim=c(0,1))
abline(h=0.5,lty=2,col=2)
```



7.解一元非线性方程的牛顿算法

```
Newtons=function(fun,x,ep=1e-5,it_max=100){
  index=0;k=1
  while(k<=it_max){
     x1=x;obj=fun(x)
     x=x-obj$f/obj$J
     norm=abs(x-x1)
     if(norm<ep){
        index=1;break
     }
     k=k+1
  }
  obj=fun(x)
  list(root=x,it=k,index=index,Funval=obj$f)
}
```

与方程有关的函数

```
funs=function(x){
   f=x^3-x-1
   J=3*x^2-1
   list(f=f,J=J)
}
```

求解方程

```
Newtons(funs,0)

$root
[1] 1.324718

$it
[1] 21

$index
[1] 1

$Funval
[1] 2.74647e-12
```