

Computer Practical: Transformation Methods and Rejection Sampling Model Answers

Model code in Python

```

1 from scipy import *
2 from scipy import random
3 from scipy import stats
4 import pylab
5
6 # Task 1
7
8 def sampleExpo(lam, n):
9     u = random.random_sample(n)
10    x = -log(u)/lam
11    return x
12
13 # Task 2
14
15 lam = 2
16 x = sampleExpo(lam,1000)
17
18 pylab.figure()
19 pylab.hist(x, normed=True)
20 xi = linspace(0,5,100)
21 pylab.plot(xi, lam*exp(-lam*xi))
22 pylab.show()
23
24 # Task 3
25
26 def sampleGammaInt(k, lam):
27     return sum(sampleExpo(lam, k))
28
29 # Task 4
30
31 def gammaPDF(x, alpha, lam):
32     return stats.gamma(alpha).pdf(x*lam)*lam
33
34 alpha = 5.7
35 k = floor(alpha)
36 lam = 2
37
38 M = gammaPDF(alpha-k,alpha,lam)/gammaPDF(alpha-k,k,lam-1)
39
40 xi = linspace(0,10,100)
41 pylab.figure()
42 pylab.plot(xi, gammaPDF(xi,alpha,lam), 'b-')
43 pylab.plot(xi, M*gammaPDF(xi,k,lam-1), 'r:')
44 pylab.show()
45
46 # Task 5
47
48 def sampleGamma(alpha, lam):
49     k = floor(alpha)
50     M = gammaPDF(alpha-k,alpha,lam)/gammaPDF(alpha-k,k,lam-1)
51     while True:
52         x = sampleGammaInt(k,lam-1)
53         prob_of_accept = gammaPDF(x,alpha,lam)/(M*gammaPDF(x,k,lam-1))
54         if random.random_sample(1)<prob_of_accept:
55             return x
56
57 alpha = 5.7

```

```
58 lam = 2
59 x = [sampleGamma(alpha,lam) for i in xrange(1000)]
60
61 # Task 6
62
63 pylab.figure()
64 pylab.hist(x, normed=True)
65 xi = linspace(0,10,100)
66 pylab.plot(xi, gammaPDF(xi, alpha,lam))
67 pylab.show()
```

Model code in R

```
1 # Task 1
2
3 sampleExpo <- function(lam, n) {
4   u <- runif(n)
5   x <- -log(u)/lam
6   x
7 }
8
9 # Task 2
10
11 lam <- 2
12 x <- sampleExpo(lam, 1000)
13
14 hist(x, freq=FALSE, col="grey")
15 xi <- seq(0, 5, length.out=100)
16 lines(xi, lam*exp(-lam*xi), lwd=2)
17
18
19 # Task 3
20
21 sampleGammaInt <- function(k, lam) {
22   sum(sampleExpo(lam, k))
23 }
24
25 # Task 4
26
27 alpha <- 5.7
28 k <- floor(alpha)
29 lam <- 2
30
31 M <- dgamma(alpha-k, alpha, lam)/dgamma(alpha-k, k, lam-1)
32
33 xi <- seq(0, 10, length.out=100)
34
35 plot(xi, M*dgamma(xi, k, lam-1), type="l", lty=2, col="red",
36       xlab="x", ylab="Density")
37 lines(xi, dgamma(xi, alpha, lam), col="darkblue")
38 legend("topright", col=c("red", "darkblue"), lty=c(2, 1),
39       legend=c("Mg(x)_{scaled instrumental}", "f(x)_{target}"))
40
41
42 # Task 5
43
44 sampleGamma <- function(alpha, lam) {
45   k <- floor(alpha)
46   M <- dgamma(alpha-k, alpha, lam)/dgamma(alpha-k, k, lam-1)
47   while (TRUE) {
48     x <- sampleGammaInt(k, lam-1)
49     prob.of.accept <- dgamma(x, alpha, lam)/(M*dgamma(x, k, lam-1))
50     if (runif(1)<prob.of.accept)
51       return(x)
52   }
53 }
54
55 alpha <- 5.7
56 lam <- 2
57 x <- numeric(1000)
58 for (i in 1:1000)
59   x[i] <- sampleGamma(alpha, lam)
60
61 # Task 6
62
63 hist(x, freq=FALSE, col="grey")
64 xi <- seq(0, 10, length.out=100)
65 lines(xi, dgamma(xi, alpha, lam), lwd=2)
```