ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΣΤΑΤΙΣΤΙΚΗΣ

Μεταπτυχιακό Πρόγραμμα Συμπληρωματικής Ειδίκευσης
Εφαρμοσμένη Στατιστική

Μάθημα: Υπολογιστική Στατιστική
Computational Statistics

Ημερομηνία Παράδοσης εργασίας: 4/2/2020

Κωνσταντίνα Τσάμη

PAPER 1

Task 1

Το ζητούμενο της πρώτης άσκησης της εργασίας αυτής είναι να αναπτύξουμε δύο συναρτήσεις στην R που να υπολογίζουν τη δειγματική διακύμανση s^2. Η μία συνάρτηση θα την υπολογίζει μέσω του τύπου που την ορίζει και η δεύτερη βασίζεται στον τύπο υπολογισμού με το χέρι.

Για την πρώτη συνάρτηση(formation definition) εφαρμόζουμε τον τύπο της μεροληπτικής διακύμανσης $\mathbf{\Sigma(x_i-\mu)^2/n}$, ενώ για τη δεύτερη(hand calculation equation) τον τύπο της αμερόληπτης(unbiased) διακύμανσης $\mathbf{\Sigma(x_i-\mu)^2/(n-1)}$. Γνωρίζουμε ότι για μεγάλο δείγμα προσεγγίζει την ακρίβεια της αμερόληπτης διασποράς, για αυτό και στις δύο συναρτήσεις θα χρησιμοποιήσουμε μεγάλο μέγεθος δείγματος.

Η διαδικασία που ακολουθούμε και για τις δύο συναρτήσεις είναι παρόμοια.

Αρχικά θα παράγεται μέσα στη συνάρτηση τυχαίο δείγμα από κανονική κατανομή με μέσω τυπική απόκλιση και μέγεθος δείγματος τα οποία είναι τα “arguments” της συνάρτησης μας και στη συνέχεια θα υπολογίζεται το s^2 αυτού του δείγματος.

Για τον υπολογισμό λοιπόν της s^2 μέσω του “formation definition” ο κώδικας που εφαρμόζουμε είναι:

```
#function 1 using the formation definition

vardc<-function(n,mn=0,vr=1) {

        x<-rnorm(n,mn,vr)

        v<-0

        m<-mean(x)

        for (i in 1:n ) {

                v<-v+(x[i]-m)^2

                }

        var<-v/(n)

        return(var)

        }
```

Ενώ, για τον υπολογισμό της s^2 μέσω του "hand calculation equation" είναι:

```
#function 2 based on the hand calculation equation
varhc<-function(n,mn=0,vr=1) {
        x<-rnorm(n,mn,vr)
        v<-0
        m<-mean(x)
        for (i in 1:n) {
                v<-v+(x[i]-m)^2
                }
        var<-v/(n-1)
        return(var)
        }
```

Δοκιμάζοντας τις δύο συναρτήσεις για ένα μέγεθος δείγματος n=10000 και μία μεγάλη τιμή για μέσο όπως το 1000, για διαφορετικές τυπικές αποκλίσεις.

```
> x<-vardf(10000,1000,200)
> sqrt(x)
[1] 200.1545
> x<-varhc(10000,1000,200)
> sqrt(x)
[1] 202.0596
>
> x<-vardf(10000,1000,20)
> sqrt(x)
[1] 20.78012
> x<-varhc(10000,1000,20)
> sqrt(x)
[1] 19.641
```

Παρατηρούμε ότι για μεγαλύτερη διασπορά δείγματος η πρώτη συνάρτηση(formation definition) είναι πιο κοντά στην πληθυσμιακή-θεωρητική διακύμανση. Ενώ για δείγματα με μικρότερη διασπορά η δεύτερη συνάρτηση είναι σχετικά πιο ακριβής με ελάχιστη διαφορά.

Task 2

Η δεύτερη άσκηση ζητά να υπολογίσουμε τον εκτιμητή μέγιστης πιθανοφάνειας για την παράμετρο $θ_1$ της κατανομής Cauchy και σταθερό $θ_2$=1.

Για να το υλοποιήσουμε θα εφαρμόσουμε τον αλγόριθμο "Newton-Raphson" και συγκεκριμένα μία παραλλαγή του (Newton like method) την "quasi Newton-Raphson".

Αρχικά εισάγουμε τα δεδομένα μας.

```
x<-c(1.77,-0.23,2.76,3.80,3.47,56.75,-1.34,4.24,-2.44,3.29,3.71,

        -2.40,4.53,-0.07,-1.05,-13.87,-2.53,-1.75,0.27,43.21)
```

Στη συνέχεια θα δημιουργήσουμε τη συνάρτηση που υπολογίζει την log-likelihood της συνάρτησης πυκνότητας πιθανότητας της Cauchy με σταθερό $θ_2$=1.

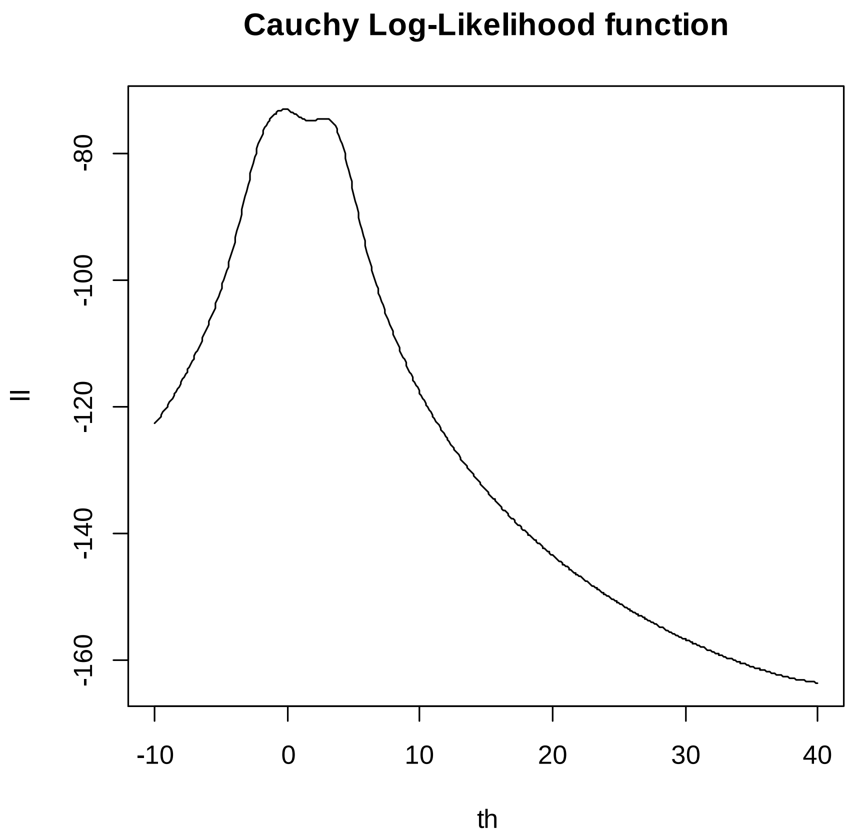
```
#log-likelihood function of Cauchy distribution

ell <- function(theta){

                sum(dcauchy(x, location = theta, scale = 1, log = TRUE))

                }
```

Σειρά έχει να εξετάσουμε την συμπεριφορά της λογαριθμικής πιθανοφάνειας μέσω του κατάλληλου γραφήματος.

```
#plot of the log-likelihood function:

th<-seq(-10,40,0.1)

ll<-numeric(length(th))

for (i in 1:length(th)) {

        ll[i]<-ell(th[i])

}

plot(th,ll,type='l',main="Cauchy Log-Likelihood function")
```

Όπως παρατηρούμε στο Διάγραμμα 1 πιο κάτω, η log-likelihood φαίνεται να παρουσιάζει ολικό μέγιστο για $θ_1$ γύρω στο 0, ενώ παρουσιάζει και ένα τοπικό μέγιστο για $θ_1$ κοντά στο 3.

Διάγραμμα 1- Log-Likelihood plot



Θα βρούμε τον MLE εκτιμητή του θ_1 δοκιμάζοντας διάφορα starting points στον αλγόριθμό μας.

#quasi NR

Διαβάζουμε τη βιβλιοθήκη για το πακέτο που θα χρησιμοποιήσουμε για τους υπολογισμούς των παραγόγων.

library(numDeriv)

Το -11 και 8 δεν είναι καλά σημεία εκκίνησης καθώς ο αλγόριθμός μας δε συγκλίνει. Δοκιμάζουμε όλα τα υπόλοιπα starting points καθώς επίσης και την διάμεσο των δεδομένων.

```
theta0<-c(-1,0,1.5,4,4.7,7,38,mean(x),median(x))  #starting value

theta<-numeric(length(theta0))

for (i in 1:length(theta0)) {

eps <- 1     #cvgce criterion

while (eps>1e-6){

        elldash <- grad(ell,theta0[i])   #first derivative
```

```
        ellddash <- hessian(ell,theta0[i])[1,1]   #second derivative
        theta[i] <- theta0[i] - elldash/ellddash
        eps <- abs(theta[i]-theta0[i])
        theta0[i] <- theta[i]
        }
}
```

```
cbind(starting.point=round(c(-1,0,1.5,4,4.7,7,38,mean(x),median(x)),1),theta)
```

```
      starting.point      theta
 [1,]           -1.0 -0.1922866
 [2,]            0.0 -0.1922866
 [3,]            1.5  1.7135868
 [4,]            4.0  2.8174722
 [5,]            4.7 -0.1922866
 [6,]            7.0 41.0408478
 [7,]           38.0 42.7953775
 [8,]            5.1 56.2533580
 [9,]            1.0  1.7135868
```

Παρατηρούμε ότι ο μέσος ($8^η$ σειρά) δεν είναι ένα καλό σημείο εκκίνησης καθώς απέχει πολύ από την εκτίμηση που μεγιστοποιεί την συνάρτησή μας. Αυτό συμβαίνει διότι στην περίπτωση της Cauchy οι παράμετροι δεν περιγράφουν τον μέσο και την τυπική απόκλιση όπως στην κανονική κατανομή. Δοκιμάσαμε σαν starting point και τη δειγματική διάμεσο η οποία δίνει καλύτερη εκτίμηση($9^η$ σειρά).

Σε γενικές γραμμές τα καλύτερα σημεία εκκίνησης είναι το -1,0 αλλά και το 4 ε.μ.π. ίσο με -0.19. Γενικά προτιμούμε να εισάγουμε σημεία εκκίνησης όσο γίνεται πιο κοντά στο μέγιστο.

PAPER 1

Task 1

Θα υπολογίσουμε τους mle εκτιμητές λ,α,κ της Γενικευμένης Γάμμα κατανομής.

Αρχικά περνάμε τα δεδομένα μας με τους χρόνους διάρκειας κύησης.

```
y<-c(2.1,4,2.6,1.5,2.5,4,2,
  3.4,4.1,3.6,4.7,2.5,4,2.7,
  4.25,5,3.6,4.7,3.4,5.25,2.75,
  5.6,5.5,6.4,7.2,4.2,6.1,3.4,
  6.4,5.7,6.8,7.25,5.9,6.5,4.2,
  7.3,6.5,7.5,8.1,6.25,6.9,4.3,
  8.5,7.25,7.5,8.5,7.3,7,4.9,
  8.75,7.3,8.25,9.2,7.5,8.45,6.25,
  8.9,7.5,8.5,9.5,7.8,9.25,7,
  9.5,8.2,10.4,10.7,8.3,10.1,9,
  9.75,8.5,10.75,11.5,8.3,10.2,9.25,
  10,9.75,14.25,10.25,12.75,10.7,
  10.4,11,14.5,12.9,14.6,
  10.4,11.2,14.3,
  16,15,
  19,16.5)
```

Στη συνέχεια θα δημιουργήσουμε τη συνάρτηση της αρνητικής λογαριθμικής πιθανοφάνειας με την παρακάτω παραμετροποίηση:

- a=ln(α)
- k=ln(κ)
- l=ln(λ)

#minus log likelihood function

```
negloglikgengamma<-function(thet,dat){
                              a<-thet[1]
                              k<-thet[2]
                              l<-thet[3]
                             n<-length(dat)
                              -(n*a+n*exp(k)*l-n*log(gamma(exp(k)))+(exp(a)*exp(k)-1)
                              *sum(log(dat))-exp(l)*sum(dat^exp(a)))
                              }
```

Θα χρησιμοποιήσουμε τη συνάρτηση *optim* της R η οποία εφαρμόζει τη μέθοδο Newton-Raphson για την εύρεση ελαχίστου. Ελαχιστοποιώντας την αρνητική λογαριθμική πιθανοφάνεια μεγιστοποιούμε την λογαριθμική πιθανοφάνεια και ως εκ τούτου την πιθανοφάνεια της Γενικευμένης Γάμμα.

Σαν σημεία εκκίνησης θα χρησιμοποιήσω τα (a,k,l)=(-100,-100,-100)

```
fitggamma<-optim(c(-100,-100,-100),negloglikgengamma,hessian=TRUE,method="BFGS",dat=y)
```

Κατασκευάζουμε τον πίνακα με τις εκτιμήσεις των παραμέτρων και τα τυπικά σφάλματα των εκτιμητών που μας έδωσε σαν αποτέλεσμα η *optim* .

```
ggammamle<-cbind(est=fitggamma$par, se=sqrt(diag(solve(fitggamma$hessian))))

ggammamle
```

```
  est            se
[1,]   1.652433 0.1184858
[2,]  -1.211966 0.2042144
[3,] -13.405832 1.7949182
```

- Η εκτίμηση της παραμέτρου a=ln(α) είναι 1.65 με τ.σ. ίσο με 0.11.
- Η εκτίμηση της παραμέτρου k=ln(κ) είναι -1.21 με τ.σ. ίσο με 0.2.
- Η εκτίμηση της παραμέτρου l=ln(λ) είναι -13.4 με τ.σ. ίσο με 1.79.

Παρατηρούμε ότι έχουμε μικρά τυπικά σφάλματα, κάτι πολύ θετικό.

Για τις εκτιμήσεις των α,κ,λ:

```
round(exp(ggammamle),4)
```

```
        est     se
[1,] 5.2197 1.1258
[2,] 0.2976 1.2266
[3,] 0.0000 6.0190
```

Τώρα, για να κατασκευάσουμε και ένα 95% διάστημα εμπιστοσύνης των εκτιμητών.

CI<-cbind(LCI=ggammamle[,1]-1.96*ggammamle[,2],UCI=ggammamle[,1]+1.96*ggammamle[,2])
exp(CI)

```
              LCI          UCI
[1,] 4.118386e+00 6.615429e+00
[2,] 1.978206e-01 4.477424e-01
[3,] 4.157909e-08 5.457168e-05
```

Task 2

Αρχικά θα φτιάξω την αρνητική λογαριθμική πιθανοφάνεια της γενικευμένης Γάμμα για fixed τιμές των k,l ίσες με τις mle εκτιμήσεις τους που βρήκαμε προηγουμένως.

```
#minus log likelihood function for alpha

negloglikgg<-function(alpha,dat) {

        a<-alpha
        k<-ggammamle[2,1]
        l<-ggammamle[3,1]
        n<-length(dat)
        -(n*a+n*exp(k)*l-n*log(gamma(exp(k)))+(exp(a)*exp(k)-1)
        *sum(log(dat))-exp(l)*sum(dat^exp(a)))
        }
```

Αυτό το κάναμε με σκοπό να χρησιμοποιήσουμε τη συνάρτηση αυτή ώστε σε κάθε νέο δείγμα που θα προσομοιώνουμε με τη συνάρτηση της γενικευμένης γάμμα με τις εκτιμήσεις των παραμέτρων που βρήκαμε στο προηγούμενο βήμα, να υπολογίζουμε εκ νέου ένα νέο εκτιμητή για το a στα προσομοιωμένα δεδομένα.

Για την προσομοίωση θα εφαρμόσουμε τη μέθοδο της αντιστοφής.

Διαβάζουμε τη βιβλιοθήκη για τη συνάρτηση που μας βοηθάει στην αντιστροφή των συναρτήσεων.

```
library(GoFKernel)
```

Ορίζω την αντίστροφη συνάρτηση που θα χρησιμοποιήσω.

```
F.x<-function(x) {integrate(gg.pdf,lower=0,x)$value}

f.x<-inverse(F.x,lower=0)
```

Ξεκινάω τη προσομοίωση υπολογίζοντας την εκτίμηση για τη παράμετρο a (thetastar) για κάθε ένα από τα 800 τυχαία δείγματα που θα παράγω από την κατανομή μου.

```
thetastar <- numeric(800)

thet<- 0

for (i in 1:800){

	#generating random samples from my distribution

	xstar<-numeric(95)

	u<-0

	for (j in 1:95) {

		u<-runif(1)

		xstar[j]<-f.x(u)

		}

	#ektimhsh gia to θ1

	thet<-optim(-100,negloglikgg,hessian=TRUE,method="BFGS",dat=xstar)

	thetastar[i]<-thet$par

	}
```

Για την εκτίμηση του bootstrap διαστήματος εμπιστοσύνης θα χρησιμοποιήσω τα ποσοστιαία σημεία της κατανομής της thetastar.

```
up<-quantile(thetastar,  probs = 0.975)

lr<-quantile(thetastar,  probs = 0.025)




CIBoot<-cbind(LCI=lr,UCI=up)  #0.47 polu ektos twn diasthmatwn bootstrap - oxi kalh prosarmogh

CIBoot

ggammamle[1,1]
```

Τα αποτελέσματα που έχω για το 95% διάστημα εμπιστοσύνης με την παραμετρική μέθοδο του bootstrap για το a=ln(α) είναι:

```
> CIBoot
          LCI    UCI
2.5% 1.253883 1.3097
```

Η εκτίμηση που είχαμε είναι:

```
> ggammamle[1,1]
     est
1.652433
```

Παρατηρώ ότι η εκτίμησή μου δεν είναι μέσα στο διάστημα εμπιστοσύνης της bootstrap. Επομένως, η Γενικευμένη Γάμμα δεν προσαρμόζεται καλά στα δεδομένα.

Task 3

Προσαρμόζω την αρνητική λογαριθμική πιθανοφάνεια της Εκθετικής κατανομής κρατώντας την παραμετροποίηση l=ln(λ).

```
minusellexp <- function(lambda,dat) {
 #exponential minus log-likelihood
 n <- length(dat)
 -n*lambda+exp(lambda)*sum(dat)
}
```

Βρίσκω τον εκτιμητή μέγιστης πιαθνοφάνειας για τον l.

```
fitexp <- optim(0.5, minusellexp, method='BFGS', dat=y, hessian=T)
fitexp
```

```
list(estim=fitexp$par,st.er.=sqrt(solve(fitexp$hessian)))
```

```
$estim
[1] -2.044224

$st.er.
          [,1]
[1,] 0.1025978
```

Η εκτίμηση του l είναι -2.04 με τυπικό σφάλμα 0.1

Το LR test έχει τιμή ελεγχοσυνάρτησης LR=-2log(l1-l2) όπου l1,l2 λογαριθμικές πιθανοφάνειες των ε.μ.π. εκτιμητών .

Στα δεδομένα μας η τιμή αυτή δίνεται μέσω της μέγιστης τιμής των log-lokelihood της γενικευμένης γάμμα και της εκθετικής που πρασαρμώσαμε.

```
lr.t<- -2*(fitggamma$value-fitexp$value)
lr.t
```

```
[1] 62.58248
```

Για τη Μέθοδο Monte Carlo και τον έλεγχο Υποθέσεων likelihood ratio θα προσομοιώσουμε δείγματα μέσω της μη παραμετρικής bootstrap παράγοντας 1000 τυχαία δείγμα από τα δεδομένα μας, με επανατοποθέτηση.

Για κάθε νέο δείγμα υπολογίζουμε εξ αρχής τις παραμέτρους ε.μ.π. των δύο πιθανοφανειών και στη συνέχεια την ελεγχοσυνάρτηση LR.

```
lrt <- numeric(1000)

for (r in 1:1000) {

x<-sample(y, replace=T)

fit1<-optim(c(-100,-100,-100),negloglikgengamma,method="BFGS",dat=x)

fit2<-optim(0.5,minusellexp,method="BFGS",dat=x)

l1<--fit1$value

l2<--fit2$value

lrt[r] <- -2*(l2-l1)

}
```

Συγκρίνοντας τα 1-α ποσοστιαία σημεία της X^2 κατανομής με αυτά της κατανομής της lrt έχουμε:

```
                80%      90%       95%      99%
simulated 64.013719 69.51460 75.809318 95.33950
chisq      3.218876  4.60517  5.991465  9.21034
```

Όπως βλέπουμε η lrt δεν προσαρμόζεται καλά στα ποσοστιαία σημεία της X^2 κατανομής με 2 βαθμούς ελευθερίας(προκύπτουν από τη διαφορά στον αριθμό των παραμέτρων των 2 πιθανοφανειών).

Υπολογίζουμε το p-value της Monte Carlo.

```
#Monte Carlo p value

(sum(lrt>lr.t) +sum(lrt<-lr.t) )/1000
```

```
[1] 0.08458248
```

Υπάρχουν επαρκείς ενδείξεις ότι μπορούμε να απορρίψουμε την μηδενική υπόθεση της μη ύπαρξης σημαντικών διαφορών ανάμεσα στις δύο πιθανοφάνειες σε επίπεδο α=10%. Επομένως η προσαρμογή της εκθετικής κατανομής προσαρμόζεται καλύτερα στα δεδομένα. Στο ίδιο συμπέρασμα οδηγούμαστε και ασυμπτωματικά (LR=62.58> $X^2_{0.95}$=5.99).

Task 4

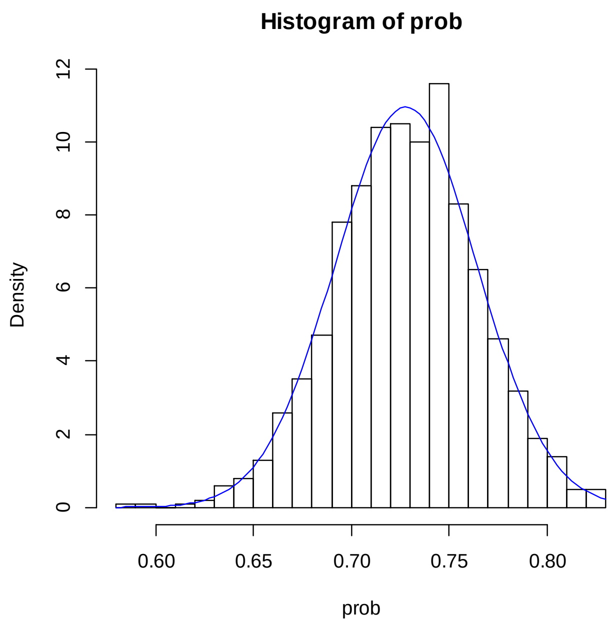
Για την εκτίμηση της πιθανότητας η διάρκεια κύησης να μην ξεπερνά τις 10 ώρες P(X<=10) θα προσομοιώσω δείγματα από την εκθετική κατανομή με την εκτίμηση της παραμέτρου που είχαμε. Σε κάθε νέο σετ δεδομένων θα εκτιμήσουμε από την αρχή την ε.μ.π. παράμετρο της εκθετικής και στη συνέχεια την πιθανότητα που ζητούμε να υπολογίσουμε.

```
n <- 95; lamda <- exp(fitexp$par)

prob <- numeric(1000)

for (r in 1:1000){

x <- rexp(n,lamda)

lambdastar<-optim(0.5, minusellexp, method='BFGS', dat=x, hessian=F)$par

fx<-function(x) {dexp(x,exp(lambdastar))}

prob[r] <- integrate(fx,0,10)$value

}
```

Από το ιστόγραμμα φαίνεται ότι η κατανομή της εκτίμησης της πιθανότητας προσεγγίζει τη κανονική και επομένως η τυπική απόκλιση θα μας δώσει και to standard error της εκτίμησης αυτής.

```
hist(prob,25,prob=T)

curve(dnorm(x,mean(prob),sd(prob)),col="blue",add=T)
```

Διάγραμμα 2: ιστόγραμμα κατανομής εκτίμησης της πιαθνότητας



Το τυπικό σφάλμα θα είναι

#se gia th pithanothta

sd(prob)

[1] 0.03635842

Η εκτίμηση της πιθανότητας στα δεδομένα μας θα είναι:

f<-function(x) {dexp(x,exp(fitexp$par))}

pr <- integrate(f,0,10)$value

pr

[1] 0.7260492

Η πιθανότητα η διάρκεια κύησης να μην ξεπερνά τις 10 ώρες είναι 72% με τυπικό σφάλμα 0.04.

PAPER 3

Task 1

Θα εφαρμόσουμε μέθοδο αντιστροφής για να προσομοιώσουμε την οριακή κατανομή πιθανότητας που μας ζητείται

```
library(GoFKernel)
```

Ορίζω την αντίστροφη συνάρτηση.

```
f<-function(t){4*(1-t)/(^3}

F<-function(x) {integrate(f,lower=0,x)$value}

F.inv<-inverse(F,0,1)
```

Τρέχω τη προσομοίωση

```
xsim<-numeric(100)

u<-0

for (r in 1:100) {

        u<-runif(1)

        xsim[r]<-F.inv(u)

        }
```

Task 2

Θα εφαρμόσουμε μέθοδο αντιστροφής για να προσομοιώσουμε την και τη δεσμευμένη σ.π.π. f(y/x) για δεδομένο x.

```
x<-0.05


f<-function(t){6*x*(1-t-x)/(1-t)^3}

F<-function(x) {integrate(f,lower=0,x)$value}
```

```
F.inv<-inverse(F,0,0.95)


z<-seq(0,0.95,0.01)

Fx<-numeric(length(z))

for (i in 1:length(z) ) {

        Fx[i]<-F(z[i])

        }

plot(z,Fx,type='l',main="plot of cdf")


#h f(x) einai "1-1" ara kai antistrepsimh

#mporw na xrhsimopoihsw methodo antistrofhs


ysim<-numeric(100)

u<-0

for (r in 1:100) {

        u<-runif(1)

        ysim[r]<-F.inv(u)

        }
```

Task 3

Για να προσομοιώσουμε από την από κοινού κατανομή f(x,y) θα χρησιμοποιήσουμε την δεσμευμένη και την οριακή σ.π.π.

f(xi,yi)=f(yi/xi)f(xi)

Για κάθε xi από τα προσομοιωμένα που έχουμε στην άσκηση 1 θα προσομοιώσουμε ένα yi από τη δεσμευμένη f(yi/xi) και έτσι θα φτιάξουμε τα ζεύγη (xi,yi) της από κοινού σ.π.π.

```
ysim<-numeric(100)
```

u<-0

u<-0

x<-0

for (r in 1:100) {

x<-simx[r]

f<-function(t){6*x*(1-t-x)/(1-t)^3}

F<-function(x) {integrate(f,lower=0,x)$value}

F.inv<-inverse(F,0,1-x)

u<-runif(1)

ysim[r]<-F.inv(u)

}

Τα προσομοιωμένα δείγματα θα είναι.

```
> ysim
  [1] 0.25657153 0.18858781 0.40341893
  [4] 0.34428373 0.52416215 0.28413545
  [7] 0.40995972 0.39664646 0.20071505
 [10] 0.29266410 0.09806468 0.49080025
 [13] 0.42764432 0.39465805 0.22858870
 [16] 0.40500185 0.29673073 0.27094994
 [19] 0.21409959 0.21730315 0.39414440
 [22] 0.38505229 0.15411813 0.33081147
 [25] 0.34335174 0.47535475 0.31293036
 [28] 0.34294495 0.18688316 0.40439238
 [31] 0.36752119 0.30534197 0.46557585
 [34] 0.37457578 0.47753220 0.18055995
 [37] 0.44083254 0.37991373 0.41257170
 [40] 0.32500980 0.23316776 0.27544408
 [43] 0.40506798 0.35098148 0.19119075
 [46] 0.30433999 0.42382864 0.37770301
 [49] 0.45504544 0.30081634 0.08240706
 [52] 0.42435443 0.43146274 0.39442426
 [55] 0.37594636 0.42551697 0.35541234
 [58] 0.30939815 0.36265904 0.26122277
 [61] 0.15386086 0.31234592 0.38729182
 [64] 0.24332821 0.24364501 0.54418321
 [67] 0.20018026 0.01718968 0.48719617
 [70] 0.25084871 0.24226657 0.13912957
 [73] 0.29953239 0.03064699 0.44667175
 [76] 0.27991735 0.37269096 0.41589335
 [79] 0.39969639 0.41940040 0.38750575
 [82] 0.43909857 0.37993818 0.40562992
```

```
 [85] 0.29496843 0.39518716 0.26194819
 [88] 0.18376913 0.37357969 0.39257271
 [91] 0.20054446 0.40689905 0.26716015
 [94] 0.38516698 0.38228143 0.47931303
 [97] 0.45048284 0.12619894 0.58489878
[100] 0.46377580
```

```
> xsim
  [1] 0.022771285 0.006496715 0.250418677 0.052160704
  [5] 0.352008021 0.147784778 0.344371326 0.228532415
  [9] 0.111236446 0.235140082 0.505453173 0.408519221
 [13] 0.340661785 0.058312134 0.143773909 0.086430341
 [17] 0.028665506 0.287174740 0.439267227 0.083469562
 [21] 0.064023430 0.084237038 0.054907510 0.077951013
 [25] 0.260125939 0.307700296 0.030912502 0.102454530
 [29] 0.244896842 0.460953636 0.146417036 0.098091147
 [33] 0.212965726 0.371159256 0.404636399 0.133274646
 [37] 0.179351828 0.036392343 0.164392322 0.413762950
 [41] 0.099827131 0.371593124 0.249413916 0.004267682
 [45] 0.202761930 0.044023716 0.129986918 0.241907320
 [49] 0.227468335 0.246521983 0.179662435 0.076162992
 [53] 0.238714027 0.138261155 0.056763230 0.311524147
 [57] 0.475804487 0.390772689 0.213189580 0.009578145
 [61] 0.264609745 0.324160399 0.068460511 0.138598834
 [65] 0.448188122 0.346749262 0.169430401 0.064014356
 [69] 0.287858301 0.498345847 0.006550481 0.126130278
 [73] 0.041424191 0.517730445 0.278776734 0.019674841
 [77] 0.036968196 0.124585920 0.196015610 0.214603844
 [81] 0.315909684 0.207748227 0.308528335 0.193083392
 [85] 0.009181012 0.471087155 0.058941350 0.114353412
 [89] 0.094158646 0.151483923 0.213641151 0.323312513
 [93] 0.219956476 0.009426819 0.020132601 0.407701963
 [97] 0.202119641 0.145896463 0.374756158 0.367415744
```

Υπολογίζω τον συντελεστή συσχέτισης για X,Y

```
cor(xsim,ysim)
[1] 0.1681808
sqrt(6)/6
[1] 0.4082483
```

Ο συντελεστής συσχέτισης είναι κοντά στο 0 το ίδιο και η θεωρητική τιμή του.

Οι δύο μεταβλητές δεν εμφανίζονται να είναι γραμμικά συσχετισμένες.