Εθνικο Μετσοβιο Πολυτεχνειο

Υπολογιστικη στατιστικη και στοχαστικη βελτιστοποιηση

Εξαμηνιαια εργασια

Ορφανουδάκης Φίλιππος Σκόβελεφ
phil.orfa@gmail.com
AM: 03400107
M.Sc. , ΕΔΕΜΜ ΔΠΜΣ

Αθήνα, Ιούνιος 2021

Περιεχόμενα

1. Άσκηση 1

1.1 Ερώτημα α

Στο ερώτημα αυτό μας ζητείται να εφαρμόσουμε την τεχνική της Monte Carlo Ολοκλήρωσης για να εκτιμήσουμε το παρακάτω ολοκλήρωμα

$$J = \int_{-\infty}^{+\infty} (x+a)^2 \phi(x) dx \tag{1.1}$$

, όπου

$$\phi(x) = (2\pi)^{-1/2} e^{\left(-\frac{x^2}{2}\right)} \tag{1.2}$$

Γνωρίζουμε από την εκφώνηση ότι η τιμή του συγκεκριμένου ολοκληρώματος είναι : $\mathbf{J = 1 + a^2}$.Την τιμή αυτή θα την εκμεταλλευτούμε για τον έλεγχο ορθότητας της μεθοδολογίας μας αλλά και των αποτελεσμάτων μας.

Πριν προχωρήσουμε στην υλοποίηση μας, να εξηγήσουμε συνοπτικά την ιδέα της Monte Carlo Ολοκλήρωσης. Αρχικά έχουμε σαν ζητούμενο τον υπολογισμό ενός ολοκληρώματος $\theta = \int f(x)\phi(x)dx$, όπου η $\phi(x)$ αποτελεί συνάρτηση πυκνότητας πιθανότητας. Αν παράξουμε τυχαίο δείγμα $x_1, \cdots, x_N$ από την $\phi(x)$, αποδεικνύεται (ζητείται σε επόμενο ερώτημα) ότι η

$$\hat{\theta} = \frac{1}{N} \sum_{n=1}^{N} f(x_i)$$

είναι συνεπής εκτιμήτρια της θ , δηλαδή έχουμε σύγκλιση κατά μέτρο στη θ.

Στη συγκεκριμένη άσκηση παρατηρούμε ότι η $\phi(x)$ είναι η συνάρτηση πυκνότητας πιθανότητας της τυποποιημένης κανονικής κατανομής, δηλαδή της $\mathcal{N}(0,\,1)$. Επομένως όπως μας ζητείται θα παραγάγουμε 100 και 1000 προσομοιωμένες τιμές και θα εφαρμόσουμε για $a = 1, 2, 3, 4$ την μέθοδο που περιγράφηκε. Ακολουθεί ο κώδικας που χρησιμοποιήθηκε :

```
mc_integr <-function(n){

  for(a in 0:4){

  x<-rnorm(n,0,1)
  y<-(x+a)^2
  ret<-sum(y)/n

  print((paste("N = ", n," a =",a,"Result = ",ret, "Real Value=",1+a^2)))
  }
}

```

```
mc_integr(100)
mc_integr(1000)
```

και τα αντίστοιχα αποτελέσματα παρουσιάζονται στον παρακάτω πίνακα:

N	a	Result	Real Value
100	0	0.978873946672843	1
100	1	1.66115492942222	2
100	2	4.43281149982029	5
100	3	10.4246774000289	10
100	4	16.1333330354816	17
1000	0	0.955702543944904	1
1000	1	2.02110145078168	2
1000	2	5.01255838652297	5
1000	3	10.2365570882809	10
1000	4	17.0235084761968	17

Παρατηρούμε ξεκάθαρα πως για $N = 1000$, δηλαδή για μεγαλύτερο πλήθος προσομοιωμένων τιμών τα αποτελέσματα πλησιάζουν με μεγαλύτερη ακρίβεια στην πραγματική τιμή.

1.2 Ερώτημα β

Σε αυτό το ερώτημα μας ζητείται να αποδείξουμε ότι ο παραπάνω Monte Carlo εκτιμητής,

$$\hat{\theta} = \frac{1}{N}\sum_{n=1}^{N} f(x_i)$$

, είναι αμερόληπτος και να βρεθεί η θεωρητική τυπική του απόκλιση.

Ο ορισμός αναφέρει ότι **ένας εκτιμητής λέγεται αμερόληπτος (unbiased), αν ισχύει**

$$E[\hat{\theta}] = \theta$$

Επομένως θέλουμε να δείξουμε ότι $E[\hat{\theta}] = J$.

$$E[\hat{\theta}] = N^{-1}\sum_{i=1}^{N} E\left(f\left(x_i\right)\right) = N^{-1}\cdot nE\left(f\left(x_i\right)\right) = E\left(f\left(x_i\right)\right) \tag{1.3}$$

Από (1.3) πρέπει να αποδείξουμε ότι $E\left(f\left(x_i\right)\right) = 1 + a^2$. Όμως τα x_i έχουν παραχθεί όπως δείξαμε από την $\phi(x)$ επομένως έχουμε τον υπολογισμό της $E\left(f\left(\phi(x)\right)\right)$ για την οποία ισχύει ότι:

$$E(f(\phi(x))) = \int \phi(x)f(x) = J$$

Στη συνέχεια θα υπολογίσουμε την θεωρητική τυπική του απόκλιση.

$$Var(\hat{\theta}) = n^{-1}Var(f(x)) = \frac{1}{n}Var\left(x^2 + 2ax + a^2\right) = \frac{1}{n}Var\left(x^2 + 2ax\right) =$$

$$= \frac{1}{n}\left(Var\left(x^2\right) + 4a^2Var(x) + 4aCov\left(x^2, x\right)\right) \tag{1.4}$$

Ισχύει ότι :

- $Var(x) = 1$, καθώς το x ακολουθεί την τυποποιημένη κανονική κατανομή
- $Cov\left(x^2, x\right) = 2\mu\sigma^2 = 0$ [1]
- $Var\left(x^2\right) = 2$, καθώς η τυποποιημένη κανονική κατανομή υψωμένη στο τετράγωνο ισούται με την chi squared κατανομή με 1 βαθμό ελευθερίας. Από θεωρία η διακύμανση της chi squared κατανομής ισούται με $2 * df$, όπου df οι βαθμοί ελευθερίας.

Από τα παραπάνω προκύπτει ότι $Var(\hat{\theta}) = \frac{4a^2+2}{n}$, συνεπώς η τυπική απόκλιση ισούται με $sd(\hat{\theta}) = \sqrt{\frac{4a^2+2}{n}}$. Από τα παραπάνω αποτελέσματα καταλήγουμε σε 2 συμπεράσματα. Πρώτον ότι ο εκτιμητής είναι συνεπής καθώς όταν $n \to \infty$ τότε $sd \to 0$. Τέλος παρατηρούμε και τον λόγο που την περίπτωση των 1000 προσομοιωμένων τιμών είχαμε καλύτερα αποτελέσματα από την περίπτωση των 100, καθώς ο εκτιμητής μας έχει μικρότερη τυπική απόκλιση.

1.3 Ερώτημα γ

Στο ερώτημα αυτό ζητείται πάλι να εκτιμήσουμε την τιμή του ολοκληρώματος

$$J = \int_{-\infty}^{+\infty} (x+a)^2 \phi(x) dx$$

αυτή τη φορά με τη μέθοδο της δειγματοληψίας σπουδαιότητας (importance sampling) με χρήση της συνάρτησης $g(x) = \phi(x-a)$.

Πριν πάμε να εφαρμόσουμε την μέθοδο, θα εξηγήσουμε τον τρόπο λειτουργίας καθώς και την υλοποίηση της.

Έχοντας την $g(x)$ θέτουμε $\psi = \frac{f\phi}{g}$ και συνεπώς έχουμε $J = \int \psi(x) g(x) dx$ και επομένως έχουμε ακριβώς την ίδια περίπτωση με το ερώτημα α, με την προϋπόθεση ότι θέλουμε να ισχύει ότι η $g(x)$ είναι συνάρτηση πυκνότητας πιθανότητας από την οποία να μπορούμε να παραγάγουμε δείγματα. Με αυτή την προϋπόθεση, θα εφαρμόσουμε ακριβώς τα ίδια βήματα και θα προσπαθήσουμε να εκτιμήσουμε το J. Τα βήματα είναι τα εξής :

- Παραγωγή τυχαίου δείγματος $x_1, \cdots, x_N$, από την κατανομή με συνάρτηση πυκνότητας πιθανότητας $g(x)$
- Εκτίμηση του J από την $\frac{1}{N} \sum_{n=1}^{N} \psi(x_i)$

Επομένως τελευταίο βήμα πριν πάμε να υλοποιήσουμε τον παραπάνω αλγόριθμο και επαναλάβουμε τα πειράματα του ερωτήματος α είναι να βρούμε ποια κατανομή ακολουθεί η $g(x)$.

$$g(x) = \phi(x-a) = (2\pi)^{-\frac{1}{2}} e^{-\frac{(x-a)^2}{2}}$$

, άρα η $g(x)$ είναι η συνάρτηση πυκνότητας πιθανότητας της $\mathcal{N}(a, 1)$.

Ακολουθεί ο κώδικας που χρησιμοποιήθηκε:

```
import_samp <-function(n){

  for(a in 1:5){

    x<-rnorm(n,a,1)
    y<-((x+a)^2)*exp(0.5*(a^2-2*a*x))
    ret<-sum(y)/n

    print((paste("N = ", n," a =",a,"Result = ",ret, "Real Value=",1+a^2)))
  }
}
import_samp(100)
import_samp(1000)
```

και τα αντίστοιχα αποτελέσματα παρουσιάζονται στον παρακάτω πίνακα:

N	a	Result	Real Value
100	0	1.02618606578447	1
100	1	2.09582094903574	2
100	2	3.9858166981791	5
100	3	13.2683858768216	10
100	4	36.5369552625995	17
1000	0	0.940416323090887	1
1000	1	1.94045368090429	2
1000	2	5.16539858767967	5
1000	3	6.87529638905978	10
1000	4	15.5590972480507	17

Παρατηρούμε πως τα αποτελέσματα μας για τις μεγάλες τιμές του α δεν είναι ιδιαίτερα κοντά στις πραγματικές τιμές. Θα προσπαθήσουμε να το αιτιολογήσουμε υπολογίζοντας την τυπική απόκλιση.

Στη συνέχεια θα αποδείξουμε ότι ο εκτιμητής μας είναι αμερόληπτος με τον ίδιο τρόπο που αποδείχθηκε και στο ερώτημα β.

$$E[\hat{\theta}] = N^{-1}\sum_{i=1}^{N} E\left(\psi\left(x_i\right)\right) = N^{-1}\cdot nE\left(\psi\left(x_i\right)\right) = E\left(\psi\left(x_i\right)\right) \tag{1.5}$$

Όμως τα x_i έχουν παραχθεί όπως δείξαμε από την $g(x)$ επομένως έχουμε τον υπολογισμό της $E\left(\psi\left(g(x)\right)\right)$ που όπως ξέρουμε

$$E(\psi((x))) = \int \psi(x)g(x) = \int \frac{f(x)\phi(x)}{g(x)}g(x) = J$$

Τέλος θα υπολογίσουμε την θεωρητική τυπική απόκλιση του εκτιμητή
$Var(\hat{\theta}) = n^{-1}Var(\psi(x)) = n^{-1}\left(E\left[\psi(x)^2\right] - E[\psi(x)]^2\right)$

- Εχουμε δείξει ότι $E[\psi(x)] = J \Rightarrow E[\psi(x)]^2 = (1+a^2)^2$
- $\psi^2(x) = (x+a)^4 e^{a^2-2ax}$, και γνωρίζουμε ότι το x έχει παραχθεί απο την $g(x) = \phi(x-a)$, επομένως $E\left[\psi(x)^2\right] = \int \frac{1}{\sqrt{2\pi}} e^{\frac{-(x-a)^2}{2}} e^{\left(a^2-2ax\right)}(x+a)^4 = \int (x+a)^4 \frac{1}{\sqrt{2n}} e^{-\frac{(x+a)^2}{2}} e^{a^2} = e^{a^2}\int (x+a)^4\phi(x+a)$. Βλέπουμε λοιπόν οτι προκύπτει η ροπή $4\eta\varsigma$ τάξης της $\mathcal{N}(-a,\,1)$ για την οποία γνωρίζουμε την τιμή της [2] ότι ισούται με $3\sigma^4 = 3$. Οπότε $E\left[\psi(x)^2\right] = 3e^{a^2}$

Τα τελικά αποτελέσματα είναι τα εξής :

$$Var(\hat{\theta}) = \frac{1}{n}\left(3e^{a^2} - 1 - 2a^2 - a^4\right)$$

,

$$sd(\hat{\theta}) = \sqrt{\frac{1}{n}\left(3e^{a^2} - 1 - 2a^2 - a^4\right)}$$

Παρατηρούμε ότι ο εκτιμητής είναι και συνεπής αφού η τυπική απόκλιση είναι ανάλογη του $\frac{1}{\sqrt{n}}$

N	a	MCI sd	IS sd
100	0	0.14142135623731	0.14142135623731
100	1	0.244948974278318	0.203834380941419
100	2	0.424264068711929	1.17811056399403
100	3	0.616441400296898	15.559322537542
100	4	0.812403840463596	516.31427020298
1000	0	0.0447213595499958	0.0447213595499958
1000	1	0.0774596669241483	0.0644580909225299
1000	2	0.134164078649987	0.372551271772669
1000	3	0.194935886896179	4.92028980678234
1000	4	0.256904651573303	163.272908228903

Ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

```
comp <-function(n){
  for(a in 0:4){
    sdmc<- sqrt((4*a^{2}+2)/n)
    sdimp<-sqrt((3*exp(a^2) - 1 - 2*(a^2) - a^4)/n)
    print((paste("N = ", n," a =",a,"MC = ",sdmc, "IM=",sdimp)))
  }
}

comp(100)
comp(1000)
```

Συγκρίνοντας τις τυπικές αποκλίσεις, βλέπουμε ότι για $\alpha = 0, 1$ έχουμε κοντινές ή πανομοιότυπες τιμές και αυτό αντικατοπτρίζεται και στα αποτελέσματα καθώς είναι αρκετά κοντά οι τιμές με τις πραγματικές. Για $\alpha = 2, 3, 4$ όμως η τεχνική της δειγματοληψίας σπουδαιότητας δημιουργεί εκτιμητή με μεγάλη τυπική απόκλιση σε αντίθεση με την τεχνική της Monte Carlo Ολοκλήρωσης κάτι το οποίο είναι πολύ έντονο και στα αποτελέσματα μας.

1.4 Ερώτημα δ

Στο ερώτημα αυτό μας ζητείται να χρησιμοποιήσουμε την τεχνική bootstrap για να εκτιμήσουμε το τυπικό σφάλμα του εκτιμητή του ερωτήματος α.

Σε αυτό το σημείο θα περιγράψουμε τα βήματα που θα ακολουθήσουμε για να φτάσουμε στο ζητούμενο.

- Η τεχνική Bootstrap χρησιμοποιείται για την παραγωγή διαφορετικών δειγμάτων, μέσα από ένα συγκεκριμένο δείγμα. Αυτό πραγματοποιείται με την επανάθεση, δηλαδή αν έχουμε ένα αρχικό δείγμα μεγέθους N , [$x_1, \cdots, x_N$] και επιλέξουμε να δημιουργήσουμε B δείγματα τότε έχουμε [$x_{11}, \cdots, x_{1N}$] , [$x_{21}, \cdots, x_{2N}$],...,[$x_{B1}, \cdots, x_{BN}$] τα οποία έχουν δημιουργηθεί με τυχαία δειγματοληψία απο το αρχικό δείγμα, η οποία μπορεί να επιλέξει και ένα στοιχείο που έχει ήδη επιλεχθεί.

- Από την στιγμή που μας ζητείτε ο εκτιμητής του ερωτήματος α , η παραγωγή του αρχικού δείγματος θα γίνει μέσω της $\mathcal{N}(0, 1)$.

- Για κάθε παραγόμενο δείγμα (Bootstrap) θα εφαρμόσουμε τον εκτιμητή μας για να πάρουμε την τιμή του και θα την αποθηκεύσουμε

- Τέλος το τυπικό σφάλμα που θα υπολογιστεί μέσα από αυτή τη διαδικασία, ακολουθεί τον συγκεκριμένο τύπο

$$\sqrt{\frac{1}{B-1}\sum_{i=1}^{B}(Boot\,(Xi) - \overline{Boot})^2}$$

. Παρόλα αυτά σύμφωνα με την βιβλιογραφία το bootstrap standard error ισούται με το standard deviation των bootstrap δειγμάτων [8] , συνεπώς θα συγκρίνουμε με την θεωρητική τυπική απόκλιση που υπολογίσαμε στο ερώτημα β.

- Η μοναδική ελευθερία που μας δίνεται σε αυτό το ερώτημα είναι η επιλογή του μεγέθους B , δηλαδή το πλήθος των δειγμάτων που θα δημιουργήσουμε μέσω Bootstrap για να έχουμε μια καλή προσέγγιση. Σύμφωνα με τον Bradley Efron και τον Robert J. Tibshirani στο βιβλίο που εισήγαγαν την έννοια του Bootstrap Sampling[8] προτείνουν, με βάση την εμπειρία τους ότι σπάνια πάνω από $B = 200$ θα είναι αναγκαίο για μια καλή προσέγγιση. Συνεπώς στα πλαίσια του ερωτήματος θα δοκιμάσουμε για $B = 50, 100, 200, 500$.

Ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

```
bootstrap <- function(b){
  boot<-vector()

  x<-rnorm(1000,0,1)

  for(i in 1:b){
    sel<-sample(x,1000,replace=TRUE)
    y<-(sel+4)^2
    ret<-sum(y)/1000
    boot<-append(boot,ret)
  }
  mean_boot<-sum(boot)/b
  se_boot<-sqrt((1/(b-1))*(sum((boot-mean_boot)^2)))
  theor_var<- (4*4^{2}+2)/1000
  theor_sd <-sqrt(theor_var)
  theor_se <- theor_sd/sqrt(1000)
  print(paste("Bootstrap SE = ", se_boot , "Theoretical SD",theor_sd))
}

bootstrap(50)
bootstrap(100)
bootstrap(200)
bootstrap(500)
```

και τα αποτελέσματα είναι τα παρακάτω:

B	Bootstrap SE	Theoretical SD
50	0.257705784389064	0.256904651573303
100	0.247626402801696	0.256904651573303
200	0.253601511206696	0.256904651573303
500	0.261795894016343	0.256904651573303

Παρατηρούμε πολύ κοντινές τιμές στα αποτελέσματα μας.

2. Άσκηση 2

2.1 Ερώτημα α

Μας ζητείται να προσομοιώσουμε 1000 τιμές από την συνάρτηση πυκνότητας πιθανότητας

$$f(x) = \frac{1}{e^3 - 1} e^x, x \in [0, 3]$$

με χρήση της μεθόδου αντιστροφής. Αρχικά θα αριθμήσουμε τα βήματα της μεθόδου και στη συνέχει θα τα εφαρμόσουμε.

- Υπολογίζουμε την συνάρτηση κατανομής $F(x)$ και το πεδίο τιμών της $[a_1, a_2]$.
- Ελέγχουμε αν είναι αντιστρέψιμη και αν είναι υπολογίζουμε την αντίστροφη της $F^{-1}(x)$ και το πεδίο ορισμού της $[b_1, b_2]$.
- Ορίζουμε ως U την ομοιόμορφη κατανομή $U \sim [b_1, b_2]$.
- Η συνάρτηση $F^{-1}(U)$ έχει την ζητούμενη κατανομή, δηλαδή έχει παραχθεί από την συνάρτηση πυκνότητας πιθανότητας $f(x)$

Επομένως:

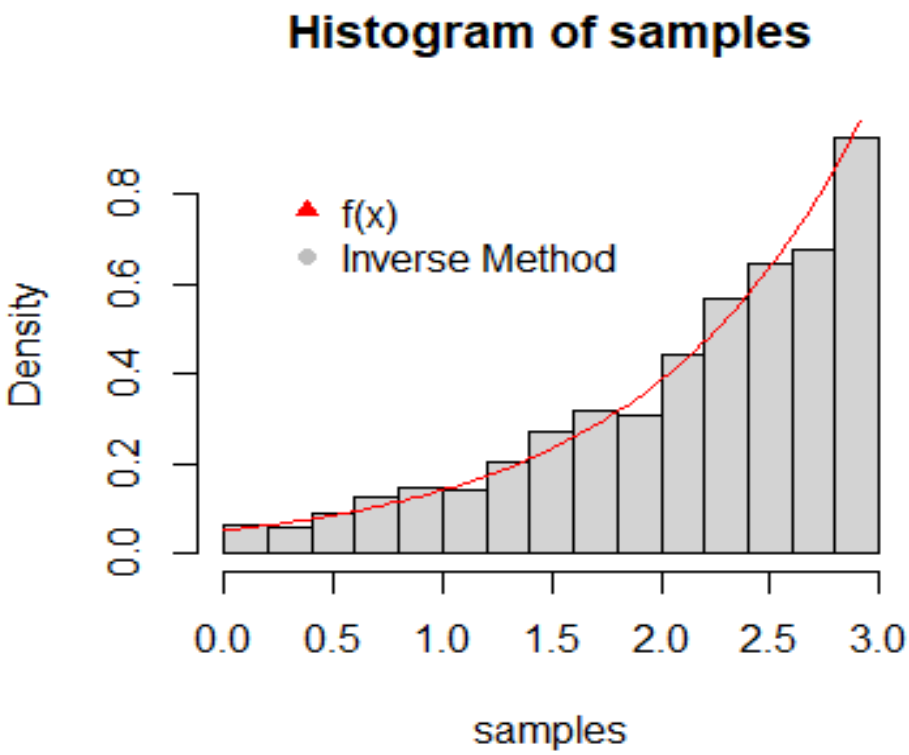
- $F(x) = \int_{-\infty}^{x} f(t)dt = \int_{0}^{x} f(t)dt = \frac{e^x - 1}{e^3 - 1}$, εύκολα βλέπουμε ότι είναι γνησίως αύξουσα επομένως το πεδίο τιμών της θα είναι το $[F(0), F(3)] = [0, 1]$
- Αφού είναι γνησίως αύξουσα θα είναι και αντιστρέψιμη με πεδίο ορισμού το $[0, 1]$ και ο τύπος της θα είναι $F^{-1}(x) = \ln(x(e^3 - 1) + 1)$
- Ορίζουμε ως U την ομοιόμορφη κατανομή U $[0, 1]$.
- Παράγουμε 1000 τιμές από την U και τις εφαρμόζουμε στην $F^{-1}(x)$ και παίρνουμε το ιστόγραμμα της. Παράλληλα σχεδιάζουμε και την $f(x)$ και συγκρίνουμε τα αποτελέσματα μας

Ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

```
inverse <-function(){

  U<-runif(1000,0,1)
  samples<-log((exp(3)-1)*U +1)

  hist(samples,
       prob = TRUE)
  curve(exp(x)/(exp(3)-1),add=TRUE,col = "red")
```

```
  legend("topleft",
         legend = c("f(x)", "Inverse Method"),
         col = c('red',
                 'grey'),
         pch = c(17,19),
         bty = "n",
         text.col = "black",
         horiz = F ,
         inset = c(0.1, 0.1))
}

inverse()

inverse()
```

και τα αποτελέσματα που πήραμε τα οποία είναι αρκετά ενθαρρυντικά είναι τα εξής (φαίνεται το διάγραμμα της $f(x)$ ως η κόκκινη καμπύλη):



2.2 Ερώτημα β

Σε αυτό το ερώτημα έχουμε το ίδιο ζητούμενο με το προηγούμενο ερώτημα αλλά αυτή τη φορά θα χρησιμοποιήσουμε την μέθοδο απόρριψης. Θα αναλύσουμε τα βήματα της μεθόδου, στη συνέχεια θα τα εφαρμόσουμε στη δικιά μας περίπτωση και τέλος θα παρουσιάσουμε τον κώδικα.

- Αντί να παράξουμε μια τιμή απο την συνάρτηση πυκνότητας πιθανότητας $f(x)$ παράγουμε απο μια άλλη συνάρτηση πυκνότητας πιθανότητας $g(x)$
- Θέλουμε να ισχύει ότι $f \leq M \cdot g$, για αυτό το λόγο θέτουμε ως $M = \max_y \frac{f(y)}{g(y)}$
- Παράγουμε απο την $g(x)$ ένα σημείο y και θέτουμε ως $K = \frac{f(y)}{Mg(y)}$

- Παράγουμε απο την $Uniform(0,1)$ ένα σημείο u
- Αν $u \leq K$ τότε αποδεχόμαστε το σημείο y και το αποθηκεύουμε , διαφορετικά επαναλαμβάνουμε τη διαδικασία από το τρίτο bullet.

Στην συγκεκριμένη περίπτωση έχουμε :

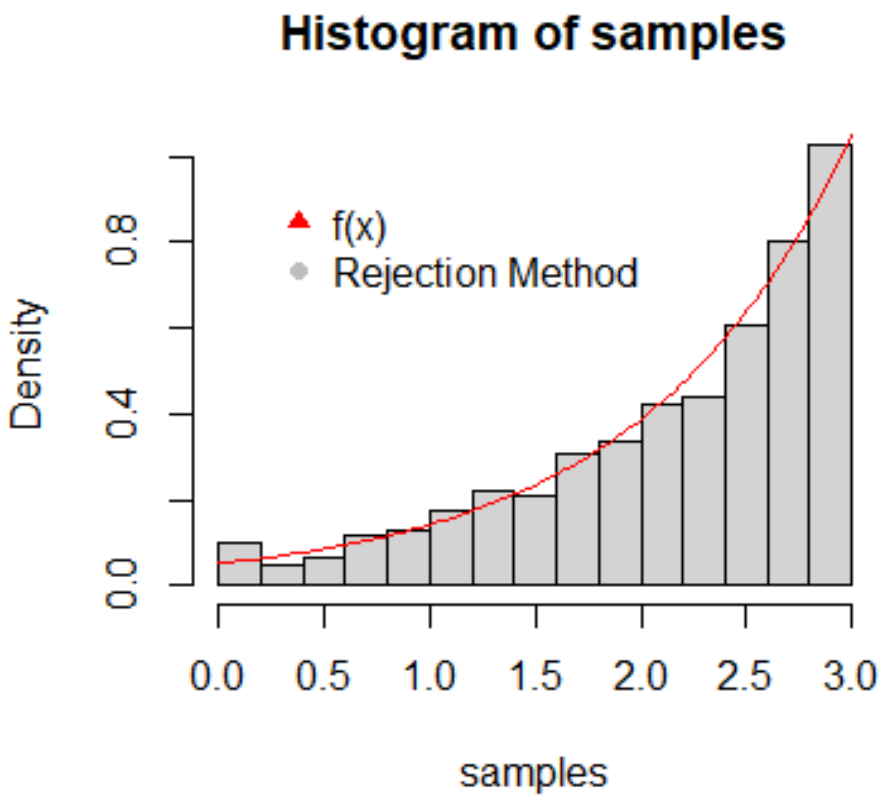
- Επιλέγουμε σαν συνάρτηση πυκνότητας πιθανότητας $g(x)$ την ομοιόμορφη $U \sim U[0,3]$. Φυσικά χρησιμοποιούμε το προηγούμενο ερώτημα σαν γνώση, έτσι ώστε να επιλέξουμε τη συγκεκριμένη συνάρτηση, καθώς ξέρουμε την μορφή της $f(x)$
- Γνωρίζουμε ότι $g(x) = \frac{1}{3-0}$, όπως επίσης ότι η $f(x)$ είναι γνησίως αύξουσα, άρα η μέγιστη τιμή της είναι η $f(3)$. Επομένως το M προκύπτει ως $M = 3 * f(3) = 3.157187$
- Παράγουμε από την $g(x)$ ένα σημείο y και θέτουμε ως $K = \frac{3*f(y)}{3.157187}$
- Παράγουμε από την $Uniform(0,1)$ ένα σημείο u
- Εξετάζουμε αν $u \leq K$ για α αποθηκεύσουμε το σημείο y
- Θέτουμε μια μεταβλητή για να μετράμε πόσα σημεία έχουν γίνει αποδεκτά και πόσα σημεία έχουν απορριφθεί, η συνθήκη τερματισμού είναι: όταν παρατηρηθούν 1000 αποδεκτά σημεία.

Ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

```
phi_x<-function(x){
  return (exp(x)/(exp(3)-1))
}

M_uni<-3*exp(3)/(exp(3)-1)
print(M_uni)

rejection_uni<-function(){
  acc<-0
  reject<-0
  samples<-vector()
  while(acc<1000){
    y<-runif(1,0,3)
    u<-runif(1)
    paron_uni<-M_uni/3
    if((phi_x(y)/paron_uni)>=u){
      acc=acc+1
      samples<-append(samples,y)
    }
    else{
      reject<-reject+1
    }
  }
  print(reject)
  hist(samples,prob = TRUE)
  curve(exp(x)/(exp(3)-1),add=TRUE,col = "red")
  legend("topleft",
         legend = c("f(x)", "Rejection Method"),
         col = c('red',
                 'grey'),
         pch = c(17,19),
         bty = "n",
         text.col = "black",
         horiz = F ,
```

```
        inset = c(0.1, 0.1))

}

rejection_uni()
```

Τα αποτελέσματα μας είναι ότι έγιναν αποδεκτά **1000** σημεία και απορρίφθηκαν **2136** σημεία. Το ιστόγραμμα που παράχθηκε είναι το εξής και φαίνεται και το διάγραμμα της $f(x)$ ως η κόκκινη καμπύλη:



2.3 Ερώτημα γ

Σε αυτό το ερώτημα η εκτίμηση της $f(x)$ θα γίνει με την βοήθεια της μεθόδου πυρήνων. Η μέθοδος αυτή διαθέτει μια υπερπαράμετρο η οποία είναι το πλάτος (bandwidth) h, και καλούμαστε να ελαχιστοποιήσουμε το MISE (Mean Integrated squared error) μεγιστοποιώντας την cross validated πιθανοφάνεια με την τεχνική του "leave one out". Θα εξηγήσουμε τις έννοιες που αναφέρθηκαν ξεχωριστά και θα αναλύσουμε πως εφαρμόζονται στην δικιά μας περίπτωση.

Αρχικά η μέθοδος των πυρήνων τοποθετεί μια συνάρτηση - πυρήνα γύρω από κάθε σημείο x_i, η οποία καθορίζει μια συμμετρική καμπύλη με πλάτος h. Αυτή η καμπύλη ποσοτικοποιεί την βαρύτητα και το πόσο κοντά στο κέντρο είναι τα σημεία που μας ενδιαφέρουν. Με αυτό τον τρόπο έχουμε ένα άθροισμα της βαρύτητας όλων των σημείων για το διάστημα h. Αν πραγματοποιηθεί για κάθε σημείο η συγκεκγιμένη διαδικασία, τότε αυτό το άθροισμα μετατρέπεται σε ένα ιστόγραμμα από το οποίο μπορούμε να πάρουμε την προσομοιωμένη καμπύλη και εκτίμηση της συνάρτηση που αναζητούμε. Στη δικιά μας περίπτωση η συνάρτηση πυρήνας είναι ο πυρήνας **Epanechnikov** με τύπο :

$$K(u) = \frac{3}{4}\left(1 - u^2\right), |u| \leq 1$$

.

Με τη χρήση αυτού του πυρήνα η εκτίμηση της συνάρτησης $f(x)$, έχοντας παραγάγει 100 τιμές με την μέθοδο αναστροφής είναι η εξής :

$$\hat{f}(x) = \frac{1}{N \cdot h} \cdot \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right), N = 100$$

Συνεπώς, γνωρίζουμε τα πάντα εκτός από το h. Ένα αρκετά μεγάλο h θα αναγκάσει την εκτίμηση μας να μεγαλώσει την μεροληψία της και να έχουμε λεία καμπύλη , μην ανταποκρίνοντας στις πιθανές κορυφές. Ενώ ένα μικρό h θα αναγκάσει το σύστημα μας να μεγαλώσει τη διασπορά του και να έχουμε μια καμπύλη που να διακριτοποείται, δηλαδή να έχει πολλές κορυφές. Συνεπώς η επιλογή του h είναι πολύ σημαντική και ο καθορισμός της θα γίνει μέσα από την cross-validated πιθανοφάνεια με την τεχνική "leave one out".

Η συνάρτηση πιθανοφάνειας είναι η εξής :

$$L(h) = \prod_{i=1}^{N} \hat{f}_h\left(x_i\right)$$

Αν υποθέσουμε ότι αφαιρούμε από το δείγμα 100 σημείων μας, το i-οστό σημείο τότε ορίζεται η εξής εκτίμηση :

$$\hat{f}_{h,-i}(x) = \frac{1}{(N-1)h} \cdot \sum_{J=1, J\neq i}^{N} K\left(\frac{x - x_J}{h}\right)$$

και αντίστοιχα η "leave one out" πιθανοφάνεια :

$$L(h) = \prod_{i=1}^{N} \hat{f}_{h,-i}\left(x_i\right)$$

Αυτή τη συνάρτηση πιθανοφάνειας μπορούμε να την χρησιμοποιήσουμε στο cross validation μας και να την υπολογίσουμε για ένα εύρος υποψήφιων πλατών h. Αυτή η "leave one out" cross validation likelihood function θα μας δώσει το h που ψάχνουμε, το οποίο θα αντιστοιχεί στην μέγιστη τιμή της. Αναλυτικά ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

```
set.seed(4)
inverse_3 <-function(){

  U<-runif(100,0,1)
  samples<-log((exp(3)-1)*U +1)

  return (samples)
}

sim<-inverse_3()


epanechnikov<-function(x){

  if(abs(x)<1){
    ret<-3*(1-x^2)/4
  }
  else{
    ret<-0
  }
  return (ret)
}

leave_one_out_f<-function(h,curr,i){
  N<-99
  pol<-1/(99*h)
  vect<-sim[-i]
  ep<-0
```

```
  for(j in 1:99){
    val<-(curr-vect[j])/h
    epan<-epanechnikov(val)
    ep<-ep+epan
  }
  return (ep*pol)
}

calculate_likel<-function(h)
{
  temp<-1
  for (j in 1:100){


      temp<-temp*leave_one_out_f(h,sim[j],j)

  }

  return(temp)
}


max_likelihood<-function(){
  h<-seq(0.001, 0.5, by = 0.001)
  max<-0
  h_opt<-0
  for(i in 1:length(h)){
    print(i)
    res<-calculate_likel(h[i])
    if (res>max){
      max<-res
      h_opt<-h[i]
    }
  }

  return (h_opt)

}

h_final<-max_likelihood()
```

Δοκιμάζοντας διάφορα υποψήφια διαστήματα τιμών για το h, καταλήξαμε εμπειρικά πως το καλύτερο h βρίσκεται στο διάστημα $(0.001, 0.5)$, για αυτό το λόγο παραγάγαμε τιμές ομοιόμορφα σε αυτό το διάστημα, με απόσταση 0.001 μεταξύ τους. Το αποτέλεσμα που μας έδωσε ο παραπάνω κώδικας είναι $\mathbf{h = 0.149}$ και η τελική μας συνάρτηση εκτίμηση είναι η εξής :

$$\hat{f}(x) = \frac{1}{N \cdot h} \cdot \sum_{i=1}^{N} K\left(\frac{x - x_i}{0.149}\right), N = 100$$

Για να την αναπαραστήσουμε , την υλοποιήσαμε αρχικά , στη συνέχεια παραγάγαμε ομοιόμορφα τιμές στο διάστημα $[0, 3]$ με απόσταση 0.001 και τέλος αναπαραστήσαμε τα σημεία στο παρακάτω διάγραμμα.

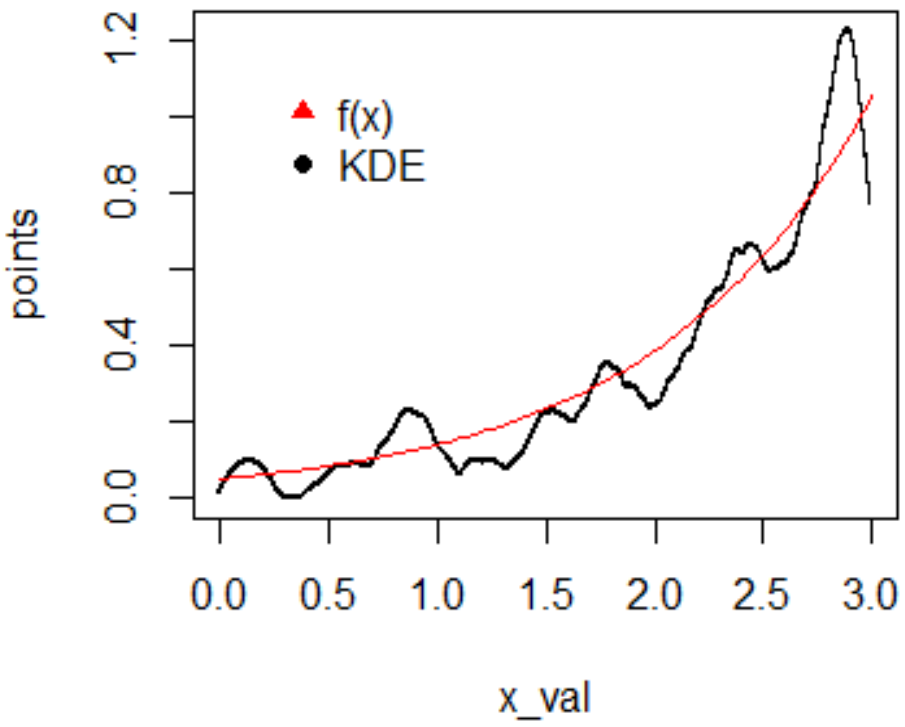


Figure 2.1: h=0.149

Παρακάτω παρατίθεται ο κώδικας που χρησιμοποιήθηκε :

```
1
2  f_hat<-function(x){
3    par<-1/(100*h_final)
4    sum<-0
5    for(i in 1:100){
6      val<-(x-sim[i])/h_final
7      ep<-epanechnikov(val)
8      sum<-sum+ep
9    }
10   return (par*sum)
11 }
12
13 x_val<-seq(0,3,0.001)
14
15 points<-vector()
16
17 for(i in 1:length(x_val)){
18   points<-append(points,f_hat(x_val[i]))
19 }
20 plot(x_val,points,cex=0.1)
21 curve(exp(x)/(exp(3)-1),add=TRUE,col = "red")
22 legend("topleft",
23        legend = c("f(x)", "KDE"),
24        col = c('red',
25                'black'),
26        pch = c(17,19),
27        bty = "n",
28        text.col = "black",
29        horiz = F ,
30        inset = c(0.1, 0.1))
```

Για να έχουμε καλύτερη εποπτεία των αποτελεσμάτων μας θα παράγουμε το διάγραμμα της $\hat{f}(x)$ για $h = 0.3$ και $h = 0.05$

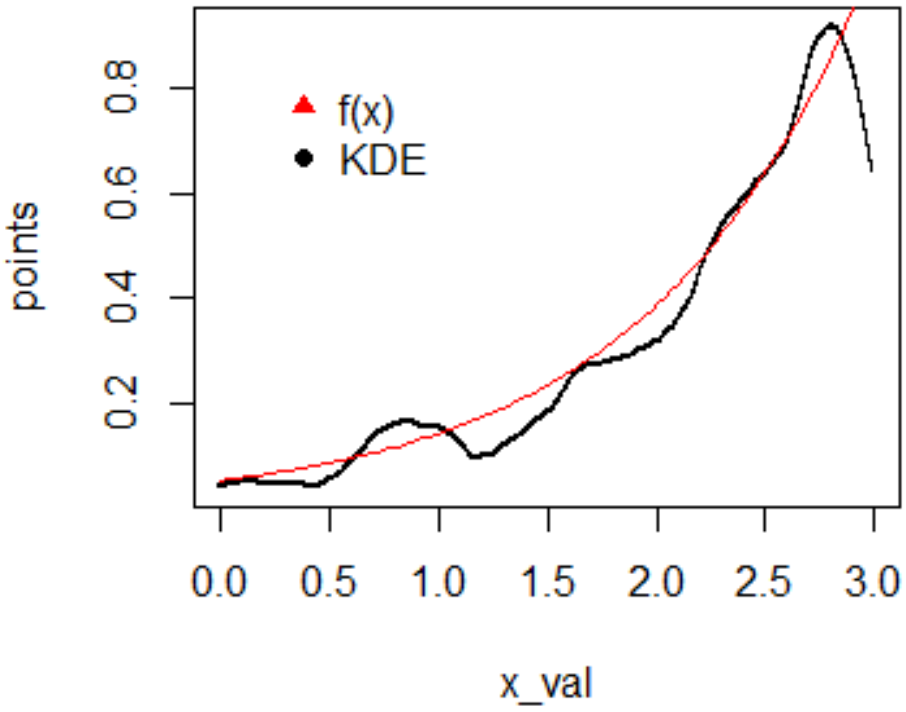


Figure 2.2: h=0.3

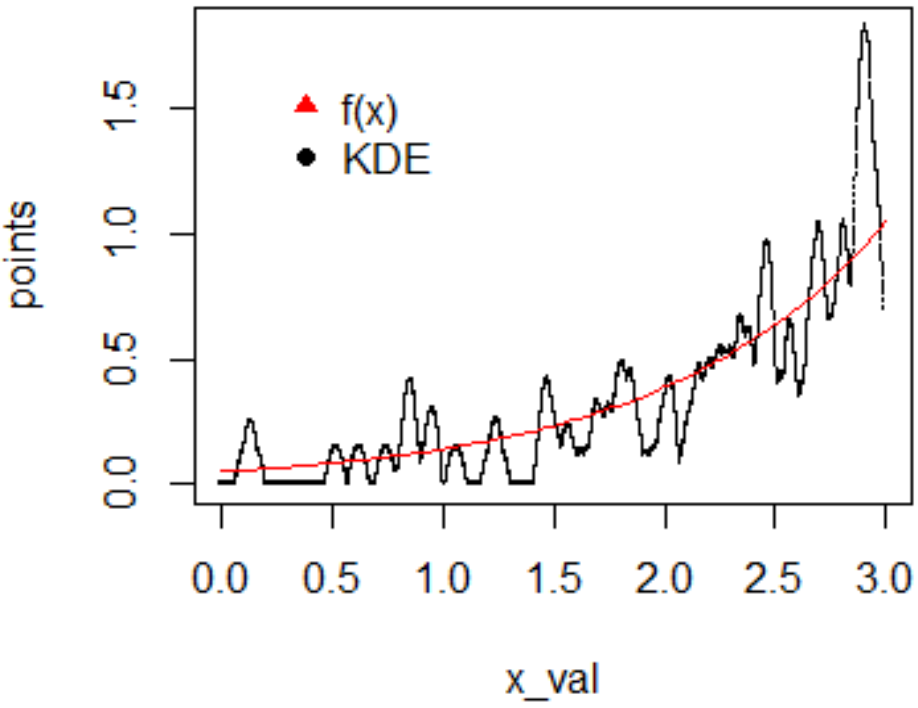


Figure 2.3: h=0.005

Οπτικά μπορούμε να σχολιάσουμε ότι η εκτίμηση για $h = 0.3$ φαίνεται αρκετά ικανοποιητική και πλησιάζει αρκετά και ίσως και περισσότερο από την $h = 0.149$ εκτίμηση, την συνάρτηση $f(x)$.

2.4 Ερώτημα δ

Στην ενότητα αυτή θέλουμε να ελέγχξουμε την υπόθεση ότι η κατανομή $f(x)$ έχει μέση τιμή 2. Αυτήν την υπόθεση την ορίζουμε ως μηδενική υπόθεση. Η υπόθεση που υποστηρίζει το αντίθετο την ορίζουμε ως υπόθεση 1.

- **H0** : $\mu = 2$
- **H1** : $\mu \neq 2$

Για τον έλεγχο των παραπάνω υποθέσεων θα εφαρμόσουμε 2 τεχνικές. Πρώτον θα υπολογίσουμε το p-value της μηδενικής υπόθεσης και δεύτερον με την εφαρμογή ποσοστιαίων σημείων 95% διαστήματος εμπιστοσύνης.

Το p-value υπολογίζει την πιθανότητα μια υπόθεση ή αλλιώς ένα φαινόμενο που εξετάζουμε να είναι απόρροια τυχαιότητας. Στη συγκεκριμένη περίπτωση εξετάζουμε το φαινόμενο : "η μέση τιμή να ισούται με 2" . Όσο πιο μικρή είναι η τιμή τόσο μεγαλύτερη είναι η πιθανότητα η μέση τιμή του δείγματος που εξετάζουμε να πάρει τιμή πιο ακραία απ' ότι εκείνη που ορίζεται από την μηδενική υπόθεση. Αρκετά συχνά στη βιβλιογραφία χρησιμοποιείται σαν threshold στο p-value η τιμή 0.05 [7], αυτή την τιμή θα χρησιμοποιήσουμε και εμείς. Δηλαδή αν το p-value προκύψει μικρότερο από 0.05 απορρίπτεται το φαινόμενο, καθώς θεωρούμε ότι αν παρατηρηθεί οφείλεται σε τυχαιότητα. Αντιθέτως αν η τιμή που προκύψει είναι μεγαλύτερη ή ίση αναφέρουμε πως δεν απορρίπτουμε το φαινόμενο και την μηδενική υπόθεση H0, αλλά δεν μπορούμε και με βεβαιότητα να το αποδεχτούμε.

Για την περίπτωση μας η μηδενική υπόθεση μπορεί να εκφραστεί μαθηματικά με την ελεγχοσυνάρτηση

$$T = |\bar{x} - 2|$$

Στα δεδομένα της άσκησης έχουμε 10 τιμές που έχουν παραχθεί από την $f(x)$ με την μέθοδο αντιστροφής. Με την χρήση Bootstrap θα παραγάγουμε αρκετά ακόμα δείγματα, τα οποια θα χρησιμοποιηθούν για να εξετάζουμε κάθε φορά αν λαμβάνουμε πιο ακραίες τιμές από το δείγμα που θέλουμε να ελέγξουμε. Από θεωρία γνωρίζουμε ότι τα δείγματα αυτά πρέπει να έχουν παραχθεί από την μηδενική υπόθεση. Επομένως τα 10 αρχικά σημεία τα μετατρέπουμε ώστε να έχουν μέση τιμή ίση με 2 και ύστερα εφαρμόζουμε Bootstrap. Στη συνέχεια σε κάθε ένα από τα 1000 δείγματα υπολογίζουμε την μέση τιμή και την απόσταση της από το 2. Αν αυτή η απόσταση είναι μεγαλύτερη από την απόσταση των 10 αρχικών σημείων τότε θεωρούμε ότι πρέπει να αυξηθεί η πιθανότητα να ισχύει η μηδενική υπόθεση, καθώς για ένα "τυχαίο" δείγμα για το οποίο ισχύει η H0, εντοπίσαμε πιο ακραία τιμή απ' ότι στο φαινόμενο που εξετάζουμε. Πιο συγκεκριμένα ο τύπος για τον υπολογισμό του p-value είναι ο εξής :

$$p - value = \frac{m+1}{B+1}$$

όπου **B : πληθος Bootstrap δειγμάτων** και **m: πλήθος δειγμάτων με** $T_i > T$. Όλα τα παραπάνω υλοποιήθηκαν με τον εξής κώδικα :

```
set.seed(90)
inverse_4 <-function(){

  U<-runif(10,0,1)
  samples<-log((exp(3)-1)*U +1)

  return (samples)
}

simul<-inverse_4()
```

```
m<-mean(simul)
T<-abs(m-2)
simul_m<-simul-m+2


bootstrap_4_p <- function(){
  b<-1000
  boot_p<-vector()
  for(i in 1:b){
    sel<-sample(simul_m,10,replace=TRUE)
    med<-abs(mean(sel)-2)
    boot_p<-append(boot_p,med)
  }

  ar<-sum(boot_p>T)+1
  par<-b+1
  return (ar/par)

}

print(bootstrap_4_p())
```

Η p-value που προέκυψε είναι **0.8931069** > 0.05 , συνεπώς **ΔΕΝ απορρίπτεται η H0** και η υπόθεση ότι η $f(x)$ έχει μέση τιμή ίση με 2 θα μπορούσε να γίνει αποδεκτή.

Για το ίδιο ερώτημα, αν δηλαδή η $f(x)$ έχει μέση τιμή ίση με 2, θα προσπαθήσουμε να απαντήσουμε εφαρμόζοντας ακόμα μια μέθοδο. Στη συνέχεια θα συγκρίνουμε τα αποτελέσματα με την παραπάνω.

Η μέθοδος αυτή είναι να παράγουμε την μέση τιμή για κάθε Bootstrap δείγμα, να τις ταξινομήσουμε σε αύξουσα σειρά και να επιλέξουμε ένα διάστημα για το οποίο θα ελέγξουμε αν η μέση τιμή 2 είναι μέσα. Αυτή η μέθοδος λέγεται Bootstrap με διάστημα εμπιστοσύνης βασισμένο σε ποσοστιαία σημεία. Στην περίπτωση μας το διάστημα εμπιστοσύνης που ζητείται είναι 95% επομένως θέλουμε να κρατήσουμε από την $B * 0.025$ έως και την $B * 0.075$ τιμή. Αυτό υλοποιήθηκε ως εξής :

```
bootstrap_4_95 <- function(){
  b<-1000
  boot_95<-vector()
  for(i in 1:b){
    sel<-sample(simul,10,replace=TRUE)
    boot_95<-append(boot_95,mean(sel))
  }

  ci=sort(boot_95)[25:975]
  hist(ci)
  print(paste("[",ci[1],",",ci[951],"]"))
  if((ci[1]<=2)&&(ci[950]>=2)){
    print("H0 NOT REJECTED")
  }
  else{
    print("H0 REJECTED")
  }

}

bootstrap_4_95()
```

Το αποτέλεσμα της παραπάνω μεθόδου είναι ότι το διάστημα εμπιστοσύνης 95% είναι το $[2.05666732177961, 2.60975009735722]$, βλέπουμε ότι δεν είναι η τιμή 2 μέσα, επομένως **απορρίπτουμε την μηδενική υπόθεση**.

Στο σημείο αυτό να αναφέρουμε ότι η συγκεκριμένη μέθοδος μπορεί να θεωρηθεί αξιόπιστη μόνο όταν η κατανομή των τιμών που εξετάζουμε είναι συμμετρική. Συνεπώς παρουσιάζουμε το ιστόγραμμα των μέσων τιμών

που προέκυψαν από τα Bootstrap δείγματα.

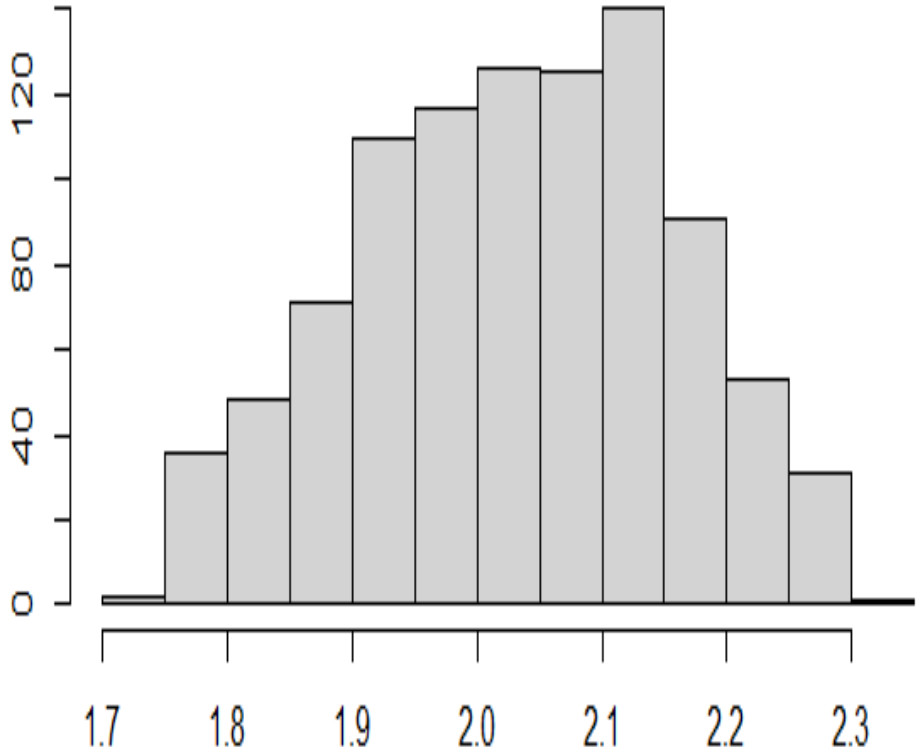


Figure 2.4: Histogram of Bootstrap's mean values

Μπορεί να μην παρατηρείται η απόλυτη συμμετρία, αλλά η κατανομή είναι αρκετά κοντά στη συμμετρική και για αυτό τον λόγο μπορούμε να θεωρήσουμε την μέθοδο Bootstrap με διάστημα εμπιστοσύνης βασισμένο σε ποσοστιαία σημεία, αρκετά αξιόπιστη για τα συγκεκριμένα δεδομένα.

Σύμφωνα με τα παραπάνω η μέθοδος ελέγχου υποθέσεων δεν απέρριψε την H0, ενώ η μέθοδος των ποσοστιαίων σημείων την απέρριψε. Έχουμε τη δυνατότητα να υπολογίσουμε την μέση τιμής της συνάρτησης πυκνότητας πιθανότητας και να συγκρίνουμε ποια διαδικασία μπορεί να θεωρηθεί πιο αξιόπιστη, τουλάχιστον για τα δεδομένα της άσκησης. Η διαδικασία υπολογισμού της μέσης τιμής είναι η εξής :

$$\int_0^3 xf(x) = \int_0^3 \frac{xe^x}{e^3-2} = \frac{1}{e^3-1}\left[e^x(x-\alpha)\right]_0^3 = 2.157$$

Συμπερασματικά η μεθοδολογία των ποσοστιαίων σημείων μπορεί να θεωρήθεί πιο αξιόπιστη στο συγκεκριμένα ζητούμενο, καθώς η μέση τιμή δεν είναι 2.

3. Άσκηση 3

3.1 Ερώτημα α

3.1.1 i

Στο πρώτο σκέλος του ερωτήματος μας ζητείτε να βρούμε την επαρκής στατιστική συνάρτηση , όταν μας δίνεται ένα τυχαίο δείγμα μήκους n που ακολουθεί την κατανομή Gamma(a,b) με συνάρτηση πυκνότητας πιθανότητας την εξής:

$$f(x) = \frac{b^a}{\Gamma(a)} x^{\alpha-1} e^{-bx}$$

όπου $a, b > 0$ άγνωστες παράμετροι.

Μια στατιστική συνάρτηση ονομάζεται μια συνάρτηση $T = r(X_1, X_2, ..., X_n)$ ενός τυχαίου δείγματος μήκους n που γνωρίζουμε ότι έχει παραχθεί από μια οικέγενεια κανατομών με αγνώστες παράμετρους την $a, b, ...$ κτλπ. **Επαρκής** στατιστική συνάρτηση ονομάζουμε εκείνη για την οποία αν γνωρίζουμε την τιμή της, τότε μπορούμε να υπολογίσουμε τις άγνωστες παραμέτρους της συνάρτησης πυκνότητας πιθανότητας της κατανομής που εξετάζουμε, με την ίδια ακρίβεια αν γνωρίζαμε το δείγμα. Για παράδειγμα αν έχουμε ένα δείγμα $X_1, X_2, ..., X_n$ από την κανονική κατανομή για την οποία άγνωστες παράμετροι είναι μόνο η μέση τιμή και θεωρούμε ότι η διασπορά ειναι γνωστή, τότε αν γνωρίζουμε ότι έχουμε δείγμα n σημείων και γνωρίζουμε την τιμή της επαρκούς στατιστικής συνάρτησης με τύπο :

$$T = r(X_1, X_2, ..., X_n) \sum_{i=1}^{n} Xi$$

μπορούμε να υπολογίσουμε και με την ίδια ακρίβεια την μέση τιμής της, όπως θα κάναμε αν είχαμε το δείγμα. Αν είχαμε το δείγμα θα μπορούσαμε να χρησιμοποιήσουμε τον αμερόληπτο εκτιμητή :

$$\bar{X}_n = \frac{1}{n} \quad \sum_{i=1}^{n} X_i = \frac{1}{n} T$$

Σύμφωνα με το Fisher–Neyman factorization theorem [3] αν έχουμε τυχαίο δείγμα $X_1, X_2, ..., X_n$ με joint pdf $f(x_1, x_2, \dots x_n \mid \theta)$, τότε η στατιστική συνάρτηση $T = r(X_1, X_2, ..., X_n) \sum_{i=1}^{n} Xi$ είναι επαρκής αν και μόνο αν μπορούμε να μετατρέψουμε την joint pdf στην εξής μορφή :

$$f(x_1, x_2, \dots x_n \mid \theta) = h(x) \cdot g(T, \theta)$$

Ακριβώς αυτή θα είναι και η μεθοδολογία μας, την οποία υλοποιούμε αναλυτικά παρακάτω :

Η joint pdf στην περίπτωση μας είναι της μορφής

$$f(x_1, x_2, \dots x_n \mid a, b) = \prod_{i=1}^{n} \left(\frac{b^a}{\Gamma(a)} \right) xi^{a-\alpha} e^{-bx_i} = \left(\frac{b^a}{\Gamma(a)} \right)^n \cdot \left[\prod_{i=1}^{n} xi^{a-1} \right] e^{-b\sum_{i=1}^{n} x_i} = \frac{b^{an}}{[\Gamma(a)]^n} \cdot \left[\prod_{i=1}^{n} x_i \right]^{a-1} \cdot \left[e^{-b\sum_{1}^{n} x_i} \right]$$

Θέτω :

- $h(x) = 1$
- $T = [\prod_{i=1}^{n} x_i, \sum_{1}^{n} x_i]$
- $g(T, a, b) = \frac{b^{an}}{[\Gamma(a)]^n} \cdot [T[1]]^{a-1} \cdot \left[e^{-bT[2]}\right]$
- $f\left(x_1, x_2, \ldots x_n \mid \theta\right) = h(x) \cdot g(T, a, b)$

Συνεπώς καταλήξαμε στη ζητούμενη επαρκή στατιστική συνάρτηση $T = [\prod_{i=1}^{n} x_i, \sum_{1}^{n} x_i]$ **διάστασης 2**.

3.1.2 ii

Στο δεύτερο σκέλος του ερωτήματος μας ζητείται να αναπτύξουμε θεωρητικά τα βήματα, δηλαδή τον όρο ανανέωσης, της μεθόδου Newton Raphson των α,β για την μεγιστοποίηση της λογαριθμικής πιθανοφάνειας της συνάρτησης πυκνότητα πιθανότητας που μας δίνεται.

Αρχικά να αναφέρουμε ότι η μέθοδος Newton Raphson, αποτελεί μια μέθοδο για να προσεγγίσουμε την ρίζα μιας εξίσωσης. Πιο συγκεκριμένα αν έχουμε μια συνάρτηση, συνεχή και παραγωγίσιμη, $f(x)$ και θέλουμε να βρούμε το x_0 για το οποίο ισχύει $f(x_0) = 0$, τότε επιλέγουμε μια αρχική τιμή για το x, για την οποία θεωρούμε ότι βρίσκεται κοντά στο x_0 και επαναλαμβάνουμε την εξής ανανέωση ή αλλίως βήμα :

$$x_{i+1} = x_i - \frac{f\left(x_i\right)}{f'\left(x_i\right)}$$

Για τον τερματισμό της διαδικασίας συνήθως θέτουμε ένα όριο στην ελάχιστη διαφορά μεταξύ δυο διαδοχικών επαναλήψεων.

Συνεπώς στην συγκεκριμένη περίπτωση έχουμε να μεγιστοποιήσουμε την λογιστική πιθανοφάνεια. Αρχικά η πιθανοφάνεια ορίζεται ως εξής :

$$L(a, b) = \left(\frac{b^a}{\Gamma(a)}\right)^n \cdot \left[\prod_{i=1}^{n} xi\right]^{a-1} e^{-b\sum_{i=1}^{n} x_i}$$

και ο λογάριθμος είναι :

$$l(a, b) = na\log(b) - n\log\Gamma(a) + (a-1)\sum_{i=1}^{n}\log x_i$$

Θέλουμε να μεγιστοποιήσουμε την παραπάνω συνάρτηση επομένως πρέπει να ισχύει ότι :

- $$\frac{\partial \ell(a, b)}{\partial b} = \frac{na}{\hat{b}} - \sum_{i=1}^{n} xi = 0 \Rightarrow \hat{b} = \frac{a}{\bar{x}} \tag{3.1}$$
- $$\frac{\partial l(a, b)}{\partial a} = 0 \Rightarrow n \cdot \log b - n\psi(a) + \sum_{i=1}^{n}\log x_i = 0 \tag{3.2}$$

  , όπου $\psi(a)$: digamma function , $\frac{d\log\Gamma(x)}{dx}$

- Αντικαθιστάμε την (3.1) στην (3.2) και έχουμε

$$\frac{\partial l(a, \hat{b})}{\partial a} = 0 \Rightarrow n \cdot \log\frac{a}{\bar{x}} - n\psi(a) + \sum_{i=1}^{n}\log x_i = 0$$

Επομένως για τον μονοδιάστατο Newton Raphson, αφού πλεον σε κάθε βήμα έχουμε ανανέωση μόνο της παραμέτρου α, θα ισχύει ο εξής κανόνας :

$$a_{i+1} = a_i - \frac{l'_a(ai)}{l''_a(a_i)}$$

Τον αριθμητή τον υπολογίσαμε παραπάνω , ενώ ο παρονομαστής θα πάρει την εξής τιμή :

$$\frac{\partial^2 l(a)}{\partial a^2} = \frac{n}{a} - n\psi'(a)$$

, οπου $\psi'(a)$: trigamma function , $\frac{d^2 \log\Gamma(x)}{dx^2}$
Συνεπώς το τελικό μας αποτέλεσμα που αφορα το βήμα του Newton Raphson είναι :

$$a_{i+1} = a_i - \frac{\sum_{i=1}^{n} \log x_i + n\log\frac{a_i}{\bar{x}} - n\psi(a_i)}{\frac{n}{a_i} - n\psi'(a_i)} \tag{3.3}$$

Όταν το κριτήριο τερματισμού που θέσουμε, ολοκληρωθεί, τότε μπορούμε να υπολογίσουμε και το b που προκύπτει απο τον τύπο (3.1) με χρήση όμως του προηγούμενου α, δηλαδή $b_i = \frac{\alpha_{i-1}}{\hat{x}}$.

3.2 Ερώτημα β

3.2.1 i

Στο πρώτο κομμάτι του ερωτήματος, μας ζητείται να αποδείξουμε ότι μια τυχαία μεταβλητή X, ακολουθεί την κατανομή Polya(a,b) με συνάρτηση πυκνότητας πιθανότητας :

$$f(x) = \frac{\Gamma(x+a)}{x!\Gamma(a)} \left(\frac{b}{1+b}\right)^{\alpha} \left(\frac{1}{1+b}\right)^{x}$$

όταν ,

$$X \mid \theta \sim Poisson(\theta)$$

$$\theta \sim Gamma(a, b)$$

Προκείται για ιεραρχική προσέγγιση μιξης μοντέλου και ο τρόπος που θα το αντιμετωπίσουμε βασίζεται στη χρήση του θεωρήματος της ολικής πιθανότητας[4] και του θεωρήματος του Bayes (Extended form) για συνεχείς τυχαίες μεταβλητές [5], το οποίο αναφέρει το εξής :

$$f_Y(y) = \int_{-\infty}^{+\infty} f_{y|x=\xi}(y) f x(\xi) d\xi$$

Πριν συνεχίσουμε στην επιλύση του ζητήματος να αναφέρουμε πως η αρχική ιδέα βασίστηκε στο βιβλίο Statistical Inference [6] και πιο συγκεκριμένα στη παράγραφο 4.4.

Αρχικά η τ.μ θ προέρχεται απο την κατανομή Gamma συνεπώς δεν μπορεί να πάρει αρνητικές τιμές. Επομένως έχω ότι :

$$f_X(x) = \int_0^{\infty} \left[\frac{e^{-\theta}\theta^x}{x!}\right] \cdot \left[\frac{b^a}{\Gamma(\alpha)}\theta^{\alpha-1}e^{-b\theta}\right] d\theta = \frac{b^a}{x!\Gamma(\alpha)} \int_0^{\infty} \theta^{x+a-1}e^{-\theta(1+b)}d\theta \tag{3.4}$$

Από σημειώσεις του μαθήματος γνωρίζω ότι :

$$\int_0^{\infty} \theta^{\alpha-1}e^{-b\theta}d\theta = \frac{\Gamma(\alpha)}{b^{\alpha}} \tag{3.5}$$

Συνεπώς αν θέσω στη (3.4) μέσα στο ολοκλήρωμα όπου $b+1 \to b$ και όπου $x+\alpha \to \alpha$ τότε προκύπτει ακριβώς η (3.5) και με αντίστροφη ανάθεση καταλήγω τελικα στο :

$$f(x) = \frac{b^a}{x!\Gamma(a)} \frac{\Gamma(x+a)}{(b+1)^{x+a}} = \frac{\Gamma(x+a)}{x!\Gamma(a)} \left(\frac{b}{b+1}\right)^a \left(\frac{1}{b+1}\right)^x$$

Έχουμε το ζητούμενο !

3.2.2 ii

Στο δεύτερο κομμάτι του ερωτήματος θεωρούμε ότι έχουμε παρατηρήσιμο δείγμα $X_1, X_2, ..., X_n$, όπου n=10000. Το δείγμα αυτό προέρχεται απο την κατανομή Polya(α,β) , η οποία έχει κρυφές παραμέτρους τα θ_i. Συνεπώς μπορούμε να θεωρήσουμε σαν ολόκληρο δείγμα την τούπλα (X_i, θ_i) . Αν γνωρίζουμε ότι το παρατηρήσιμο δείγμα προέρχεται από την κατανομή Polya αλλά δεν γνωρίζουμε τις παραμέτρους της , τότε η σύνηθης διαδικασία εύρεσης των παραμέτρων είναι η μεγιστοποίηση της λογαριθμικής πιθανοφάνειας.

Κατα τη διαδικασία αυτή σχηματίζουμε την συνάρτηση λογαριθμικής πιθανοφάνειας και στόχος μας είναι να την μεγιστοποιήσουμε, καθώς η μεγιστοποίηση της σημαίνει και μεγιστοποίηση της πιθανότητας τα παρατηρήσιμα δεδομένα να προέρχονται από την αντίστοιχη κατανομή.

Σε πολλές περιπτώσεις όμως σε "real world" προβλήματα υπάρχουν είτε κρυφές μεταβλητές είτε τιμές στο δείγμα που έχουν αλλοιωθεί και θεωρούνται χαμένες. Σε αυτές τις περιπτώσεις ο αλγόριθμος της μεγιστοποίησης της λογαριθμικής πιθανοφάνειας δεν μπορεί να εφαρμοστεί και προτιμάται ο αλγόριθμος Expectation Maximization για την προσέγγιση των παραμέτρων της κατανομής που μεγιστοποιούν την λογαριθμική πιθανοφάνεια.

Πιο συγκεκριμένα στη δικία μας περίπτωση στόχος είναι ο εξής :

$$\max L(a, b \mid X)$$

Δηλαδή αν έχουμε το δείγμα X η μέγιστη πιθανότητα τα a,b της Polya(a,b) να τα έχουν παράξει. Όμως οι κρυφές παράμετροι θ_i δεν μας επιτρέπουν να υπολογίσουμε τα συγκεκριμένα a,b. Η γενικη προσέγγιση του αλγόριθμου EM είναι να αρχικοποιεί με μια τιμή που θεωρεί κοντά στις πραγματικές, τις ζητούμενες παραμέτρους a,b. Με βάση αυτές τις τιμές να υπολογίζεται η αναμενόμενη τιμή των κρυφών μεταβλητών. Τέλος με βάση τις αναμενόμενες τιμές να ακολουθούμε τον αλγόριθμο μεγιστοποίησης της λογαριθμικής πιθανοφάνειας που εξηγήσαμε παραπάνω και να εκτιμάμε τις νέες παραμέτρους a,b. Αυτή η επανάληψη του υπολογισμού της αναμενόμενης τιμής των κρυφών μεταβλητών και της μεγιστοποίησης της λογαριθμικής πιθανοφάνειας επαναλαμβάνεται μέχρι να ικανοποιηθεί κάποιο κριτήριο. Η πρώτη από τις 2 διαδικασίες ονομάζεται Expectation step και η δεύτερη ονομάζεται Maximization step.

Πρέπει να παρατηρήσουμε ότι το Expectation step είναι ένα βοηθητικό βήμα, έτσι ώστε να κάνει την διαδικασία του Maximization εφικτή. Όπως είδαμε και στο ερώτημα (α) η επαρκής στατιστική συνάρτηση είναι ικανή για να υπολογίσουμε τις παραμέτρους μια συνάρτησης πυκνότητας πιθανότητας. Πιο συγκεκριμένα αν έχουμε την επαρκή στατιστική συνάρτηση της συνάρτησης πυκνότητας πιθανότητας της Gamma(a,b) τότε μπορούμε να υπολογίσουμε τα a,b. Συνεπώς στη περίπτωση μας μετατρέπουμε τα βήματα του EM αλγορίθμου στα εξής:

- Expectation step: Υπολογισμός της επαρκούς στατιστικής συνάρτησης της Gamma(a,b) με χρήση των ήδη υπολογισμένων τιμών a,b. Την συνάρτηση αυτή την έχουμε ορίσει στο ερώτημα (α) και είναι η εξής : $T = [\prod_{i=1}^n x_i, \sum_1^n x_i]$, όπου στη δικία μας περίπτωση τα x_i είναι τα θ_i. Για μετέπειπα ευκολία αντί για τον υπολογισμό του $\prod_{i=1}^n \theta_i$ θα προτιμήσουμε τον υπολογισμό του $\sum_{i=\alpha}^n \log \theta i$, καθώς όπως γνωρίζουμε δίνουν την ίδια πληροφορία.

- Αφού έχει οριστεί η επαρκής στατιστική συνάρτηση στο προηγούμενο βήμα θα χρησιμοποιήσουμε την τιμή της για να μεγιστοποιήσουμε την λογαριθμική πιθανοφάνεια της Gamma(a,b) με στόχο την εύρεση του a,b. Όπως αναφέρεται και στην εκφώνηση αυτό το βήμα έχει μελετηθεί με τον αναλυτικό τύπο της στο ερώτημα (α) στο οποίο γίνεται χρήση της Newton Raphson μεθόδου.

Για το **Expectation step** θα υπολογίσουμε 2 ποσότητες :

- $C_i = E\left[\theta_i \mid x_i, a_{old}, b_{old}\right]$. Έχουμε ότι

$$P\left(\theta_i \mid x_i; a_{old}, b_{old}\right) = \frac{P(x_i \mid \theta_i; a_{old}, b_{old}) \cdot P\left(\theta_i; a_{old}, b_{old}\right)}{P\left(x_i\right)} =$$

$$= \frac{\frac{\theta_i^{xi}}{xi!} \cdot \frac{b_{old}^{a_{old}}}{\Gamma(a_{old})} \cdot \theta_i^{a_{old}-1} \cdot e^{-b_{old}\theta_i}}{\frac{\Gamma(x_i+a_{old})}{x_i!\Gamma(a_{old})} \cdot \left(\frac{1}{1+b_{old}}\right)^{x_i}} = \frac{e^{-\theta_i(1+b_{old})}\theta_i^{x_i+a_{old}-1} \cdot (1+b_{old}))^{x_i+a_{old}}}{\Gamma\left(x_i + a_{old}\right)} =$$

$$= Gamma(x_i + a_{old}, 1 + b_{old})$$

Γνωρίζουμε από σημειώσεις ότι $E[Gamma(a,b)] = \frac{a}{b}$, επομένως :

$$E\left[\theta_i \mid x_i, a_{old}, b_{old}\right] = E[Gamma(x_i + a_{old}, 1 + b_{old})] = \frac{x_i + a_{old}}{1 + b_{old}}$$

- $D_i = E\left[log\theta_i \mid x_i, a_{old}, b_{old}\right]$. Από σημειώσεις γνωρίζουμε ότι $E[\log x] = \psi(a) - \log(b)$, επομένως :

$$E\left[log\theta_i \mid x_i, a_{old}, b_{old}\right] = \psi(x_i + a_{old}) - log(1 + b_{old})$$

Για το **Maximization step** θα κάνουμε χρήση της (3.3) και η ανανέωση του a γίνεται με τον εξής τύπο :

$$a_{new} = a_{old} - \frac{\sum_{i=1}^{n} D_i + n \log a_{old} - n \log \frac{1}{n} \cdot \sum_{i=1}^{n} Ci - n\psi\left(a_{old}\right)}{\frac{n}{a_{old}} - n\psi'(a_{old})}$$

και σε κάθε βήμα μπορούμε να υπολογίζουμε το b, οπως είχαμε ορίσει στο ερώτημα (α) δηλαδή :

$$b_{new} = \frac{a_{old}}{\frac{1}{n} \cdot \sum_{i=1}^{n} Ci}$$

Η πορεία του αλγορίθμου είναι Expectation step → Maximization step → Expectation step → Maximization step → ... , μέχρι να ικανοποιηθεί το κριτήριο που ορίζεται στην εκφώνηση, δηλαδή το $(a_{new} - a_{old})^2 + (b_{new} - b_{old})^2 \leq 10^{-10}$. Στο σημείο αυτό να αναφέρουμε πως το Maximization step απαιτεί την διαδικασία του αλγορίθμου Newton Raphson, για τον οποίο έχουμε αναφέρει ότι, σταματάει μόνο όταν ακολουθήσει ένα κριτήριο σύγκλισης. Επιλέγουμε αυτό το κριτήριο να είναι το $(a_{new} - a_{old})^2 \leq 10^{-10}$. Να σημειωθεί ότι ο όρος 10^{-10} στον αλγόριθμο Newton Raphson τοποθετήθηκε ύστερα από δοκιμές με κριτήριο τόσο τη ταχύτητα σύγκλισης του αλγορίθμου EM όσο και τη ποιότητα σύγκλισης.

Παρακάτω παρουσιάζουμε τον κώδικα που χρησιμοποιήθηκε για την υλοποίηση:

```
set.seed(500)
n<-10000
poly<-rpois(n,lambda = rgamma(n,shape=2,rate=5))

a_old<-1
b_old<-1

reps_em<-1

repeat{
  C<-vector()
  D<-vector()

  for(i in 1:length(poly)){
    C_mid<-(poly[i]+a_old)/(1+b_old)
    D_mid<-digamma(poly[i]+a_old)-log(1+b_old)

```

```
    C<-append(C,C_mid)
    D<-append(D,D_mid)

  }

  b_new<-a_old*n/(sum(C))
  reps_nr<-1
  repeat{

    ar<-sum(D)+n*log(a_old)-n*log(sum(C)/n)-n*digamma(a_old)
    par<-(n/a_old)- n*trigamma(a_old)
    a_new<-a_old-(ar/par)


    if((a_new-a_old)^2<=10^(-10)){
      break
    }
    a_old<-a_new
  }



  if((a_new-a_old)^2+(b_new-b_old)^2<=10^(-10)){
    break
  }
  if(reps_em%%50==0){
  print(paste("Criterion:",(a_new-a_old)^2+(b_new-b_old)^2))
  print(paste("a:",a_new))
  print(paste("b:",b_new))
  print(paste("EM REP COUNTER:",reps_em))
  }
  a_old<-a_new
  b_old<-b_new
  reps_em<-reps_em+1
}
print(a)
print(b)
print(reps_em)
```

Χρειάστηκαν **499** επαναλήψεις του αλγόριθμου Expectation-Maximization και τα αποτελέσματα μας είναι **a=1.95100463233796** , **b=5.01409210912051**. Παραθέτουμε έναν πίνακα με κάποιες ενδεικτικές τιμές που παρουσίασε ο αλγόριθμος στη πορεία εκτέλεσης του :

RepCount	a	b	Criterion
50	1.57631953942215	3.95783850242508	0.000354495204135619
100	1.79488162336943	4.57063342911625	5.87416433496674e-05
150	1.88451073812799	4.82467572984569	1.04613113150033e-05
200	1.92247881839585	4.93273658018286	1.92187797746111e-06
250	1.93878047142022	4.97921137559416	3.57816891195863e-07
300	1.9458197050374	4.99929412249706	6.7003189362023e-08
350	1.94886679178404	5.00799006124461	1.25779941997258e-08
400	1.95018718877151	5.01175878240322	2.3637182192877e-09
450	1.9507596195552	5.01339272820818	4.98949057098169e-10
499	1.95100463233796	5.01409210912051	9.7017098171719e-11

Τα συγκεκριμένα αποτελέσματα είναι πολύ ικανοποιητικά, αλλά οφείλουμε να πούμε πως η ποιότητα των τελικών αποτελεσμάτων οφείλεται και στην τυχαιότητα με την οποία παράγεται το αρχικό μας δείγμα. Σε

διαφορετικές δοκιμές, παραμέτρων πχ.αλλαγή του 10^{-10} στον αλγόριθμο Newton Raphson, αλλά και διαφορετικών παραγόμενων δειγμάτων λάβαμε τιμές που μπορεί να απείχαν 0.5 από τις πραγματικές τιμές.

4. Άσκηση 4

4.1 Προεπεξεργασία Εκφώνησης

Στο πλαίσιο της προεπεξεργασίας θα υλοποιήσουμε τα βήματα που αναφέρονται στην εκφώνηση, για την δημιουργία των εξαρτημένων και επεξηγηματικών μεταβλητών.

Αρχικά μας ζητείται να παραγάγουμε τιμές για τις 10 πρώτες επεξηγηματικές μεταβλητές από την πολυδιάστατη κανονική κατανομή με μέση τιμή 0 και πίνακα συνδιακύμανσης τον ταυτοτικό. Όταν ο πίνακας συνδιακύμανσης είναι ο ταυτοτικός τότε προκύπτει ότι κάθε μεταβλητή είναι ασυσχέτιστη με την άλλη. Συνεπώς η ιδιότητα της πολυδιάστατης κανονικής κατανομής που προσθέτει συσχετίσεις σε τυχαίες μεταβλητές από την κανονική κατανομή δεν θα τεθεί σε εφαρμογή στην περίπτωση μας, συνεπώς μπορούμε να παράξουμε κάθε μια από τις 10 πρώτες με τη χρήση της κανονικής κατανομής με $\mu = 0$ και $\sigma^2 = 1$ καθώς $Var(x) = Cov(x, x) \Rightarrow Var(x) = 1$.

Ο κώδικας που χρησιμοποιήθηκε είναι ο εξής :

Σημείωση: Χρησιμοποιούμε seed έτσι ώστε κάθε φορά που τρέχουμε τον κώδικα να παραγάγουμε τα ίδια δεδομένα και να έχουμε τα ίδια αποτελέσματα στις μετέπειτα αναλύσεις μας.

```
set.seed(42)

X_10<-matrix(rnorm(10*50,0,1),ncol=10)
```

Στην συνέχεια θα παραγάγουμε τις υπόλοιπες επεξηγηματικές μεταβλητές από την σχέση

$$X_{ij} \sim N\left(0.2X_{i1} + 0.4X_{i2} + 0.6X_{i3} + 0.8X_{i4} + 1.1X_{i5}, 1\right)$$

```
X_5 <- array(NA,dim = c(50,5))


for(j in 1:5){

  for(i in 1:50){

    mu <- (0.2*X_10[i,1]) + (0.4*X_10[i,2]) + (0.6*X_10[i,3]) + (0.8*X_10[i,4]) + (1.1*
   X_10[i,5])

    sim<-rnorm(1,mu,1)

    X_5[i,j] <- sim
    }
}

X <- cbind(X_10,X_5)
colnames(X) <- c("X1", "X2", "X3", "X4", "X5", "X6", "X7", "X8", "X9", "X10", "X11", "
   X12", "X13", "X14", "X15")
```

Τέλος θα παραγάγουμε και την εξαρτημένη μεταβλητή από τη σχέση :

$$Yi \sim N\left(4 + 2X_{i1} - X_{i5} + 2.5X_{i7} + 1.5X_{i11} + 0.5X_{i13}, 1.5^2\right)$$

```
Y<-array(NA,dim = c(50,1))

for(i in 1:50){
  mu<-(4+2*X[i,'X1']-X[i,'X5']+2.5*X[i,'X7']+1.5*X[i,'X11']+0.5*X[i,'X13'])
  Y[i]<- rnorm(1,mu,1.5)
}

colnames(Y) <- c("Y")
```

Έχοντας ετοιμάσει τα δεδομένα είμαστε έτοιμοι να προχωρήσουμε στα ερωτήματα της άσκησης.

4.2 Ερώτημα α

Στο ερώτημα αυτό μας ζητείτε να βρούμε το μοντέλο που ελαχιστοποιεί το κριτήριο BIC. Το κριτήριο BIC ακολουθεί τον παρακάτω τύπο :

$$BIC = \ln(n)k - 2\ln(\hat{L})$$

- n: το πλήθος των παρατηρήσεων
- k: το πλήθος των επεξηγηματικών μεταβλητών στο μοντέλο μας +2, δηλαδή την σταθερά και τη διασπορά των σφαλμάτων
- $\hat{L}$: η μεγιστοποιημένη συνάρτηση πιθανοφάνειας του μοντέλου μας.

Από τα παραπάνω, και σε συνδυασμό με το ότι στόχος είναι να ελαχιστοποιήσουμε το BIC κριτήριο, διαπιστώνουμε ότι το BIC κριτήριο προσμετρά με θετικό πρόσημο την συνάρτηση πιθανοφάνειας , δηλαδή το πόσο καλά το μοντέλο ανταπεξέρχεται στο συγκεκριμένο πρόβλημα παλινδρόμησης και προσμετρά με αρνητικό πρόσημο την πολυπλοκότητα του μοντέλου μας, δηλαδή το πλήθος των επεξηγηματικών μεταβλητών που έχει. Αυτό έχει ως αποτέλεσμα το μοντέλο μας να ταιριάζει με τον καλύτερο δυνατό τρόπο στο πρόβλημα μας με τις λιγότερες δυνατές επεξηγηματικές μεταβλητές έτσι ώστε να αποφεύγουμε και το under fit και το over fit.

Για τις ανάγκες του ερωτήματος πρέπει να κωδικοποιήσουμε κάθε πιθανό μοντέλο με τις 15 επεξηγηματικές μεταβλητές που διαθέτουμε. Από την στιγμή που σε ένα μοντέλο μια μεταβλητή μπορεί να παίρνει 2 καταστάσεις ,δηλαδή να υπάρχει ή όχι, προκύπτουν 2^{15} τέτοια μοντέλα. Συνεπώς για την κατασκευή τους αρχικά φτιάχνουμε έναν πίνακα 2^{15} γραμμών και 15 στηλών όπου έχουμε την αναπαράσταση σε δυαδικό σύστημα 15-bit όλων των αριθμών από το 0 εως το $2^{15} - 1$

```
models=data.frame(matrix(ncol = 15, nrow = ((2^15))))

for (i in 0:(2^15-1)){
  a=as.numeric(intToBits(i))
  for (j in 1:15){
    models[i, j]<-a[j]
  }
}
```

Στη συνέχεια έχοντας κάνει αυτή την κωδικοποίηση μπορούμε να θέσουμε σαν μάσκα κάθε δυαδικό αριθμό, και να επιλέξουμε κάθε φορά τις επεξηγηματικές μεταβλητές που αντιστοιχούν σε 1 στον δυαδικό αριθμό. Συνεπώς αν ο δυαδικός μας αριθμός αντιστοιχεί στο:

- 0 : 000000000000000 → κενό μοντέλο, προσαρμόζεται μόνο η σταθερά.

- $2^{15} - 1$: 111111111111111 → πλήρες μοντέλο, λαμβάνονται υπ'όψιν και οι 15 επεξηγηματικές μεταβλητές

Ύστερα σε κάθε ένα από αυτά τα μοντέλα εφαρμόζουμε το κριτήριο BIC και το αποθηκεύουμε. Τέλος παρουσιάζουμε σε φθίνουσα σειρά ποια μοντέλα είχαν το μικρότερο BIC.

```
exhaustive_search <- function (X,Y){
  BIC <-rep (NA , nrow (models))
  reg0 <-lm(Y[,'Y']~1, data =X)
  BIC [1] <-BIC ( reg0 )
  for (i in 1: (2^15-1)){
    data <-X[ which (models[i ,] %in% 1) ]
    mod <-lm(Y[,'Y'] ~ ., data = data )
    BIC[i] <- BIC(mod)
    print(i)
  }
  models <-data.frame(models,BIC)
  return(models[order(models$BIC) ,])
}


outcome<-exhaustive_search(X_inp,Y_inp)
head(outcome)
```

Τα πρώτα 6 αποτελέσματα είναι τα εξής :

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	BIC
1	0	0	0	1	0	1	0	0	0	1	0	1	0	0	194.8788
1	0	0	0	0	0	1	0	0	0	1	1	1	0	0	194.8853
1	0	0	0	0	0	1	0	0	0	1	0	1	0	1	194.8862
1	0	0	0	0	0	1	0	1	0	1	1	1	0	0	195.4728
1	0	0	0	1	0	1	0	1	0	1	0	1	0	0	195.5092
1	0	0	0	0	0	1	0	0	0	1	1	1	0	1	195.7551

Το μοντέλο που προκύπτει από την παραπάνω διαδικασία συμπεριλαμβάνει τις $X_1, X_5, X_7, X_{11}, X_{13}$. Τα συγκεκριμένο αποτέλεσμα επιβεβαιώνει την ορθότητα του αλγορίθμου μας καθώς όπως είχαμε αναφέρει στην προεπεξεργασία η μέση τιμή της εξαρτημένης μεταβλητής Y περιγράφεται από τις συγκεκριμένες επεξηγηματικές μεταβλητές.

4.3 Ερώτημα β

Στο προηγούμενο ερώτημα με την χρήση του κριτηρίου BIC επιλέξαμε το μοντέλο που το ελαχιστοποιεί, συνεπώς πραγματοποιήσαμε μια επιλογή μοντέλου. Σε αυτό το ερώτημα με την τεχνική Lasso θα καθορίσουμε το βέλτιστο μοντέλο (σύμφωνα με τα αντίστοιχα κριτήρια) και θα καθορίσουμε το πόσο σημαντικές είναι οι επεξηγηματικές μεταβλητές, ρυθμίζοντας τον πολλαπλασιαστικό παράγοντα της καθεμίας. Συνεπώς δεν θα ποινικοποιήσουμε τόσο αυστηρά μια μεταβλητή αν δεν προσφέρει άμεσα στον υπολογισμό της εξαρτημένη μεταβλητής, με το να την αφαιρέσουμε από το μοντέλο όπως κάναμε στο προηγούμενο ερώτημα, αλλά θα μειώσουμε πολύ το αντίστοιχο coefficient.

Όπως γνωρίζουμε η τεχνική των Ελαχίστων Τετραγώνων στη γραμμική παλινδρόμηση στοχεύει στην εκπαίδευση ενός μοντέλου , δηλαδή στο καθορισμό των coefficients που θα ελαχιστοποιούν την συνάρτηση :

$$OLS = \sum_{i=1}^{n} (y_i - bx_i)^2$$

Η τεχνική του Lasso Regression στοχεύει στην εύρεση των coefficients που ελαχιστοποιούν την συνάρτηση :

$$Lasso = \sum_{i=1}^{n} (y_i - bx_i)^2 + \lambda \sum_{j=1}^{p} |b_j| \tag{4.1}$$

Όπως φαίνεται και από την συνάρτηση μια μεγάλη τιμή του λ θα οδηγήσει σε μεγαλύτερη μείωση τιμών των coefficients.

Για να ολοκληρώσουμε με τους ορισμούς ορίζουμε σαν παράγοντα συρρίκνωσης

$$S = \frac{||b||_1}{max||b||_1}$$

Ο παράγοντας συρρίκνωσης ποσοτικοποιεί το πόσο μικρότερο είναι το άθροισμα της διασποράς των εκτιμήσεων από την μέθοδο Lasso συγκριτικά με το άθροισμα της διασποράς των εκτιμήσεων από την μέθοδο των Ελαχίστων Τετραγώνων.

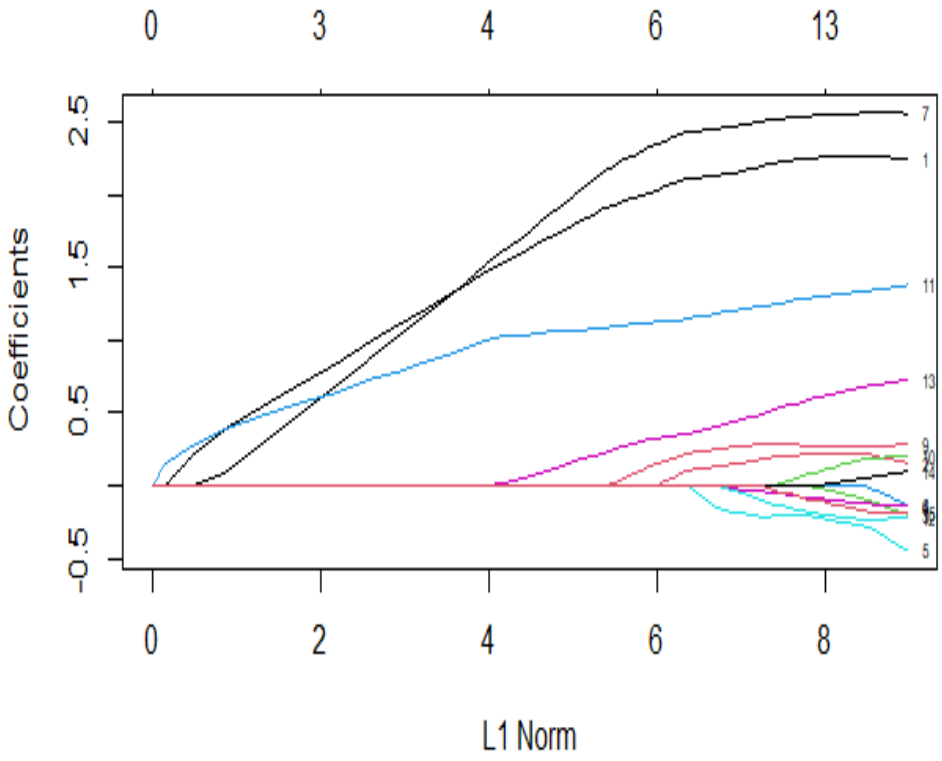
Η τεχνική του Lasso Regression προτιμάται σε περιπτώσεις όπου :

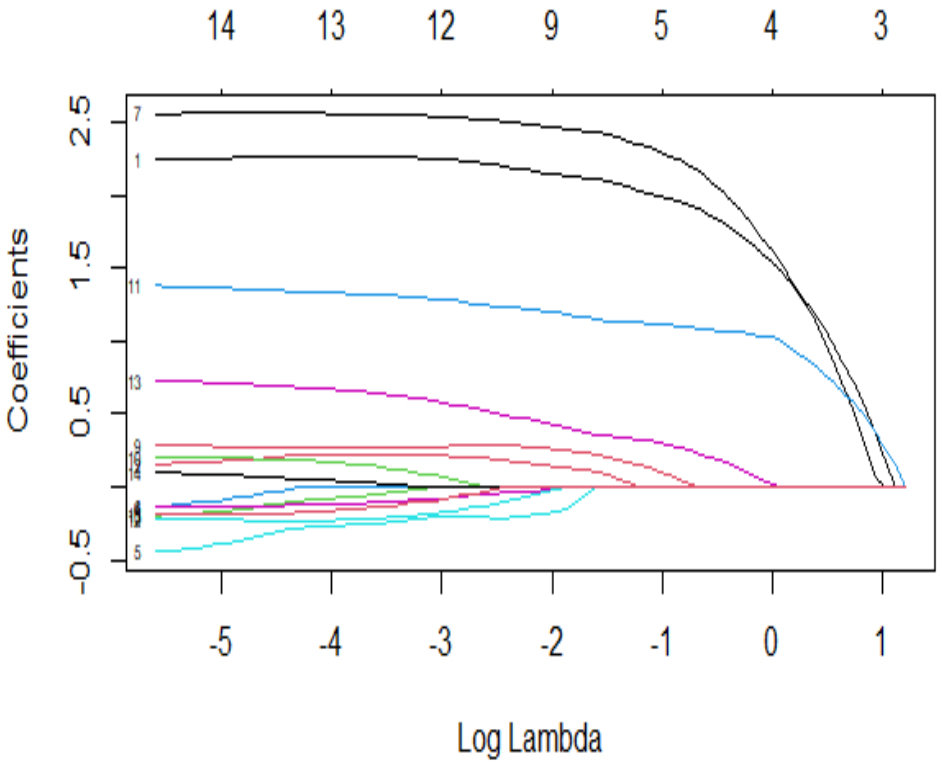
- Παρατηρούμε πολυσυγγραμικότητα στις επεξηγηματικές μεταβλητές μας. Μέσω της Lasso τεχνικής οι μεταβλητές που εξαρτώνται τείνουν να έχουν πολύ μικρή τιμή για coefficient και υψηλή τιμή να έχει μόνο η μια που έχει την μεγαλύτερη επίδραση στον υπολογισμό της εξαρτημένης.
- Όταν έχουμε λίγες παρατηρήσεις και αρκετές επεξηγηματικές μεταβλητές παρατηρούμε ότι η τεχνική του Lasso δίνει πολύ καλύτερα αποτελέσματα συγκριτικά με την τεχνική του OLS
- Χρησιμοποιείται για την αποφυγή του overfitting, καθώς μειώνει το variance του μοντέλου με το να συρρικνώνει τις τιμές των coefficients. Φυσικά θέλει προσοχή στην επιλογή καθώς μια αυστηρή πολιτική ποινικοποίησης μπορεί να οδηγήσει στην αύξηση του bias και συνεπώς να έχουμε ένα χειρότερο μοντέλο.

Όταν εφαρμόζεται αυτή η μέθοδος τότε έχουμε σαν υπερπαράμετρο το λ ή το S. Για την εύρεση του βέλτιστου μοντέλου θα εφαρμόσουμε Cross Validation μέσω της βιβλιοθήκης glmnet. Η συγκεκριμένη βιβλιοθήκη πραγματοποιεί Cross Validation με υπερπαράμετρο το λ, το σκεπτικό αυτό θα ακολουθήσουμε και εμείς. Αν επιλέγαμε να κάνουμε την αναζήτηση μας με υπερπαράμετρο το S, τότε θα καταλήγαμε ακριβώς στο ίδιο αποτέλεσμα καθώς υπάρχει $1-1$ αντιστοιχία μεταξύ των δυο παραμέτρων, και αντιστρόφως ανάλογη σχέση, καθώς όσο αυξάνεται το λ έχω μεγαλύτερη συρρίκνωση και όσο αυξάνεται το S έχω μικρότερη συρρίκνωση.

Αρχικά παρουσιάζουμε ορισμένα διαγράμματα για να κατανοήσουμε την συμπεριφορά του μοντέλου με διάφορες τιμές του S και του λ.

```
lasso <- glmnet(X_inp, Y_inp[,'Y'])
plot(lasso, label=T)
plot(lasso,xvar='lambda', label=T)
```





Παρατηρούμε ότι όσο μεγαλώνει η τιμή του LogLambda (ή αντίστοιχα όσο μειώνεται η τιμή του L1Norm που αντιστοιεί στο "S"), δηλαδή το λ, παραμένουν σημαντικές οι μεταβλητές X_1, X_7, X_{11}, X_{13}, καθώς η αντίστοιχη γραμμή αντιστοιχει στην τιμή που παίρνει το coefficient. Τις 4 συγκεκριμένες τις είχαμε συμπεριλάβει στο μοντέλο που είχε προκύψει με το κριτήριο BIC μαζί με τη μεταβλητή X_5, η οποία όπως βλέπουμε στο διάγραμμα είναι η 9η μεταβλητή από τις 15 που το coefficient της οδηγείται στο 0.

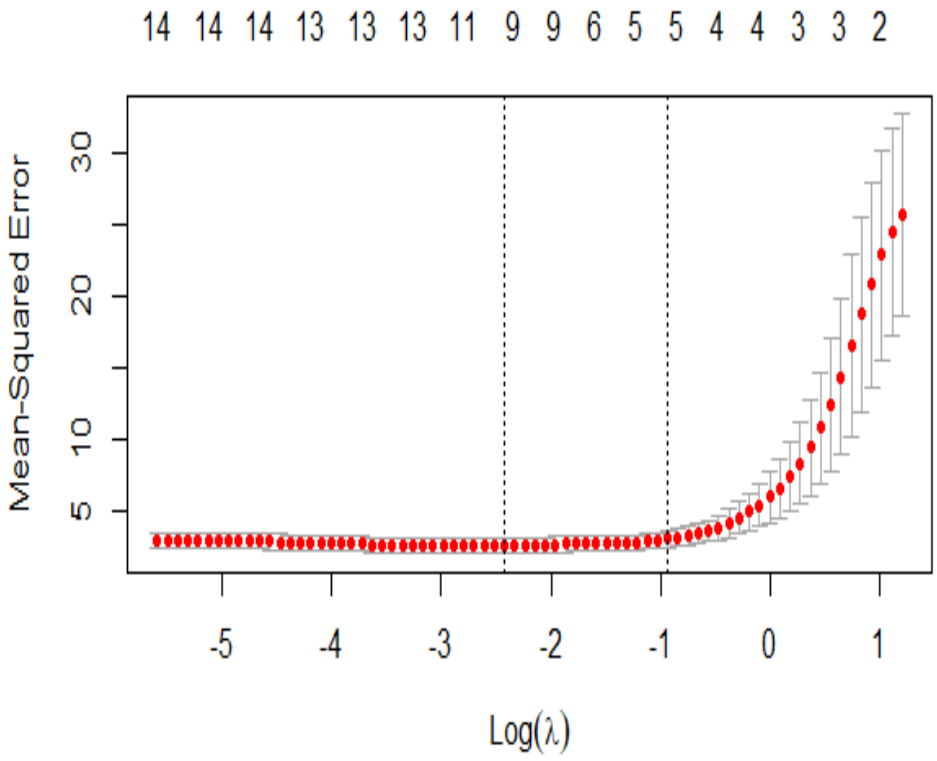
Στη συνέχεια θα εξηγήσουμε την μέθοδο του Cross Validation που λαμβάνει υπόψη και το 1-se (standard error), ή αλλιώς one standard error rule. Κάθε Cross Validation run εφαρμόζεται με ένα k-fold, δηλαδή χωρίζουμε το dataset σε k τμήματα και εφαρμόζουμε κάθε υποψήφιο μοντέλο (δηλαδή κάθε τιμή του λ) σε όλα τα k τμήματα (η cv.glmnet έχει default τιμή $k = 10$). Από αυτή τη διαδικασία η μέση τιμή του σφάλματος μας δίνει την τιμή του σφάλματος για το συγκεκριμένο μοντέλο. Όταν πραγματοποιηθεί η διαδικασία για όλα τα μοντέλα, τότε έχουμε μια συλλογή από σφάλματα και το ελάχιστο από όλα αυτά είναι το $min(CV_{error})$. Παρόλα αυτά μπορούμε από την συλλογή σφαλμάτων να υπολογίσουμε το τυπικό σφάλμα $se(CV_{error})$. Στη μέθοδο του one stardard error rule θέτουμε το διάστημα :

$$[min(CV_{error}) - se(CV_{error}), min(CV_{error}) + se(CV_{error})]$$

και επιλέγουμε το μεγαλύτερο λ που το CV_{error} του είναι μέσα σε αυτό το διάστημα. Με αυτό τον τρόπο θεωρούμε ότι έχουμε επιλέξει ένα μοντέλο με ακόμα πιο αυστηρή πολιτική ποινικοποίησης που ανταπεξέρχεται στο πρόβλημα ελαχιστοποίησης της (4.1) περίπου με την ίδια επιτυχία με το βέλτιστο.

Στη συνέχεια θα κατασκευάσουμε το διάγραμμα που δείχνει το CV_{MSE} του μοντέλου για διάφορες τιμές του λ.

```
X_inp1<-as.matrix(X_inp)
lasso1 <- cv.glmnet(X_inp1, Y_inp[,'Y'])
plot(lasso1)
```



Στο παραπάνω διάγραμμα βλέπουμε τη σχέση του MSE με το LogLamba, και η πρώτη κάθετη διακεκκομένη δείχνει την τιμή του LogLamda που αντιστοιχεί στο ελάχιστο MSE και η δεύτερη διακεκκομένη γραμμή την τιμή του LogLamda για που αντιστοιχεί στο one standard error rule που εξηγήσαμε παραπάνω.

Στη συνέχεια θα παρουσιάσουμε το λ που προκύπτει από το CV, με και χωρίς το one standard error rule.

```
lasso1$lambda.min
log(lasso1$lambda.min)
lasso1$lambda.1se
log(lasso1$lambda.1se)
```

- $min(CV_{error})$ Lambda = 0.08842548
- $min(CV_{error})$ LogLambda = -2.425595
- $min(CV_{error})$, 1-se rule Lamda = 0.3917798
- $min(CV_{error})$, 1-se rule LogLamda = -0.9370552

Από τα παραπάνω αποτελέσματα αποφασίζουμε να επιλέξουμε το 1-se rule Lamda = 0.3917798 για να υλοποιήσουμε το μοντέλο μας και να προχωρήσουμε στην ανάλυση μας, φυσικά η διαδικασία που ακολουθεί μπορεί να πραγματοποιηθεί με το Lambda = 0.08842548 και αφήνεται στην κρίση του καθενός.

Τελευταίο βήμα της ανάλυσης είναι να βρεθεί η παράμετρος συρρίκνωσης, όπως επίσης θα παρουσιάσουμε τα coefficients που προέκυψαν για κάθε μεταβλητή.

```
blasso_1se <- coef(lasso1)
zblasso_1se <- blasso_1se[-1] * apply(X_inp, 2, sd)
zbols_1se <- coef(reg)[-1] * apply(X_inp, 2, sd)
s_1se <- sum(abs(zblasso_1se)) / sum(abs(zbols_1se))
print(s_1se)

blasso_1se
```

Προκύπτει ότι $\mathbf{S = 0.6017031}$. Και τα coefficients είναι :

Lasso	
Intercept	3.69445683
X_1	1.97785509
X_2	0
X_3	0
X_4	0
X_5	0
X_6	0
X_7	2.26926465
X_8	0
X_9	0.08040966
X_{10}	0
X_{11}	1.10537179
X_{12}	0
X_{13}	0.28198869
X_{14}	0
X_{15}	0

OLS	
Intercept	3.66899827
X_1	2.23097540
X_2	0.12617908
X_3	-0.24812484
X_4	-0.19606963
X_5	-0.53277546
X_6	-0.14462129
X_7	2.55469554
X_8	-0.00159201
X_9	0.28553912
X_{10}	0.21262308
X_{11}	1.39449109
X_{12}	-0.21003082
X_{13}	0.74577372
X_{14}	0.11785714
X_{15}	-0.18765070

Αρχικά θα σχολιάσουμε ότι το λ είναι μικρή τιμή και συνεπώς το S είναι σχετικά μεγάλη τιμή, αυτό έχει ως αποτέλεσμα να μην έχουμε μεγάλη συρρίκνωση, δηλαδή να μην τείνουν τα coefficients στο 0 με μεγάλη αυστηρότητα. Αυτό παρατηρούμε και όταν συγκρίνουμε τους 2 πίνακες, καθώς βλέπουμε ότι οι πολύ μικρές τιμές έχουν πάει στο 0 αλλά οι υπόλοιπες έχουν διατηρηθεί. Τέλος, συγκρίνοντας τις δύο μεθόδους BIC και Lasso παρατηρούμε συμφωνία στις επεξηγηματικές μεταβλητές X_1, X_7, X_{11}, X_{13}, αλλά το BIC κριτήριο ανέδειξε και την X_5, ενώ η μέθοδος Lasso και την X_9, αν και η τιμή που της έχει ανατεθεί είναι πάρα πολύ μικρή. Η ασυμφωνία αυτή οφείλεται στη διαφορετική προσέγγιση κάθε μεθόδου αλλά μεγάλο ρόλο έχει και η τυχαιότητα στη παραγωγή των δειγμάτων.

Βιβλιογραφία

[1] https://math.stackexchange.com/questions/353957/covariance-of-a-normal-with-its-square. [Online].

[2] https://en.wikipedia.org/wiki/Normal_distribution#Moments. [Online].

[3] https://tinyurl.com/4k9khswf. [Online].

[4] https://en.wikipedia.org/wiki/Law_of_total_probability. [Online].

[5] https://tinyurl.com/bcdfwdwt. [Online].

[6] George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2021.

[7] Giovanni Di Leo and Francesco Sardanelli. "Statistical significance: p value, 0.05 threshold, and applications to radiomics—reasons for a conservative approach". In: *European radiology experimental* 4.1 (2020), pp. 1–8.

[8] Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.