

Κεφάλαιο 6

Προσομοίωση

Αυτό το κεφάλαιο συμπληρώνει το προηγούμενο κεφάλαιο το οποίο αναφερόταν στο πώς μπορεί να γίνει απλός προγραμματισμός στην R. Θα χρησιμοποιηθούν οι έννοιες του προγραμματισμού για να εξαχθούν αποτελέσματα από απλές προσομοιώσεις γνωστών πιθανοθεωρητικών αποτελεσμάτων.

6.1 Ο Ασθενής Νόμος των Μεγάλων Αριθμών

Σύμφωνα με τον ασθενή νόμο των μεγάλων αριθμών, αν X_1, \dots, X_n είναι ανεξάρτητες και ισόνομες τυχαίες μεταβλητές με πεπερασμένη μέση τιμή μ , τότε

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \rightarrow \mu,$$

κατά πιθανότητα, όταν το $n \rightarrow \infty$. Συγκεκριμένα, αν X_1, \dots, X_n είναι δίτιμες τυχαίες μεταβλητές με $P(X_i = 1) = p$, τότε

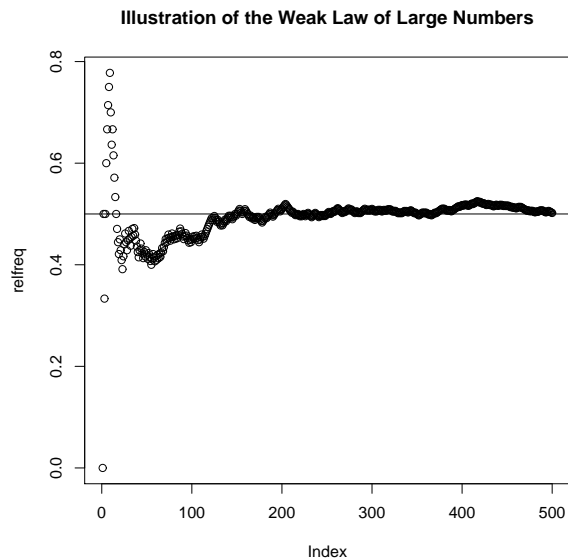
$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \rightarrow p,$$

κατά πιθανότητα, όταν το $n \rightarrow \infty$. Το αποτέλεσμα αυτό μπορεί να παρουσιαστεί εμπειρικά στην R με τις ακόλουθες συναρτήσεις :

```
uniforms <- runif(500)
tosses <- as.numeric(I(uniforms > 0.5))
relfreq <- cumsum(tosses)/(1:500)
plot(relfreq)
```

```
abline(0.5,0)
title(main="Illustration of the Weak Law of Large Numbers")
```

Η πρώτη εντολή δίνει τυχαίο δείγμα U_1, \dots, U_{500} από την ομοιόμορφη στο $(0,1)$. Στη συνέχεια ορίζουμε τη δίτιμη τυχαία μεταβλητή $X_i = I(U_i > \frac{1}{2})$, όπου I δείκτρια, $i = 1, \dots, 500$. Μετά θεωρούμε τη συνάρτηση \bar{X} σαν ακολουθία, δηλαδή το ακολουθιακό ποσοστό επιτυχιών (γιατί ;). Το γράφημα (Σχήμα 6.1) μας δίνει την σύγκλιση της ακολουθίας στο 0.5, όταν $n \rightarrow \infty$ σύμφωνα με το νόμο των μεγάλων αριθμών.



Σχήμα 6.1: Ο ασθενής νόμος των μεγάλων αριθμών για δυαδικές τυχαίες μεταβλητές.

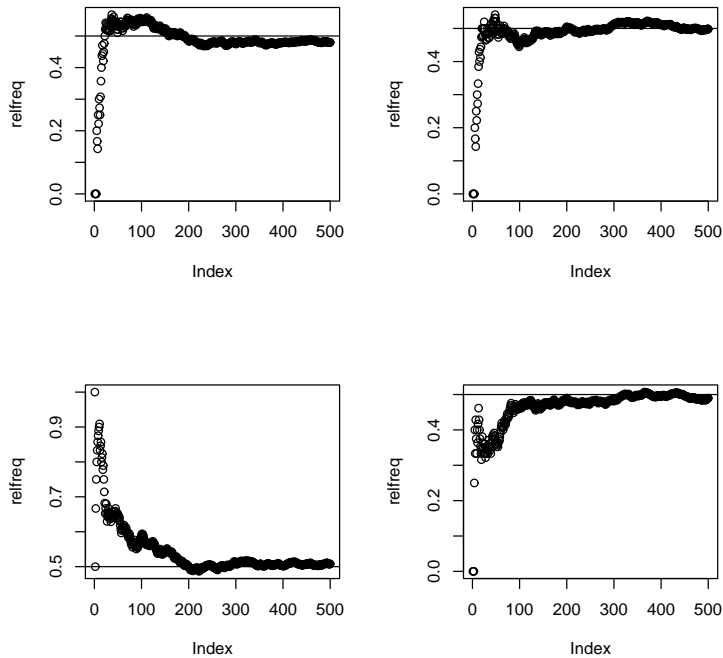
Εφαρμόζοντας τα πιο πάνω τέσσερις φορές και απεικονίζοντας τις γραφικές παραστάσεις σε ένα 2×2 γράφημα (Σχήμα 6.2), έχουμε ότι,

```
par(mfrow=c(2,2))
for(rep in 1:4)
{
  uniforms <- runif(500)
  tosses <- as.numeric(uniforms > 0.5)
  relfreq <- cumsum(tosses)/(1:500)
```

```

plot(relfreq)
abline(0.5,0)
}

```



Σχήμα 6.2: Ο ασθενής νόμος των μεγάλων αριθμών για δυαδικές τυχαίες μεταβλητές.

Δηλαδή το γράφημα αυτό δείχνει καθαρά την τυχειότητα αλλά και τη σύγκλιση. Τι συμβαίνει όμως όταν η αναμενόμενη τιμή της τυχαίας μεταβλητής δεν υπάρχει; Ένα πολύ γνωστό παράδειγμα μιας τέτοιας τυχαίας μεταβλητής είναι η κατανομή Cauchy

$$f(x) = \frac{1}{\pi(1+x^2)}, \quad x \in R,$$

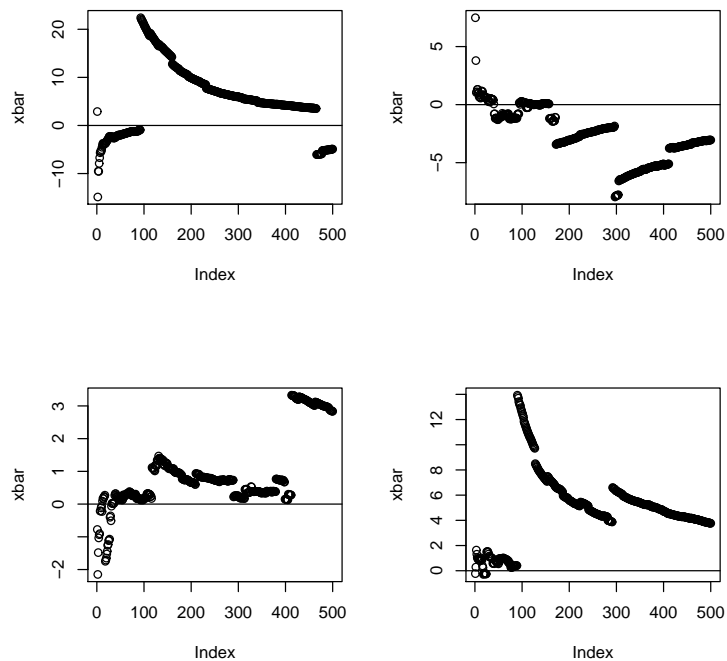
της οποίας η αναμενόμενη τιμή δεν υπάρχει (γιατί:). Οι ακόλουθες συναρτήσεις μαζί με το Σχήμα 6.3 καταδεικνύουν ότι η ακολουθία των μέσων τιμών δεν συγκλίνει.

```
par(mfrow=c(2,2))
```

```

for(rep in 1:4)
{
  cauchys <- rcauchy(500)
  xbar    <- cumsum(cauchys)/(1:500)
  plot(xbar)
  abline(0,0)
}

```



Σχήμα 6.3: Ακολουθία μέσων τιμών από την κατανομή Cauchy.

6.2 Κεντρικό Οριακό Θεώρημα

Έστω X_1, \dots, X_n τυχαίες μεταβλητές με πεπερασμένη μέση τιμή μ και διασπορά σ^2 . Τότε, σύμφωνα με το κεντρικό οριακό θεώρημα

$$\sqrt{n}(\bar{X} - \mu) \Rightarrow \mathcal{N}(0, \sigma^2),$$

κατά κατανομή, όταν το $n \rightarrow \infty$. Στο πιο πάνω, το \mathcal{N} συμβολίζει την κανονική κατανομή. Η προσομοίωση μπορεί να βοηθήσει διαισθητικά στην κατανόηση του θεωρήματος.

Έστω X_1, \dots, X_{100} ανεξάρτητες και ισόνομες τυχαίες κατανομές από την Poisson με παράμετρο $\lambda = 1$. Τότε $\mu = \sigma^2 = 1$ και συνεπώς, από το κεντρικό οριακό θεώρημα

$$E(\bar{X}) = 1$$

και

$$\text{Var}(\bar{X}) = 1/100 = 0.01.$$

Στην R, η πιο κάτω συνάρτηση παράγει δείγματα από την ασυμπτωτική κατανομή της μέσης τιμής

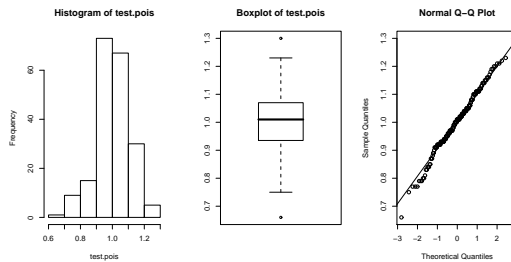
```
poisson.clt <- function(k,n, parameter)
{
  samples.mean <- rep(NA, k)
  for (i in 1:k)
  {
    samples.mean[i] <- mean(rpois(n, lambda=parameter))
  }
  return(samples.mean)
}
```

Τρέχοντας αυτή την συνάρτηση παράγονται τα ακόλουθα αποτελέσματα:

```
> test.pois <- poisson.clt(200,100,1)
> summary(test.pois)
  Min. 1st Qu. Median Mean 3rd Qu. Max.
  0.69  0.93     1 1.001  1.072 1.31
> var(test.pois)
[1] 0.009584601
> par(mfrow=c(1,3))
> hist(test.pois)
> boxplot(test.pois)
> qqnorm(test.pois)
> qqline(test.pois)
```

Δηλαδή, δημιουργούμε k δείγματα από την Poisson, και για καθένα από αυτά υπολογίζουμε το μέσο όρο τους, έστω, $\bar{X}_1, \dots, \bar{X}_k$. Στο παράδειγμα $k = 200$ και

το δείγμα που παίρνουμε κάθε φορά έχει μέγεθος 100. Να εξηγήσετε λεπτομερώς κάθε βήμα του προγράμματος. Τα αριθμητικά αποτελέσματα μαζί με το Σχήμα 6.4 παρουσιάζουν εμπειρικά το κεντρικό οριακό θεώρημα.



Σχήμα 6.4: Κεντρικό οριακό θεώρημα για την κατανομή Poisson.

Πρέπει να σημειωθεί ότι η συνάρτηση `poisson.clt` δεν είναι και ο πιο αποτελεσματικός τρόπος προγραμματισμού, αλλά παρουσιάζει την γενική ιδέα πίσω από τους υπολογισμούς. Για πιο αποτελεσματική χρήση των βρόγχων σε σχέση με την μνήμη και τον υπολογιστικό χρόνο του υπολογιστή, η συνάρτηση `lapply` είναι καταλληλότερη.

6.3 Προσέγγιση της Διωνυμικής Κατανομής από την Κανονική και την Poisson

Σε αυτό το σημείο θα εξερευνηθεί το πως προσεγγίζεται η διωνυμική κατανομή με τη βοήθεια του κεντρικού οριακού θεωρήματος αλλά και από την κατανομή Poisson. Έστω η διωνυμική κατανομή $Bin(n, p)$. Από τη θεωρία είναι γνωστό ότι αυτή προσεγγίζεται από

- την κατανομή Poisson όταν το n είναι μεγάλο και το p είναι μικρό και
- την κανονική κατανομή όταν το n είναι μεγάλο.

Στην προσομοίωση που ακολουθεί θα δούμε πόσο καλή είναι η προσέγγιση για διάφορες τιμές του n και του p . Στην αρχή επιλέγονται οι τιμές για το n και το p να είναι:

`p<-c(0.01,0.1,0.3,0.5)`

`n<-c(10,100,1000,10000)`

Για κάθε συνδυασμό (n, p) θα συγκριθούν τρεις κατανομές, η διωνυμική κατανομή, η προσέγγιση από την Poisson, και η προσέγγιση από την κανονική. Για καλύτερη εποπτική ανάλυση θα παρασταθεί γραφικά η συνάρτηση πυκνότητας πιθανότητάς τους ταυτόχρονα στο ίδιο γράφημα. Θα κατασκευαστεί ένα γράφημα για κάθε συνδυασμό (n, p) .

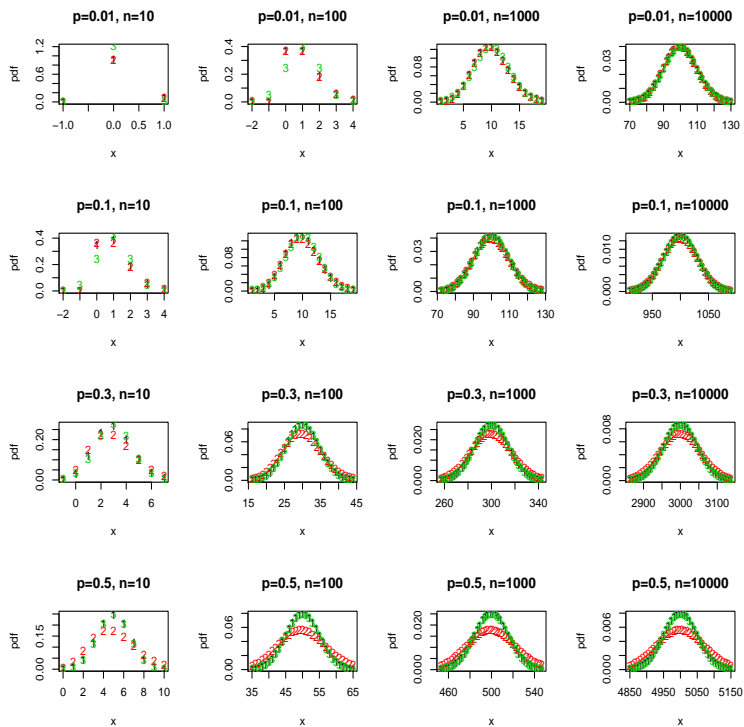
```
par (mfrow=c(4,4))
for (i in 1:4)
{
  for (j in 1:4)
  {
    mu <- n[j]*p[i]
    sd <- sqrt(mu * (1-p[i]))
    lo <- round(mu-3*sd)
    hi <- round(mu+3*sd)
    if (hi-lo<40)
    x <- seq(lo,hi,by=1) else
    x <- round(seq(lo,hi,len=40))
    pdf <- cbind(dbinom(x,n[j],p[i]),dpois(x,mu),dnorm(x,mu,sd))
    pdf[x<0,1:2]<- 0
    matplot(x,pdf,main=paste("p=", p[i],",", n=",n[j],sep=""))
  }
}
```

Στο πιο πάνω πρόγραμμα, pdf είναι ένας πίνακας με τρεις στήλες οι οποίες περιέχουν τις συναρτήσεις πυκνότητας πιθανότητας πιθανότητας της διωνυμικής, της Poisson και της κανονικής, αντίστοιχα. Η εντολή matplot κατασκευάζει γράφημα πίνακα. Παρουσιάζει τη γραφική παράσταση της κάθε στήλης συναρτήσεως του x . Η πρώτη στήλη αναπαρίστανται με το σύμβολο "1", η δεύτερη με το σύμβολο "2" και η τρίτη με το σύμβολο "3" (Σχήμα 6.5).

Πρώτα θα εξεταστεί στο γράφημα η προσέγγιση από την Poisson. Σε κάθε στήλη, η προσέγγιση από την Poisson είναι μεγαλύτερη στο πάνω μέρος της στήλης όπου το p είναι μικρό, και σταδιακά γίνεται ασθενέστερη όσο το p μεγαλώνει (πηγαίνοντας προς τα κάτω). Ο λόγος είναι ότι η μέση τιμή της διωνυμικής κατανομής είναι np και η διακύμανση $np(1-p)$. Παρόλο που η μέση τιμή και η διακύμανση δεν είναι ίσες, προσεγγιστικά γίνονται ίσα όταν το p είναι πολύ μικρό. Για την Poisson με παράμετρο λ , η μέση τιμή και η διακύμανση είναι ίση με λ . Συνεπώς, η κατανομή Poisson δεν μπορεί να είναι καλή προσέγγιση μιας κατανομής της

οποίας η μέση τιμή και η διακύμανση είναι πολύ διαφορετικές μεταξύ τους και έτσι η προσέγγιση είναι καλή για τη διωνυμική μόνο όταν το p είναι μικρό.

Στην περίπτωση της προσέγγισης από την κανονική κατανομή, η προσέγγιση δεν είναι καλή στο πάνω αριστερό κομμάτι του γραφήματος. Συγκεκριμένα, όταν ($p = .01, n = 100$) ή ($p = .1, n = 10$), η προσέγγιση από την κανονική επιτρέπει στο x να είναι αρνητικό ενώ η διωνυμική (και η Poisson) κατανομή απαιτεί το x να μην παίρνει αρνητικές τιμές. Σε κάθε γραμμή, η προσέγγιση από την κανονική γίνεται καλύτερη όσο προχωρούμε διαμέσου της γραμμής. Αυτό είναι το αποτέλεσμα από το κεντρικό οριακό θεώρημα. Επίσης, σε κάθε στήλη, η προσέγγιση από την κανονική γίνεται καλύτερη όσο προχωρούμε προς τα κάτω. Αυτό συμβαίνει επειδή η διωνυμική κατανομή είναι συμμετρική όταν $p = 0.5$, δηλαδή έχει μορφή παρόμοια με την κανονική, και συνεπώς δεν χρειάζεται μεγάλο n για να είναι καλή η προσέγγιση από την κανονική. Σε αντίθεση, η διωνυμική με $p = 0.01$ είναι πολύ λοξή, δηλαδή δεν μοιάζει με την κανονική, και συνεπώς χρειάζεται μεγάλο n για να γίνει καλή η προσέγγιση από την κανονική.



Σχήμα 6.5: Προσέγγιση της διωνυμικής από την Poisson και την κανονική.

6.4 Monte Carlo Ολοκλήρωση

Έστω ότι είναι αναγκαίο να εκτιμηθεί η αναμενόμενη τιμή της τυχαίας μεταβλητής X με συνάρτηση πυκνότητας πιθανότητας $f(x)$ δεδομένου ότι αυτή υπάρχει. Συγκεκριμένα, έστω ότι το δ μπορεί να οριστεί από

$$\delta = \int c(x)f(x)dx = \mathbf{E}_f [c(X)],$$

και υποθέστε ότι υπάρχει και είναι πεπερασμένο. Υπάρχουν διάφοροι μέθοδοι για υπολογισμό του δ και ίσως η πιο γνωστή ανάμεσα στους στατιστικούς είναι αυτές που βασίζονται στη Monte Carlo ολοκλήρωση. Ανάλογα, αν X_1, \dots, X_n τυχαίο δείγμα από την $f(x)$, τότε σύμφωνα με τον ασθενή νόμο των μεγάλων αριθμών, η εκτιμήτρια

$$\hat{\delta} = \frac{1}{n} \sum_{i=1}^n c(X_i)$$

προσεγγίζει το δ με μεγάλη πιθανότητα όταν το n τείνει στο άπειρο.

Παρατίθεται πως μπορεί η R να χρησιμοποιηθεί για να υπολογίσει τέτοια ολοκληρώματα. Έστω ότι η X είναι τυχαία μεταβλητή που ακολουθεί τη κατανομή Beta με συνάρτηση πυκνότητας πιθανότητας

$$f(x) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1}$$

για $x \in (0, 1)$ και έστω ότι είναι αναγκαίο να υπολογιστούν τα ακόλουθα δύο ολοκληρώματα:

$$\delta_1 = \int_{0.2}^{0.4} f(x)dx = \int I_{[0.2, 0.4]} f(x)dx,$$

όπου I είναι η δείκτρια συνάρτηση, και

$$\delta_2 = \int \sin(x)e^{-x} f(x)dx$$

όπου υποθέτουμε ότι η f είναι η συνάρτηση πυκνότητας πιθανότητας της κατανομής Βήτα με παράμετρους 2.5 και 5. Τότε η ακόλουθη συνάρτηση δίνει τα επιθυμητά αποτελέσματα σε μορφή πίνακα.

```
estim.beta <- function(k)
{
  delta1 <- rep(NA, k)
  delta2 <- rep(NA, k)
  for (i in 1:k)
```

```

{
x  <- rbeta(500, shape1=2.5, shape2=5)
delta1[i] <- mean(as.numeric(I(0.2 <x & x<0.4)))
delta2[i] <- mean(sin(x)*exp(-x))
}
return(cbind(delta1,delta2))
}

```

Να εξηγήσετε λεπτομερώς τι κάνει κάθε βήμα στην πιο πάνω συνάρτηση. Τρέχοντας τη συνάρτηση παίρνονται τα ακόλουθα

```

> delta.estim <- estim.beta(500)
> apply(delta.estim,2,mean)
delta1 delta2
0.23072 0.2172346
> apply(delta.estim,2,var)
delta1 delta2
0.0003627351 9.412981e-006
> sqrt(apply(delta.estim,2,var))
delta1 delta2
0.01904561 0.003068058

```

Συνεπώς, το δ_1 εκτιμάται να είναι ίσο με 0.03086395 με τυπικό σφάλμα 0.0027, ενώ το δ_2 εκτιμάται να είναι ίσο με 0.08337394 με τυπικό σφάλμα 0.0025.

Παρόλο που εδώ υπάρχουν διάφορες συνηθισμένες μέθοδοι για να παραχθούν ψευδο-τυχαία αποτελέσματα για αρκετές κατανομές, συνήθως ένα τέτοιο εγχείρημα μπορεί να είναι αρκετά απαιτητικό εξαιτίας της μορφής της συνάρτησης πυκνότητας πιθανότητας. Για αυτό το λόγο κάποιες άλλες τεχνικές μπορούν να χρησιμοποιηθούν εναλλακτικά και μια από τις πιο δημοφιλείς μεθόδους προσομοίωσης είναι η *importance sampling* η εφαρμογή της οποίας παρουσιάζεται πιο κάτω:

- Γέννηση Z_1, \dots, Z_n ανεξάρτητων και ισόνομων τυχαίων μεταβλητών με συνάρτηση πυκνότητα πιθανότητας $g(z)$ των οποίων ο φορέας, έστω A , περιέχει το φορέα της $f(\cdot)$.
- Αφού παρατηρηθεί ότι

$$\delta = \int c(z) \frac{f(z)}{g(z)} g(z) dz$$

$$= \int c(z)w(z)g(z)dz = \mathbf{E}_g [c(Z)w(Z)],$$

με $w = f/g$, κατασκευάζεται η ακόλουθη εκτιμήτρια

$$\tilde{\delta} = \frac{1}{n} \sum_{i=1}^n c(Z_i)w(Z_i).$$

Προφανώς, η εκτιμήτρια $\tilde{\delta}$ μιμείται την εκτιμήτρια $\hat{\delta}$ στην έννοια ότι η αναμενόμενη τιμή σε σχέση με το X ακολουθώντας την f , αντικαθίσταται από την αντίστοιχη αναμενόμενη τιμή σε σχέση με το Z , ακολουθώντας την g . Θα ήταν εκπαιδευτικό σκόπιμο να προσπαθήσει ο αναγνώστης να χρησιμοποιήσει την R για να υπολογίσει το $\tilde{\delta}$ για το πιο πάνω παράδειγμα όταν η g είναι η συνάρτηση πυκνότητας από την ομοιόμορφη κατανομή.

6.5 Βελόνα του Buffon

Ένα τραπέζι χωρίζεται σε παράλληλες ευθείες οι οποίες απέχουν d μονάδες μεταξύ τους. Ρίχνουμε μία βελόνα μήκους l στο τραπέζι (με $l \leq d$) n φορές και μετράμε R τον αριθμό των φορών που η βελόνα τέμνει μία ευθεία. Έστω X η απόσταση από το κέντρο της βελόνας στην πιο κοντινή παράλληλη ευθεία και θ η γωνία που σχηματίζει η κάθετη ευθεία από το κέντρο της βελόνας στην πιο κοντινή παράλληλη ευθεία. Τότε η βελόνα θα τέμνει μία από τις παράλληλες ευθείες αν και μόνο αν

$$\frac{x}{\cos \theta} \leq \frac{l}{2}.$$

Όμως, αφού η X μεταβάλλεται μεταξύ 0 και $d/2$ και η θ είναι μεταξύ 0 και $\pi/2$, μπορούμε να υποθέσουμε ότι είναι ανεξάρτητες τυχαίες μεταβλητές από την ομοιόμορφη. Συνεπώς, έχουμε

$$\begin{aligned} P\left(X \leq \frac{l \cos \theta}{2}\right) &= \frac{4}{\pi d} \int_0^{\pi/2} \int_0^{l \cos y/2} dx dy \\ &= \frac{4}{\pi d} \int_0^{\pi/2} \frac{l \cos y}{2} dy \\ &= \frac{2l}{\pi d}. \end{aligned}$$

Συνεπώς, αν $\rho = l/d$, και $\phi = 1/\pi$, μία εκτιμήτρια του π δίνεται από

$$\hat{\pi}_0 = \frac{1}{\hat{\phi}_0} = \frac{2\rho}{\hat{\rho}}$$

όπου $\hat{p} = R/n$.

Μία βασική ερώτηση είναι πώς να βελτιστοποιήσουμε τις τιμές των l και d για να ελαχιστοποιήσουμε την διακύμανση της εκτιμήτριας $\hat{\phi}_0$. Επειδή η R είναι διωνυμική τυχαία μεταβλητή έχουμε ότι $\text{Var}(\hat{p}) = p(1-p)/n$. Συνεπώς, $\text{Var}(\hat{\phi}_0) = 2\rho\phi(1 - 2\rho\phi)/4\rho^2n$ και αυτή η ποσότητα ελαχιστοποιείται αν $\rho = 1$ ή $l = d$.

Το παρακάτω πρόγραμμα δίνει ακριβώς την παραπάνω μεθοδολογία.

```
buf<-function(n, d,l) # n is the number of simulations,
                      #d is the distance between the lines
{
                      # and l is the needle's length (l =< d).

    R                <- rep(NA, n)
    x                <- runif(n, 0, d/2)
    theta           <- runif(n, 0, pi/2)
    y                <- (1/2)*cos(theta)
    R               <- ifelse(y > x,1,0)
    pi              <- cumsum(R)/(1:n)
    rho             <- l/d
    phi             <- pi/(2*rho)
    pi.hat          <- 1/phi
    pi.hat
    plot(1:n, pi.hat, type="l", xlab="Number of Simulations",
         ylab="Proportion of Hits")
}
```

6.6 Εμπειρική Σύγκριση Εκτιμητριών

Η προσομοίωση μπορεί να χρησιμοποιηθεί για εμπειρική σύγκριση διαφόρων εκτιμητριών οι οποίες χρησιμοποιούνται για την εκτίμηση συγκεκριμένης παραμέτρου. Ας υποθέσουμε ότι στο Πανεπιστήμιο φοιτούν 3000 φοιτητές εκ των οποίων το 30% είναι μέλη συνδικαλιστικών οργανώσεων ενώ το άλλο 70% είναι ανεξάρτητοι. Υπάρχει μία μελλοντική εκλογή προέδρου των φοιτητών και ας υποθέσουμε ότι δύο ανεξάρτητοι υποψήφιοι, ο Α και Β, διεκδικούν την εκλογή. Έστω

θ_U = ποσοστό μη ανεξάρτητων φοιτητών που υποστηρίζουν τον Α

θ_I = ποσοστό ανεξάρτητων φοιτητών που υποστηρίζουν τον Α

θ = ποσοστό φοιτητών που υποστηρίζουν τον A

Γίνεται δειγματοληπτική έρευνα σε 100 φοιτητές για την εκτίμηση της παραμέτρου θ και προτείνονται 3 διαφορετικές μέθοδοι εκτίμησης.

1. Τυχαία επιλογή 100 φοιτητών και εκτίμηση μέσω της στατιστικής συνάρτησης

$$\hat{\theta}_1 = \text{ποσοστό φοιτητών που υποστηρίζουν τον A.}$$

2. Τυχαία επιλογή 100 φοιτητών και εκτίμηση μέσω της στατιστικών συναρτήσεων

$$\hat{\theta}_U = \text{ποσοστό μη ανεξάρτητων φοιτητών που υποστηρίζουν τον A στο δείγμα}$$

$$\hat{\theta}_I = \text{ποσοστό ανεξάρτητων φοιτητών που υποστηρίζουν τον A στο δείγμα}$$

$$\hat{\theta}_2 = 0.30\hat{\theta}_U + 0.70\hat{\theta}_I$$

3. Επιλογή 30 μη ανεξάρτητων και 70 ανεξάρτητων. Τότε

$$\hat{\theta}_U = \text{ποσοστό από μη ανεξάρτητους φοιτητές που υποστηρίζουν τον A}$$

$$\hat{\theta}_I = \text{ποσοστό από ανεξάρτητους φοιτητές που υποστηρίζουν τον A}$$

$$\hat{\theta}_3 = 0.30\hat{\theta}_U + 0.70\hat{\theta}_I$$

Ποια διαδικασία είναι η καλύτερη; Αν και μπορούμε να απαντήσουμε θεωρητικά στην ερώτηση, εδώ θα δούμε πώς μπορεί να μας βοηθήσει η προσομοίωση. Αρκεί να επαναλάβουμε κάθε διαδικασία αρκετές φορές και μετά να εξετάσουμε πόσο ακριβή είναι τα αποτελέσματα. Πρέπει να επιλέξουμε τις αληθινές τιμές των παραμέτρων φυσικά και η όλη θεωρία προσομοιώνεται με τον παρακάτω κώδικα. Τα αποτελέσματα δίνουν την γραφική παράσταση 6.6.

```
# choose "true" theta.u and theta.i
theta.u <- .8
theta.i <- .4
prop.u <- .3
prop.i <- 1 - prop.u
theta <- prop.u * theta.u + prop.i * theta.i

sim.1 <- function() {
x <- rbinom(1,sampsize,theta)
```

```

return ( x / sampsize )
}

sim.2 <- function() {
n.u <- rbinom ( 1, sampsize, prop.u )
n.i <- sampsize - n.u
x.u <- rbinom ( 1, n.u, theta.u )
x.i <- rbinom ( 1, n.i, theta.i )
t.hat.u <- x.u / n.u
t.hat.i <- x.i / n.i
return ( prop.u * t.hat.u + (1-prop.u) * t.hat.i )
}

sim.3 <- function() {
n.u <- sampsize * prop.u
n.i <- sampsize * prop.i
x.u <- rbinom ( 1, n.u, theta.u )
x.i <- rbinom ( 1, n.i, theta.i )
t.hat.u <- x.u / n.u
t.hat.i <- x.i / n.i
return ( prop.u * t.hat.u + (1-prop.u) * t.hat.i )
}

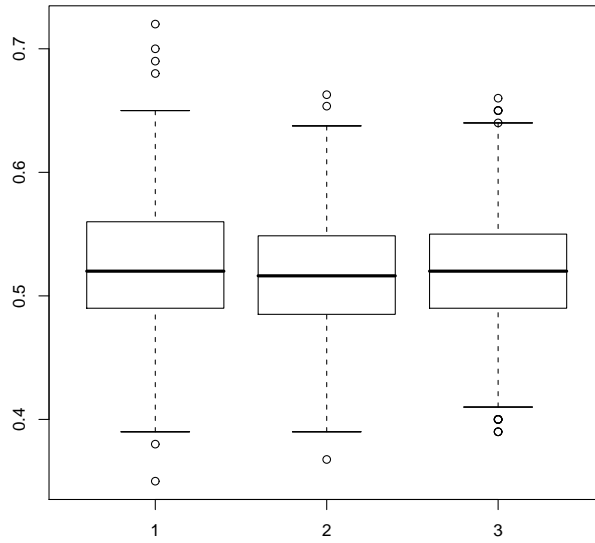
sampsize <- 100
n.times <- 1000 # should be enough
theta.hat <- matrix ( NA, n.times, 3 )
for ( i in 1:n.times ) {
theta.hat[i,1] <- sim.1()
theta.hat[i,2] <- sim.2()
theta.hat[i,3] <- sim.3()
}

print ( apply(theta.hat,2,mean) )

[1] 0.5228600 0.5178319 0.5189100

boxplot ( theta.hat ~ col(theta.hat) )

```



Σχήμα 6.6: 1000 προσομοιώσεις της $\hat{\theta}$ για τρεις διαφορετικές μεθόδους εκτίμησης.

