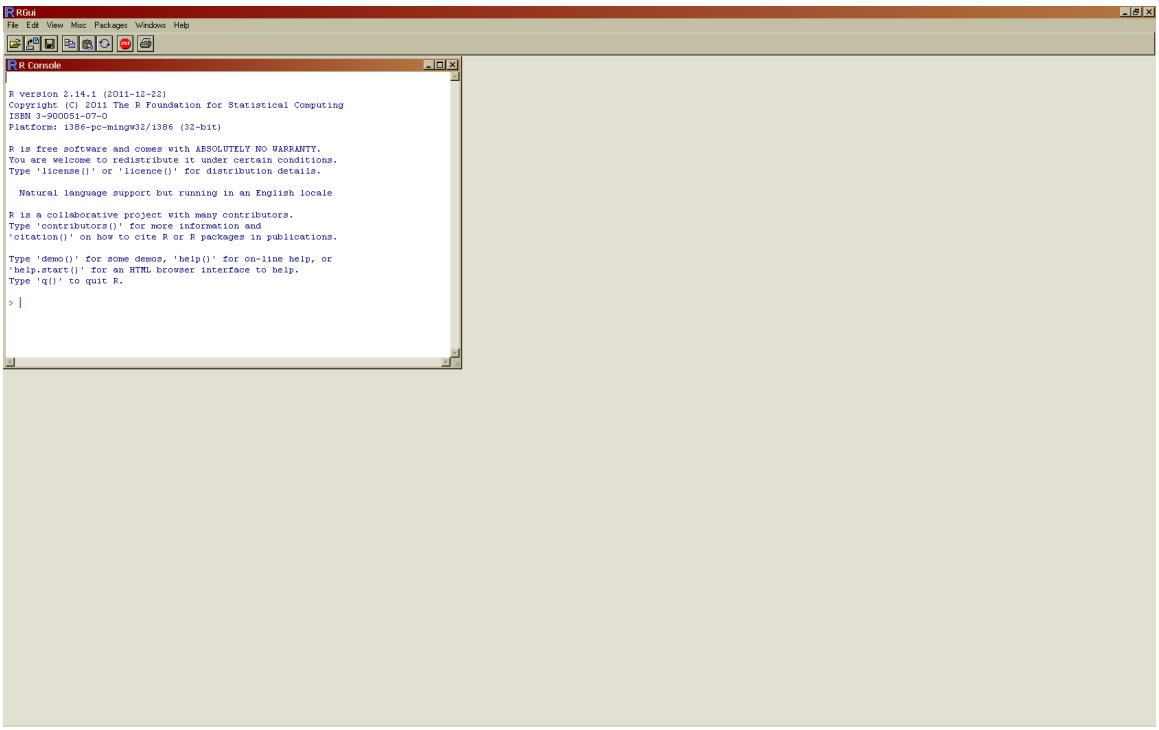
Οδηγίες χρήσης του R, μέρος 1^{ο}

Προκαταρκτικά

Κατεβάζουμε το λογισμικό από την ιστοσελίδα http://cran.cc.uoc.gr/bin/windows/base/

Εγκαθιστούμε το λογισμικό στον υπολογιστή μας εκτελώντας το αρχείο που κατεβάσαμε.

Τρέχουμε το λογισμικό με διπλό κλικ στο μπλε εικονίδιο και βλέπουμε το περιβάλλον του R:



Αρχικά θα πληκτρολογούμε τις εντολές μας στο λευκό παράθυρο. Αργότερα θα αποθηκεύουμε σειρές εντολών σε ξεχωριστά αρχεία, ώστε να έχουμε πρόσβαση σε προηγούμενες εργασίες.

Βασικές αρχές

Α. Αριθμητικές πράξεις

Το R εκτελεί αριθμητικές πράξεις γράφοντας απλώς τη σχετική παράσταση. Π.χ., αν γράψουμε 2+3 θα μας δώσει το αποτέλεσμα 5:

```
> 2+3
[1] 5
>
```

Αυτά που γράφουμε εμείς εμφανίζονται κόκκινα, ενώ οι απαντήσεις του λογισμικού είναι μπλε, ώστε να τις ξεχωρίζουμε εύκολα. Το [1] σημαίνει ότι βλέπουμε το *πρώτο* στοιχείο της απάντησης στο ερώτημά μας (τον αριθμό 5), που στην περίπτωση αυτή είναι και το μοναδικό.

Παρομοίως μπορούμε να κάνουμε πρόσθεση, αφαίρεση κλπ.:

```
> (8-5)*2/17
[1] 0.3529412
>
```

Για να υποδείξουμε την επιθυμητή σειρά των πράξεων, χρησιμοποιούμε παρενθέσεις. Στο παραπάνω παράδειγμα, θέλουμε η διαφορά 8−5 να υπολογιστεί πρώτα, και μετά το αποτέλεσμα να πολλαπλασιαστεί με το δύο.
Τα σύμβολα των πράξεων είναι: πρόσθεση +, αφαίρεση −, πολλαπλασιασμός *, διαίρεση /
Για να υψώσουμε σε δύναμη, χρησιμοποιούμε το σύμβολο ^ :

```
> 2^3
[1] 8
>
```

Το 2 στην $3^{η}$ δύναμη, δηλαδή πολλαπλασιασμένο με τον εαυτό του τρεις φορές, είναι 2×2×2=8.

Β. Μεταβλητές

Το R χρησιμοποιεί μεταβλητές, στις οποίες μπορούμε να καταχωρίσουμε αριθμητικές ή κατηγορικές τιμές. Για παράδειγμα, μπορούμε να αποθηκεύσουμε την τιμή 5 στη μεταβλητή a:

```
> a <- 5
>
```

Το σύμβολο <− δείχνει πως ο αριθμός 5 «τοποθετείται» στην ετικέτα a. Το ίδιο ακριβώς αποτέλεσμα μπορούμε να πετύχουμε και γράφοντας πρώτα την τιμή, αρκεί το βέλος να δείχνει προς τη σωστή κατεύθυνση, δηλαδή το όνομα της μεταβλητής:

```
> 5 -> a
>
```

Μετά την ανάθεση, το a έχει πάρει την τιμή 5, και θα τη διατηρήσει μέχρι να την αλλάξουμε. Στη συνέχεια μπορούμε να χρησιμοποιούμε το a σε οποιαδήποτε πράξη σα να ήταν αριθμός:

```
> a-2
[1] 3
>
```

Για ονόματα μεταβλητών μπορούμε να χρησιμοποιήσουμε ό,τι μας βολεύει. Για παράδειγμα, για να υπολογίσουμε ένα δείκτη σωματικής μάζας, μπορούμε να γράψουμε:

```
> ypsos <- 1.81
> varos <- 85
> BMI <- varos / ypsos^2
> BMI
[1] 25.94548
>
```

Στο παράδειγμα αυτό χρησιμοποιήσαμε τη μεταβλητή με όνομα `ypsos` για το ύψος (σε μέτρα), τη μεταβλητή με όνομα `varos` για το βάρος (σε κιλά), και τη μεταβλητή με όνομα `BMI` για το δείκτη. Φυσικά θα μπορούσαμε να είχαμε χρησιμοποιήσει οποιαδήποτε άλλα ονόματα:

```
> a <- 1.81
> b <- 85
> i <- b / a^2
> i
[1] 25.94548
>
```

Προσοχή, η υποδιαστολή των δεκαδικών αριθμών είναι τελεία, όχι κόμμα!

Γ. Συναρτήσεις

Λέγοντας συναρτήσεις εννοούμε, πολύ χοντρικά, προκαθορισμένες αντιστοιχίες ή λειτουργίες που εφαρμόζονται σε έναν ή περισσότερους αριθμούς και δίνουν συγκεκριμένο αποτέλεσμα. Για παράδειγμα, η τετραγωνική ρίζα ενός αριθμού δίνεται από τη συνάρτηση `sqrt`:

```
> sqrt(16)
[1] 4
>
```

Η συνάρτηση round χρησιμοποιείται για τη στρογγυλοποίηση των δεκαδικών ψηφίων. Έτσι, μπορούμε να κρατήσουμε μόνο ένα δεκαδικό από το δείκτη σωματικής μάζας που υπολογίσαμε παραπάνω:

```
> round(BMI,1)
[1] 25.9
>
```

Αν θέλαμε δύο δεκαδικά ψηφία, θα έπρεπε να δώσουμε `round(BMI,2)`. Αν παραλείψουμε εντελώς την παράμετρο αυτή (το δεύτερο αριθμό), εννοείται το μηδέν (κανένα δεκαδικό).

Όλες οι συναρτήσεις στο R δέχονται τα ορίσματά τους μέσα σε παρενθέσεις. Πρέπει να είμαστε πάντα πολύ προσεκτικοί στις παρενθέσεις, να μην τις ξεχνάμε, διότι αυτό οδηγεί σε λάθη, π.χ.

```
> round BMI,2
Error: unexpected symbol in "round BMI"
>
```

Ένα σημείο που πρέπει να προσέχουμε ιδιαίτερα είναι ότι όσες παρενθέσεις ανοίγουμε πρέπει αντίστοιχα να τις κλείνουμε. Αν λοιπόν έχουμε συναρτήσεις μέσα σε συναρτήσεις, ή πράξεις με παρενθέσεις μέσα σε συναρτήσεις, θα πρέπει να ελέγχουμε ότι για κάθε παρένθεση που ανοίγει (αριστερή) υπάρχει και η αντίστοιχη που κλείνει (δεξιά).

```
> round(85/(1.81^2))
[1] 26
>
```

Αν ξεχάσουμε να κλείσουμε κάποια παρένθεση, το R θεωρεί ότι δεν έχουμε τελειώσει με την εντολή μας και μας παρουσιάζει ένα + (σε κόκκινο χρώμα) αναμένοντας τη συνέχεια.

```
> round(85/(1.81^2)
+
```

Στην περίπτωση αυτή, δίνουμε την τελική παρένθεση συνεχίοντας στην επόμενη σειρά:

```
+ )
[1] 26
>
```

Δ. Ακολουθίες

Πολλές φορές χρειάζεται να επεξεργαστούμε ένα σύνολο από αριθμούς, ως μια ομάδα ή μια ακολουθία. Για παράδειγμα, μπορεί να θέλουμε να υπολογίσουμε το άθροισμα περισσότερων από δύο αριθμών. Θα μπορούσαμε να τους προσθέσουμε στη σειρά, με το σύμβολο της πρόσθεσης:

```
> 5 + 2 + 6 + 8 + 3
[1] 24
>
```

Έτσι όμως χάνουμε τη δυνατότητα να διατηρήσουμε αυτή την ομάδα αριθμών για άλλες πράξεις, και πρέπει να τους γράφουμε από την αρχή κάθε φορά. Για την περίπτωση ενός μοναδικού αριθμού, είδαμε παραπάνω ότι μπορούμε να χρησιμοποιήσουμε μια μεταβλητή που να διατηρεί την τιμή του. Αντίστοιχα, για την περίπτωση ομάδων αριθμών, χρησιμοποιούμε τη συνδυαστική συνάρτηση `c`, η οποία ενώνει μια ομάδα αριθμών σε μια σταθερή ακολουθία. Η ομάδα αυτή δίνεται μέσα σε ζεύγος παρενθέσεων, όπου οι μεμονωμένοι αριθμοί χωρίζονται με κόμμα.

```
> omada <- c(5,2,6,8,3)
>
```

Με τον τρόπο αυτό, η μεταβλητή `omada` περιέχει πλέον αυτήν την πεντάδα αριθμών.

```
> omada
[1] 5 2 6 8 3
>
```

Έτσι, αν θέλουμε το άθροισμα αυτής της ομάδας, μπορούμε να χρησιμοποιήσουμε απευθείας τη συνάρτηση `sum` :

```
> sum(omada)
[1] 24
>
```

Το R μας δίνει συναρτήσεις για διάφορους χρήσιμους υπολογισμούς ομάδων αριθμών, όπως για παράδειγμα τον εντοπισμό του μέγιστου και του ελάχιστου.

```
> min(omada)
[1] 2
> max(omada)
[1] 8
>
```

Το πλήθος των στοιχείων μιας ακολουθίας δίνεται από τη συνάρτηση `length`

```
> length(omada)
[1] 5
>
```

Αργότερα θα δούμε περισσότερες συναρτήσεις, που θα τις χρειαστούμε για τις αναλύσεις μας.

Προς το παρόν, εξασκηθείτε στη βασική χρήση του R χρησιμοποιώντας τις παραπάνω πράξεις και συναρτήσεις, και κατασκευάζοντας παρόμοια δικά σας παραδείγματα. Πειραματιστείτε με διαφορετικές τιμές και παραστάσεις και μη φοβάστε όταν δίνετε κάτι λάθος!

Αθανάσιος Πρωτόπαπας
Μάρτιος 2012

Οδηγίες χρήσης του R, μέρος 2^{ο}

Ελληνικά

Αν προσπαθήσουμε να γράψουμε ελληνικά ή να ανοίξουμε κάποιο αρχείο δεδομένων με ελληνικούς χαρακτήρες στο R, μπορεί αντί για ελληνικά να δούμε λατινικούς χαρακτήρες με τόνους ή άλλα καλλικαντζαράκια. Τότε δίνουμε την παρακάτω εντολή για να γυρίσει το R στα ελληνικά:

```
> Sys.setlocale("LC_CTYPE","Greek")
[1] "Greek_Greece.1253"
```

Η απόκριση του R (με μπλε) επιβεβαιώνει τη ρύθμιση των ελληνικών.

Πλαίσια δεδομένων

Το R διατηρεί μετρήσεις μέσα σε δομές που ονομάζονται «πλαίσια δεδομένων» (data frame). Κάθε πλαίσιο δεδομένων περιέχει μία ή περισσότερες μεταβλητές.

Για παράδειγμα, ας καταγράψουμε το φύλο και την ηλικία δύο ατόμων, του Γιάννη και της Μαρίας, σε ένα πλαίσιο δεδομένων το οποίο αναθέτουμε σε μια μεταβλητή με όνομα `atoma`:

```
> atoma<-data.frame(sex=c("M","F"),age=c(21,22))
```

Με τη συνάρτηση `data.frame` ορίζουμε ένα πλαίσιο δεδομένων. Στη συνάρτηση δίνουμε ως ορίσματα τις μεταβλητές που θέλουμε να περιέχει το πλαίσιο, δηλαδή `sex` (φύλο) και `age` (ηλικία). Σε κάθε μεταβλητή δίνουμε τις αντίστοιχες μετρήσεις, ως ακολουθία τιμών (χρησιμοποιώντας τη συνάρτηση `c` που είδαμε στο πρώτο μέρος των οδηγιών).
Αν θέλουμε (δεν είναι υποχρεωτικό) μπορούμε να προσθέσουμε ετικέτες στις σειρές του πλαισίου για να αναγνωρίζουμε ονομαστικά τα δεδομένα:

```
> rownames(atoma)<-c("Γιάννης","Μαρία")
```

Τώρα μπορούμε να δούμε τα περιεχόμενα του πλαισίου δεδομένων atoma:

```
> atoma
        sex age
Γιάννης   M  21
Μαρία     F  22
```

Το R μας δίνει, σε μορφή πίνακα, όλα τα δεδομένα του πλαισίου. Για να εξετάσουμε τη δομή του πλαισίου μπορούμε να χρησιμοποιήσουμε τη συνάρτηση `str`, η οποία μας δίνει περιληπτικά το είδος των δεδομένων και ενδεικτικές τιμές για κάθε στήλη (μεταβλητή):

```
> str(atoma)
'data.frame':   2 obs. of  2 variables:
 $ sex: Factor w/ 2 levels "F","M": 2 1
 $ age: num  21 22
```

Στην πρώτη σειρά της απόκρισης, η συνάρτηση `str` μας ενημερώνει ότι το `atoma` είναι πλαίσιο δεδομένων (`'data frame'`), το οποίο περιέχει δύο «παρατηρήσεις» (`obs.`=observations) και δύο «μεταβλητές». Λέγοντας «παρατηρήσεις» αναφερόμαστε στις σειρές του πλαισίου, ενώ λέγοντας «μεταβλητές» αναφερόμαστε στις στήλες.
Στις επόμενες δύο σειρές της απόκρισης δίνονται οι πληροφορίες που αφορούν σε καθεμιά μεταβλητή ξεχωριστά:

Η πρώτη μεταβλητή ονομάζεται `sex` και είναι τύπου `Factor`, δηλαδή «παράγοντας». Αυτό, στην ορολογία του R, σημαίνει ότι πρόκειται για *κατηγορική* μεταβλητή. Περιλαμβάνει 2 «επίπεδα» (`levels`), δηλαδή δύο κατηγορίες, οι οποίες ονομάζονται `"F"` και `"M"` (γυναίκες και άντρες). Το `"F"` αναφέρεται πρώτο διότι το R χρησιμοποιεί από μόνο του αλφαβητική σειρά για την αναφορά σε κατηγορίες. Η σειρά περιγραφής της μεταβλητής `sex` ολοκληρώνεται με τα πρώτα στοιχεία της στήλης, δηλαδή τον αριθμό 2 (αναφέρεται στη δεύτερη κατηγορία, `"M"`) και τον αριθμό 1 (πρώτη κατηγορία, `"F"`). Αυτό μας λέει ότι η πρώτη σειρά δεδομένων είναι τύπου `"F"` και η δεύτερη τύπου `"M"`.

Η δεύτερη μεταβλητή ονομάζεται `age` και είναι τύπου `num`, δηλαδή «αριθμητική» (`numeric`). Αυτό, στην ορολογία του R σημαίνει ότι πρόκειται για *ποσοτική* μεταβλητή. Δεν χρειάζονται άλλες διευκρινίσεις, καθώς στις ποσοτικές μεταβλητές τα νούμερα είναι αυτονόητα. Η σειρά ολοκληρώνεται με τα πρώτα στοιχεία της στήλης, δηλαδή τους αριθμούς 21 και 22, οι οποίοι αντιστοιχούν στην πρώτη και τη δεύτερη σειρά δεδομένων, αντίστοιχα.

> Να θυμάστε ότι τα κατηγορικά δεδομένα στο R είναι τύπου **factor** ενώ τα ποσοτικά δεδομένα είναι τύπου **numeric**.

Όπως βλέπουμε στην απόκριση της συνάρτησης `str`, πριν από κάθε μεταβλητή εμφανίζεται ένα δολλάριο (`$`). Το σύμβολο του δολλαρίου στο R χρησιμοποιείται για να δηλώνουμε συγκεκριμένες μεταβλητές μέσα σε πλαίσια δεδομένων. Έτσι, για να αναφερθούμε στις ηλικίες (μεταβλητή `age`) που βρίσκονται μέσα στο πλαίσιο `atoma` γράφουμε

```
> atoma$age
[1] 21 22
```

ενώ για να αναφερθούμε στο φύλο (μεταβλητή `sex`) γράφουμε

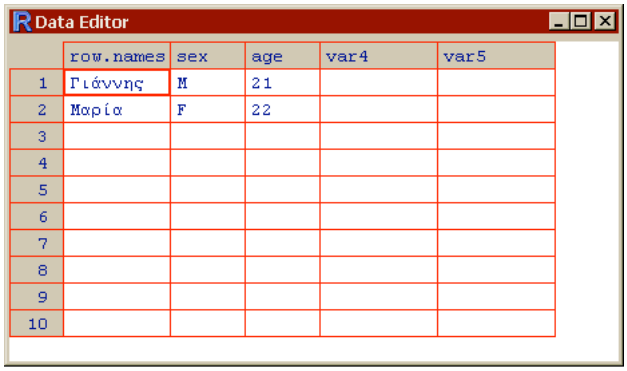
```
> atoma$sex
[1] M F
Levels: F M
```

Στην περίπτωση της κατηγορικής μεταβλητής το R μας ενημερώνει και για το σύνολο των κατηγοριών που περιλαμβάνει η συγκεκριμένη μεταβλητή.
Απομονώνοντας τις μεταβλητές με αυτόν τον τρόπο, μπορούμε να τις χειριστούμε ως κοινές ακολουθίες. Για παράδειγμα, μπορούμε να χρησιμοποιήσουμε τις συναρτήσεις από το πρώτο μέρος των οδηγιών για να υπολογίσουμε το άθροισμα, το πλήθος κλπ.

```
> sum(atoma$age)
[1] 43
> length(atoma$sex)
[1] 2
```

Για την επεξεργασία δεδομένων σε πλαίσια, το R μας δίνει τη συνάρτηση `fix`, με την οποία μας εμφανίζει ένα ειδικό παράθυρο στο οποίο μπορούμε να τροποποιήσουμε ή να προσθέσουμε στοιχεία σε ένα πλαίσιο δεδομένων.

```
> fix(atoma)
```



Όταν τελειώσουμε την προσθήκη ή επεξεργασία των στοιχείων, κλείνουμε το ειδικό παράθυρο με κλικ στο ✕ (πάνω δεξιά γωνία) και η μεταβλητή `atoma` ενημερώνεται αυτόματα.

Γραφική παρουσίαση δεδομένων

Το R διαθέτει πολλές συναρτήσεις για την παρουσίαση και λεπτομερειακή εξέταση και ανάλυση των δεδομένων μας. Ας υποθέσουμε ότι έχουμε πέντε μετρήσεις ύψους ενός ατόμου:

```
> alexh <- c( 1.85, 1.85, 1.81, 1.82, 1.83 )
```

Η μεταβλητή `alexh` περιέχει μια ακολουθία πέντε αριθμών. Με τη συνάρτηση `table` («πίνακας») μπορούμε να μετρήσουμε πόσες φορές εμφανίζεται κάθε τιμή:

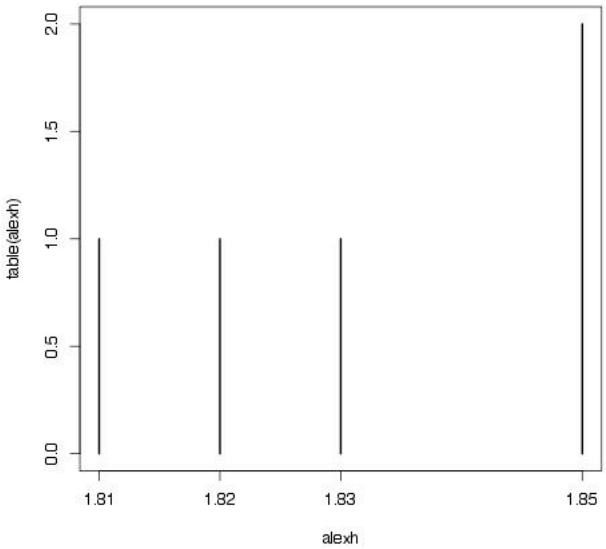
```
> table(alexh)
alexh
1.81 1.82 1.83 1.85
   1    1    1    2
```

Αυτός είναι ένας απλός πίνακας συχνοτήτων. Βλέπουμε ότι η τιμή 1.85 εμφανίζεται δύο φορές ενώ οι άλλες τιμές από μία φορά. Η τιμή που εμφανίζεται τις περισσότερες φορές ονομάζεται «δεσπόζουσα» (mode). Αν η καλύτερη τιμή έβγαινει με ψηφοφορία, η δεσπόζουσα είναι εκείνη που θα κέρδιζε λόγω πλειοψηφίας.

Την πληροφορία αυτή μπορούμε να τη δούμε και γραφικά, με τη συνάρτηση plot:

```
> plot(table(alexh))
```

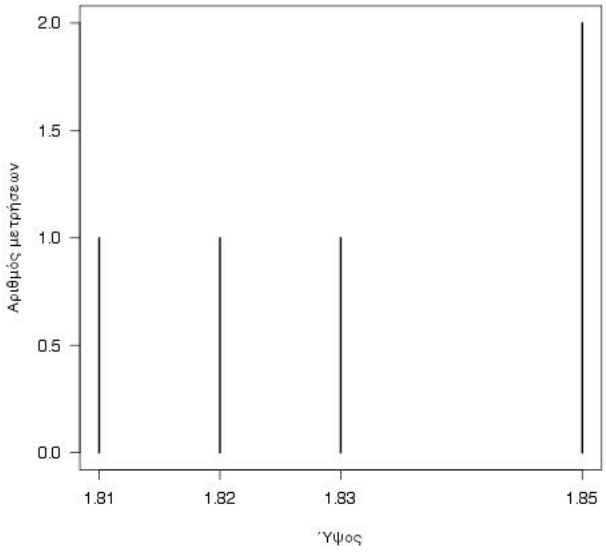
Το R ανοίγει ένα νέο παράθυρο για τη γραφική απεικόνιση, στο οποίο εμφανίζει το εξής :



Εδώ βλέπουμε ένα ραβδόγραμμα με το ύψος στον οριζόντιο άξονα και το πλήθος των αντίστοιχων μετρήσεων στον κατακόρυφο. Κάθε μέτρηση εμφανίζεται σα μια γραμμούλα που φτάνει σε ύψος 1.0, ενώ στο ύψος 1.85, που υπάρχουν δύο μετρήσεις, εμφανίζονται δύο γραμμούλες η μία πάνω στην άλλη, κάνοντας μαζί μια μακρύτερη που φτάνει στο ύψος 2.0. Αυτό είναι ένα διάγραμμα συχνοτήτων, που μας λέει πόσο συχνά εμφανίζεται κάθε αριθμός. Μπορούμε να καλλωπίσουμε κάπως τη γραφική απεικόνιση, προσθέτοντας ετικέτες:

```
> plot(table(alexh),las=1,xlab="Ύψος",ylab="Αριθμός μετρήσεων")
```

Η παράμετρος las στρίβει την αρίθμηση στον κατακόρυφο άξονα ώστε να διαβάζεται όρθια, ενώ οι δύο παράμετροι lab (από το label=ετικέτα) καθορίζουν τις ετικέτες στον οριζόντιο άξονα (με το x) και στον κατακόρυφο άξονα (με το y).



Για να δούμε γραφικά την κατανομή των μετρήσεων στην κλίμακα το R μας δίνει τη συνάρτηση `hist` (histogram=ιστόγραμμα). Την κατανομή αυτή, με περισσότερη αριθμητική λεπτομέρεια αλλά χωρίς γραφικά, μπορούμε να δούμε με τη συνάρτηση `stem` που παράγει διάγραμμα μίσχου-φύλλων.

Για το σχετικό πλήθος των επιμέρους κατηγοριών σε κατηγορικά δεδομένα, έχουμε τη συνάρτηση `table`, που είδαμε παραπάνω ότι μας δίνει τον πίνακα κατανομής, καθώς και τη συνάρτηση `pie`, που μας δίνει γραφικά την ίδια πληροφορία με κυκλικό διάγραμμα (pie chart).

Δοκιμάστε τις συναρτήσεις αυτές στα δικά σας δεδομένα!

Περίληψη δεδομένων

Μια πολύ χρήσιμη συνάρτηση για γρήγορη επισκόπηση των δεδομένων μας είναι η περίληψη:

```
> summary(alexh)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.810   1.820   1.830   1.832   1.850   1.850
```

Τα αποτελέσματα της περίληψης περιλαμβάνουν την ελάχιστη (`Min.`) και μέγιστη (`Max.`) τιμή, το μέσο όρο (`Mean`), καθώς και τρεις ακόμα δείκτες. Ο πιο σημαντικός είναι η διάμεσος (`Median`), δηλαδή η τιμή που είναι μεγαλύτερη από τις μισές μετρήσεις και μικρότερη από τις άλλες μισές. Για να το καταλάβουμε καλύτερα, ας δούμε τις τιμές μας σε αύξουσα σειρά:

```
> sort(alexh)
[1] 1.81 1.82 1.83 1.85 1.85
```

Η μικρότερη τιμή είναι 1.81 (πρώτη) και η μεγαλύτερη 1.85 (τελευταία). Αφαιρώντας δύο τιμές από κάθε άκρη μένει η μεσαία μέτρηση, που είναι 1.83. Αυτή είναι η διάμεσος.

Η ελάχιστη, μέγιστη, μέση, και διάμεσος τιμή υπολογίζονται στο R απευθείας με τις συναρτήσεις `min, max, mean` και `median`, αντίστοιχα.

Αν κόψουμε το κάθε μισό στη μέση μπορούμε να βρούμε τη διάμεσο του κάθε μισού, που χωρίζουν το πρώτο τέταρτο και το τελευταίο τέταρτο των δεδομένων. Τα σημεία αυτά ονομάζονται *τεταρτημόρια*: Το πρώτο τεταρτημόριο (1st quartile) χωρίζει το χαμηλότερο 25%. Το δεύτερο τεταρτημόριο είναι η διάμεσος και χωρίζει το 50%. Το τρίτο τεταρτημόριο (3rd quartile) χωρίζει το υψηλότερο 25%. Αυτές είναι οι επιπλέον τιμές που μας δίνει η περίληψη του R. Βέβαια για τόσο λίγες τιμές που έχουμε εδώ αυτό δεν έχει πολύ νόημα, είναι όμως πάρα πολύ χρήσιμο σε μεταβλητές με δεκάδες ή εκατοντάδες μετρήσεις.

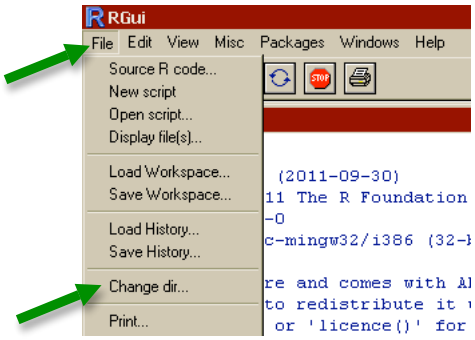
Η περίληψη εφαρμόζεται και σε ολόκληρα πλαίσια δεδομένων. Στην περίπτωση αυτή μας δίνει πληροφορίες για όλες τις μεταβλητές που περιλαμβάνονται στο πλαίσιο δεδομένων και προσαρμόζεται αυτόματα σε κάθε μεταβλητή αν είναι κατηγορική ή ποσοτική. Παράδειγμα:

```
> summary(atoma)
 sex        age
 F:1   Min.   :21.00
 M:1   1st Qu.:21.25
       Median :21.50
       Mean   :21.50
       3rd Qu.:21.75
       Max.   :22.00
```

Χρήση εξωτερικών αρχείων

Το R μπορεί να διαβάσει δεδομένα που έχουμε αποθηκευμένα σε αρχεία στο δίσκο του υπολογιστή μας. Με το R μπορούμε επίσης να αποθηκεύσουμε δεδομένα, αποτελέσματα επεξεργασίας, ή και τις εντολές και συναρτήσεις που χρησιμοποιήσαμε για την ανάλυσή μας.

Για να μπορεί να χρησιμοποιηθεί κάποιο εξωτερικό αρχείο πρέπει προηγουμένως να υποδείξουμε στο R σε ποιο φάκελο βρίσκονται τα αρχεία μας. Η επιλογή φακέλου γίνεται μέσα από τον κατάλογο επιλογών File → Change dir… (dir=directory, δηλαδή κατάλογος αρχείων). Με την επιλογή αυτή το R μας εμφανίζει το γνωστό παράθυρο επιλογής φακέλου των windows. Εντοπίζουμε και επιλέγουμε την τοποθεσία όπου βρίσκονται τα αρχεία μας.



Αφού επιλέξουμε τη σωστή τοποθεσία, μπορούμε να φορτώσουμε ένα πλαίσιο δεδομένων απευθείας από το δίσκο με τη συνάρτηση `read.table`, αναθέτοντας το περιεχόμενο απευθείας σε μια μεταβλητή. Π.χ., για να χρησιμοποιήσουμε τα κατηγορικά δεδομένα του 3ου κεφαλαίου του βιβλίου, τα αναθέτουμε στη μεταβλητή `ch3` ως εξής:

```
> read.table("chapter3_1.Rdata") -> ch31
```

Προσοχή, να μην ξεχνάμε την τελίτσα μέσα στο όνομα της συνάρτησης, χωρίς κενά!
Η συνάρτηση `str` μας δείχνει το αποτέλεσμα της ανάθεσης:

```
> str(ch31)
'data.frame':    264 obs. of  1 variable:
 $ education: Factor w/ 5 levels "Άλλο","Λύκειο",..: 5 5 5 5 5 5
5 5 5 5 ...
```

Πρόκειται για ένα πλαίσιο δεδομένων με μια μοναδική κατηγορική μεταβλητή με όνομα `education` η οποία περιέχει δεδομένα πέντε κατηγοριών. Τα στοιχεία των πρώτων σειρών ανήκουν όλα στην 5η κατηγορία.

Αργότερα θα δούμε πώς μπορούμε να αποθηκεύσουμε δικά μας δεδομένα καθώς και να χρησιμοποιήσουμε αρχεία αναλύσεων και εξωτερικά πακέτα συναρτήσεων.

Αθανάσιος Πρωτόπαπας
Μάρτιος 2012

Οδηγίες χρήσης του R, μέρος 3^{ο}

Βιβλιοθήκες

Το R διαθέτει πολλές χρήσιμες συναρτήσεις χωρίς ειδική αναζήτηση, αυτόματα. Υπάρχουν όμως πάρα πολλές διαφορετικές συναρτήσεις, που χρησιμοποιούνται για διαφορετικά είδη αναλύσεων. Οι περισσότερες είναι οργανωμένες σε εξωτερικά «πακέτα» συναρτήσεων, που ονομάζονται «βιβλιοθήκες». Τα πακέτα αυτά πρέπει να τα εγκαταστήσουμε στον υπολογιστή μας ξεχωριστά από το ίδιο το R. Η εγκατάσταση κατεβάζει αυτομάτως το πακέτο από το διαδίκτυο, άρα πρέπει πρώτα να εξασφαλίσουμε ότι η σύνδεσή μας είναι ενεργή. Για να εγκαταστήσουμε ένα πακέτο, π.χ. το `psych` (για αναλύσεις που χρησιμοποιούνται συχνά στην ψυχολογία), χρησιμοποιούμε την κατάλληλη συνάρτηση μέσα από το R:

```
> install.packages("psych", depend=T)
```

Την πρώτη φορά που θα χρησιμοποιήσουμε αυτή τη συνάρτηση, το R θα μας ρωτήσει από πού να κατεβάσει το πακέτο. Επιλέγουμε την Ελλάδα ή άλλη ευρωπαϊκή χώρα, π.χ. Αυστρία. Στη συνέχεια το R κατεβάζει και εγκαθιστά ό,τι είναι απαραίτητο για τη λειτουργία του πακέτου.

Αν θέλουμε να εγκαταστήσουμε περισσότερα πακέτα μονομιάς, χρησιμοποιούμε ακολουθία:

```
> install.packages(c("e1071","nortest","Hmisc"), depend=T)
```

Το R εγκαθιστά τα πακέτα μαζί με τυχόν προαπαιτούμενα. Στο εξής θα είναι διαθέσιμα στον υπολογιστή μας ανεξάρτητα από το αν είμαστε συνδεδεμένοι στο διαδίκτυο ή όχι.

Για να χρησιμοποιήσουμε συναρτήσεις από τα νέα πακέτα, πρέπει πρώτα να «φορτώσουμε» τη βιβλιοθήκη που περιλαμβάνει το πακέτο, χρησιμοποιώντας τη συνάρτηση `library`. Έτσι, αν έχουμε ένα πλαίσιο δεδομένων `ch` και θέλουμε να χρησιμοποιήσουμε τη συνάρτηση `describe` που περιλαμβάνεται στη βιβλιοθήκη του πακέτου `psych`:

```
> library(psych)
> describe(ch)
```

Η ενεργοποίηση της βιβλιοθήκης πρέπει να γίνεται κάθε φορά που ξεκινάμε το R.

Εναλλακτικά, αν θέλουμε να χρησιμοποιήσουμε στα γρήγορα κάποια συνάρτηση από ένα πακέτο, χωρίς προηγουμένως να φορτώσουμε ολόκληρη τη βιβλιοθήκη, μπορούμε να δώσουμε το όνομα της βιβλιοθήκης μαζί με τη συνάρτηση, με δυο άνω-κάτω τελείες ανάμεσα:

```
> psych::describe(ch)
```

Η μέθοδος αυτή δεν συνιστάται διότι πρέπει κάθε φορά να ξαναγράφουμε ολόκληρο το όνομα της βιβλιοθήκης. Είναι απλούστερο να τη φορτώσουμε μια φορά με τη συνάρτηση `library`. Εννοείται ότι σε κάθε περίπτωση πρέπει προηγουμένως να έχουμε εγκαταστήσει το αντίστοιχο πακέτο, με τη συνάρτηση `install.packages` (προσοχή στην τελίτσα!).

Δείκτες πινάκων

Στο 2^{ο} μέρος των οδηγιών είδαμε ότι το R χρησιμοποιεί δομές που ονομάζονται «πλαίσια δεδομένων» και περιέχουν μία ή περισσότερες μεταβλητές. Οι δομές αυτές έχουν το χαρακτηριστικό ότι αποτελούνται από σειρές (εγγραφές) και στήλες (μεταβλητές). Π.χ., οι πρώτες σειρές του πίνακα ύψους των φοιτητών είναι οι εξής:

```
> read.table("classheight.Rdata")->ch
> ch
   sex    h
1    f 1.57
2    f 1.65
3    f 1.65
4    f 1.72
5    m 1.83
...
```

Το R μας επιτρέπει να χειριστούμε σειρές, στήλες, και μεμονωμένα στοιχεία, χρησιμοποιώντας ένα σύστημα δεικτών μέσα σε αγκύλες (τις τετράγωνες παρενθέσεις: `[]`). Συγκεκριμένα, κάθε πίνακας δύο διαστάσεων (όπως είναι το πλαίσιο δεδομένων) ορίζεται ως [σειρές,στήλες]. Μέσα στις αγκύλες γράφουμε τη σειρά (ή σειρές) που θέλουμε, κόμμα, και τη στήλη (ή στήλες) που θέλουμε. Η 1^{η} σειρά του πίνακα γράφεται [1,] ενώ η 2^{η} στήλη του πίνακα γράφεται [,2]:

```
> ch[1,]
   sex    h
1    f 1.57
> ch[,2]
[1] 1.57 1.65 1.65 1.72 1.83 1.60 1.62 1.85 1.87 1.83 1.79 1.57
...
```

Αν θέλουμε το στοιχείο της 4^{ης} σειράς, 2^{ης} στήλης, το εντοπίζουμε ως εξής:

```
> ch[4,2]
[1] 1.72
```

Ένα πολύ χρήσιμο στοιχείο στο R είναι ότι μπορούμε να επιλέξουμε στοιχεία με βάση κάποια συνθήκη. Π.χ., μπορούμε να ζητήσουμε όλες τις στήλες από τις σειρές του πλαισίου `ch` για τις οποίες η μεταβλητή `h` είναι μεγαλύτερη του 1.85, ως εξής:

```
> ch[ch$h>1.85,]
   sex    h
9    m 1.87
27   m 1.87
29   m 1.89
```

Προσέξτε ότι η συνθήκη `ch$h>1.85` τοποθετήθηκε πριν από το κόμμα, άρα κάνει επιλογή σειρών. Μετά το κόμμα δεν τοποθετήθηκε τίποτα, άρα δεν επιλέγονται στήλες, και εννοείται ότι τις ζητάμε όλες. Με τον ίδιο τρόπο θα μπορούσαμε να ζητήσουμε τις σειρές στις οποίες το φύλο είναι f και το ύψος είναι μεγαλύτερο του 1,70:

```
> ch[ch$sex=="f" & ch$h>1.70,]
   sex    h
4    f 1.72
11   f 1.79
25   f 1.75
```

Προσέξτε ότι ο έλεγχος ισότητας γίνεται με το διπλό σύμβολο == (ίσον), ενώ ο έλεγχος ανισότητας με το μονό σύμβολο > (μεγαλύτερο). Η σύζευξη των δύο ελέγχων γίνεται με το σύμβολο & (λογικό «και»). Αν θέλαμε διάζευξη (δηλαδή ή το ένα ή το άλλο ή και τα δύο) θα χρησιμοποιούσαμε το σύμβολο | (κατακόρυφη κάθετος, λογικό «ή»), π.χ.:

```
> ch[ch$h>1.85 | ch$h<1.55,]
   sex    h
9    m 1.87
16   f 1.51
27   m 1.87
29   m 1.89
```

Μπορούμε, φυσικά να επιλέξουμε μόνο μία στήλη από το αποτέλεσμα της συνθήκης:

```
> ch[ch$h>1.85 | ch$h<1.55,1]
[1] m f m m
Levels: f m
> ch[ch$h>1.85 | ch$h<1.55,2]
[1] 1.87 1.51 1.87 1.89
```

Στην περίπτωση αυτή λαμβάνουμε ως αποτέλεσμα μια απλή ακολουθία. Το τελευταίο θα μπορούσαμε να το πετύχουμε και εφαρμόζοντας τους δείκτες πάνω στην ίδια τη μεταβλητή:

```
> ch$h[ch$h>1.85 | ch$h<1.55]
[1] 1.87 1.51 1.87 1.89
```

Εδώ, φυσικά δεν χρησιμοποιούμε το κόμμα πριν κλείσουμε την αγκύλη, αφού μια μεταβλητή είναι μονοδιάστατη και δεν περιέχει στήλες.

➢ Η επιλογή στοιχείων με το σύστημα δεικτών είναι πάρα πολύ χρήσιμη και θα τη συναντήσετε σε πολλές περιπτώσεις δουλεύοντας στο R και μελετώντας τα διαθέσιμα παραδείγματα.

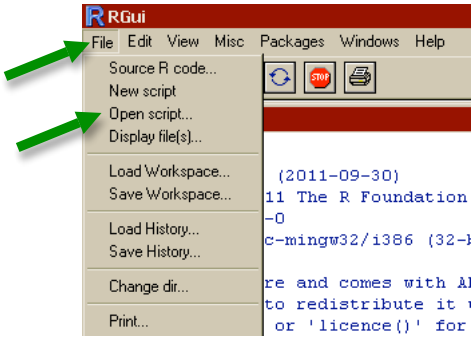
➢ Κι ένα κολπάκι: Για να μην πληκτρολογούμε ξανά τα ίδια, όταν θέλουμε να επαναλάβουμε μια προηγούμενη συνάρτηση στο χώρο αλληλεπίδρασης του R πατάμε απλώς το «βελάκι πάνω» στο πληκτρολόγιο, όσες φορές χρειάζεται, ώστε να επανέλθει στην ενεργή γραμμή.

<u>Αρχεία εντολών</u>

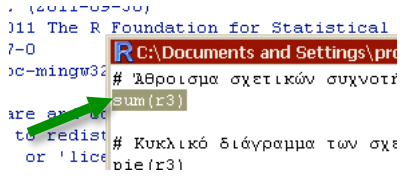
Αν έπρεπε κάθε φορά να πληκτρολογούμε εκ νέου όλες τις συναρτήσεις που χρειαζόμαστε, ή χρήση του R θα ήταν πολύ κουραστική και καθόλου αποδοτική. Στην πραγματικότητα αυτό που κάνουμε είναι να αποθηκεύουμε τις εντολές μας σε ένα αρχείο, ώστε να μπορούμε να τις τροποποιήσουμε ή να τις εκτελέσουμε αυτούσιες οποιαδήποτε άλλη στιγμή. Το αρχείο αυτό περιλαμβάνει ακριβώς την αλληλουχία των συναρτήσεων που χρησιμοποιούμε. Προαιρετικά,

προσθέτουμε σχόλια και επεξηγήσεις ώστε να μπορέσουμε να θυμηθούμε τι ακριβώς κάναμε με τις συναρτήσεις αυτές και για ποιο λόγο. Ουσιαστικά, δηλαδή, γράφουμε ένα «πρόγραμμα» στη «γλώσσα» του R για να εκτελεί τις αναλύσεις μας. Στην ορολογία του R ένα τέτοιο πρόγραμμα ονομάζεται script.

Ένα αρχείο εντολών που προϋπάρχει (π.χ. το αποθηκεύσαμε στο παρελθόν ή μας το έστειλε κάποιος άλλος) μπορούμε να το φορτώσουμε στο R από τον κύριο κατάλογο επιλογών (το «μενού») στην κορυφή της οθόνης. Η επιλογή File → Open script… μας ανοίγει το γνωστό παραθυράκι επιλογής αρχείων για να φορτώσουμε το πρόγραμμά μας.



Ένα πρόγραμμα ανοίγει στο δικό του παραθυράκι, μέσα στον ευρύτερο χώρο του R, ώστε να βλέπουμε καθαρά τη σειρά των συναρτήσεων που περιέχει. Είναι ανεξάρτητο από το χώρο όπου πληκτρολογούμε τις συναρτήσεις μας και βλέπουμε τα αποτελέσματα.
Για να χρησιμοποιήσουμε μια συνάρτηση από το πρόγραμμα που ανοίξαμε, πρώτα την «επιλέγουμε» με το ποντίκι ή με το πληκτρολόγιο (Ctrl-Shift-βελάκι δεξιά/αριστερά).
Εδώ έχουμε επιλέξει τη συνάρτηση υπολογισμού του αθροίσματος της ακολουθίας `r3` :

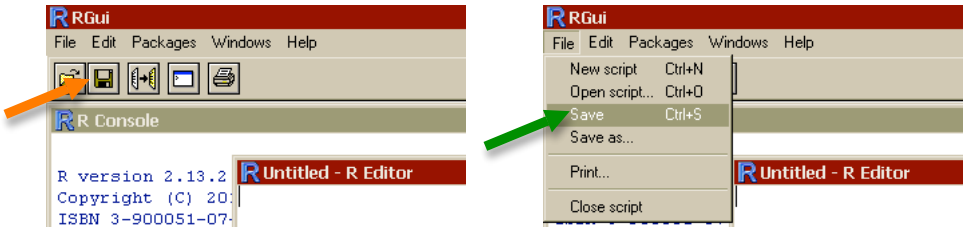


Στη συνέχεια κρατάμε πατημένο το πλήκτρο Control και πατάμε το πλήκτρο R. Ο συνδυασμός αυτός ονομάζεται Ctrl-R και στο R σημαίνει «Run» (τρέξε), εκτελεί δηλαδή ό,τι είναι φωτισμένο, αντιγράφοντάς το στο χώρο αλληλεπίδρασης με το R.

Στο παραπάνω παράδειγμα βλέπουμε ότι μια σειρά του αρχείου αρχίζει με το σύμβολο # (δίεση) και συνεχίζει «Άθροισμα σχετικών συχνοτήτων...». Αυτή η σειρά δεν είναι εντολή για το R, αλλά επεξηγηματικό σχόλιο για τον αναγνώστη/χρήστη του αρχείου. Το σύμβολο # υποδεικνύει στο R να αγνοήσει οτιδήποτε βρίσκεται πιο δεξιά από αυτό. Το χρησιμοποιούμε για να ξεκινήσουμε μια σειρά με επεξηγηματικά σχόλια ή για να προσθέσουμε μια επεξήγηση στο τέλος κάποιας συνάρτησης (στα δεξιά της). Να χρησιμοποιείτε όσο περισσότερα επεξηγηματικά σχόλια μπορείτε, θα τα βρείτε πολύ χρήσιμα όταν ξαναχρειαστείτε ένα παλαιότερο αρχείο εντολών στο R (ή σε οποιαδήποτε άλλη γλώσσα προγραμματισμού).

Αν θέλουμε να αποθηκεύσουμε μια σειρά συναρτήσεων για μελλοντική χρήση, ξεκινάμε ένα νέο πρόγραμμα, με την επιλογή File → New script από τον κύριο κατάλογο επιλογών. Αυτό

ανοίγει ένα κενό παράθυρο, στο οποίο δακτυλογραφούμε τις συναρτήσεις μας, ή τις αντιγράφουμε από το χώρο αλληλεπίδρασης με Copy-Paste (με το ποντίκι ή με Ctrl-C, Ctrl-V). Αποθηκεύουμε τη δουλειά μας, ενώ το παράθυρο του προγράμματος είναι ενεργό, με κλικ επάνω στη δισκετούλα, με Ctrl-S, ή επιλέγοντας File → Save από τον κύριο κατάλογο.



Στη συνέχεια το αρχείο αυτό θα είναι διαθέσιμο να το ανοίξουμε και να το ξαναχρησιμοποιήσουμε οποιαδήποτε στιγμή, αρκεί να επιλέξουμε πρώτα το σωστό φάκελο (File → Change dir…) και στη συνέχεια το ίδιο το αρχείο (File → Open script…).

Προσοχή! Δεν πρέπει να μπερδεύουμε την αποθήκευση και ανάγνωση *προγραμμάτων* (script) με την αποθήκευση και ανάγνωση *πινάκων δεδομένων* (data table). Τα προγράμματα αποθηκεύονται και φορτώνονται από τον κύριο κατάλογο (File →) και εμφανίζονται σε δικό τους παράθυρο μέσα στο R. Αντίθετα, τα δεδομένα αποθηκεύονται και φορτώνονται από συναρτήσεις και δεν εμφανίζονται πουθενά παρά μόνο αν τα ζητήσουμε, είτε δακτυλογραφώντας το όνομά τους είτε λ.χ. με τη συνάρτηση `fix`.

Έχουμε προηγουμένως συναντήσει τη συνάρτηση `read.table` που φορτώνει ένα πλαίσιο δεδομένων και το αναθέτει σε μια μεταβλητή. Αντίστοιχα, μπορούμε να αποθηκεύσουμε ένα πλαίσιο δεδομένων, το οποίο περιέχει στοιχεία που πληκτρολογήσαμε ή αποτελέσματα υπολογισμών, με τη συνάρτηση `write.table`. Για παράδειγμα, το πλαίσιο δεδομένων από το 2^{ο} μέρος των οδηγιών, μπορεί να δημιουργηθεί και να αποθηκευτεί ως εξής:

```
> atoma<-data.frame(sex=c("M","F"),age=c(21,22))
> rownames(atoma)<-c("Γιάννης","Μαρία")
> write.table(atoma,"atoma.Rdata")
```

Το πρώτο όρισμα της συνάρτησης `write.table` είναι το πλαίσιο δεδομένων που θέλουμε να αποθηκεύσουμε και το δεύτερο όρισμα είναι το όνομα του αρχείου με το οποίο θα αποθηκευτεί στο σκληρό δίσκο μας. Στο εξής, αυτό το πλαίσιο δεδομένων θα είναι διαθέσιμο οποτεδήποτε, επιλέγοντας τον κατάλληλο φάκελο εργασίας και χρησιμοποιώντας τη συνάρτηση `read.table("atoma.Rdata")`.

Συνοψίζοντας, τα αρχεία δεδομένων που διανέμονται (π.χ. classheight.Rdata) τα διαβάζουμε μέσα από το χώρο αλληλεπίδρασης, ή μέσα από προγράμματα, με τη συνάρτηση `read.table`. Ενώ τα αρχεία προγραμμάτων (τα συνοδευτικά των κεφαλαίων του βιβλίου, π.χ. chapter3.R) τα διαβάζουμε από τον κατάλογο επιλογών με File → Open script…

Αθανάσιος Πρωτόπαπας
Μάιος 2012