

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 ξαναγράψουμε 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.

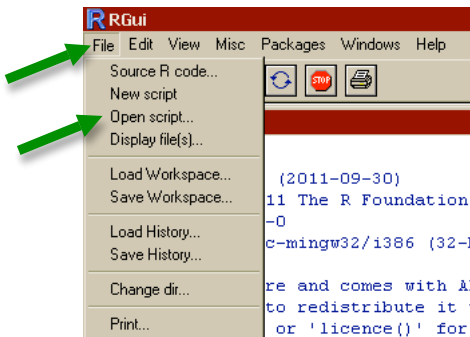
➤ Κι ένα κολπάκι: Για να μην πληκτρολογούμε ξανά τα ίδια, όταν θέλουμε να επαναλάβουμε μια προηγούμενη συνάρτηση στο χώρο αλληλεπίδρασης του R πατάμε απλώς το «βελάκι πάνω» στο πληκτρολόγιο, όσες φορές χρειάζεται, ώστε να επανέλθει στην ενεργή γραμμή.

Αρχεία εντολών

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` :

```
.(2011-09-30)
)11 The R Foundation for Statistical
7-0
c-mingw32 # Αθροισμα σχετικών συχνοτή
are an
to redist # Κυκλικό διάγραμμα των σχε
or 'lice
pie(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 «Αθροισμα 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 ξαναχρησιμοποιείτε 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...