Χειρισμός μονομεταβλητών δεδομένων

Βασικές Έννοιες

Στη στατιστική, τα δεδομένα χωρίζονται σε κατηγορικά, διακριτά αριθμητικά και συνεχή αριθμητικά. Τα συνεχή δεδομένα μπορούν να διακριτοποιηθούν αν από τις δυνατές τιμές τους κρατήσουμε τα ακέραια μέρη ή τις πλησιέστερες ακέραιες τιμές. Τα διακριτά δεδομένα μπορούν να θεωρηθούν με τη σειρά τους και ως μια μορφή κατηγορικών δεδομένων όπου οι κατηγορίες λαμβάνονται αν τα χωρίσουμε σε διαστήματα τιμών (bins).

Τα σύνολα δεδομένων που συγκεντρώνουν τις τιμές μιας μεταβλητής λέγονται *μονομεταβλητά* (univariate). Παρόμοια, υπάρχουν διμεταβλητά και πολυμεταβλητά σύνολα δεδομένων (bivariate, multivariate).

00 Εκκίνηση

Στη γραμμή εντολών γράψτε το γράμμα R (κεφαλαίο) και πατήστε ENTER για να εκκινήσει το πρόγραμμα. (στα Windows ή το MacOS κάντε κλικ στο κατάλληλο εικονίδιο του προγράμματος).

01 Πίνακες

Οι πίνακες χρησιμοποιούνται ευρέως για τη σύνοψη των δεδομένων. Η συνάρτηση του R για την πινακοποίηση ενός συνόλου δεδομένων είναι η table( ). Αν έχουμε ένα σύνολο τιμών, x, γράφοντας table(x) έχουμε ως αποτέλεσμα να εντοπιστούν όλες οι διαφορετικές τιμές στο σύνολο και να γραφούν από μία φορά μαζί με τη συχνότητα εμφάνισής τους.

Στη γραμμή εντολών γράψτε τα παρακάτω (το > είναι η προτροπή του R και προφανώς δεν το γράφετε):

```
> res=c("Y", "Y", "N", "Y", "N")
> table(res)
res
N Y
2 3
>
```

Τι αποτελέσματα πήρατε; Τι καταλαβαίνετε από τα παραπάνω;

02 Πίνακες – παράδειγμα από "βιβλιοθήκες".

Το R περιέχει και έναν αριθμό από παραδείγματα συνόλων δεδομένων που δίνονται ως ξεχωριστές "βιβλιοθήκες" και μπορούν να "φορτωθούν" με τη συνάρτηση library. Θα χρησιμοποιήσουμε τη βασική βιβλιοθήκη με το όνομα UsingR για τα επόμενα παραδείγματα.

Για να φορτωθεί η βιβλιοθήκη UsingR γράψτε:

```
> library(UsingR)
```

Δε χρειάζεται να το γράψουμε άλλη φορά όσο είμαστε μέσα στο R.

Αυτή η βιβλιοθήκη, περιέχει μεταξύ άλλων, ένα σύνολο μετεωρολογικών δεδομένων που αφορούν τη νέφωση που παρατηρήθηκε επί ένα μήνασε μια συγκεκριμένη περιοχή (central park). Φορτώνουμε αυτό το αρχείο γράφοντας απλά το όνομά του και θα δούμε να εκτυπώνονται οι τιμές του:

```
> central.park.cloud
 [1] partly.cloudy partly.cloudy partly.cloudy clear         partly.cloudy
 [6] partly.cloudy clear         cloudy        partly.cloudy clear
[11] cloudy        partly.cloudy cloudy        cloudy        clear
[16] partly.cloudy partly.cloudy clear         clear         clear
[21] clear         cloudy        cloudy        cloudy        cloudy
[26] cloudy        clear         partly.cloudy clear         clear
[31] partly.cloudy
Levels: clear partly.cloudy cloudy
```

Μάλιστα, στο τέλος γράφονται και τα "επίπεδα" νέφωσης, δηλαδή οι κατηγορίες των δεδομένων μας (ηλιοφάνεια, clear, μερική νέφωση, partly.cloudy και νέφωση, cloudy).

Προφανώς, είναι πιο χρήσιμο να δούμε πόσο συχνά παρουσιάστηκε κάθε μία περίπτωση νέφωσης στη διάρκεια του μήνα. Έτσι, γράφουμε τα δεδομένα σε μορφή πίνακα:

```
> table(central.park.cloud)
central.park.cloud
        clear partly.cloudy        cloudy
           11            11             9
>
```

Βλέπουμε τις ονομασίες των κατηγοριών ως επικεφαλίδες και από κάτω, τις συχνότητες εμφάνισης.

04 Ραβδογράμματα (barplots).

Τα ραβδογράμματα είναι ίσως η απλούστερη γραφική παράσταση για τη σύνοψη ενός συνόλου δεδομένων. Η συνάρτηση που τα κατασκευάζει είναι η barplot( ). Στο παρακάτω παράδειγμα, εισάγουμε ένα μικρό σύνολο δεδομένων με 4 δυνατές τιμές (π.χ. ρετσίνα που προτιμούν να πίνουν 25 ερωτώμενοι φοιτητές: Μαλαματίνα, Κουρτάκη, Γεωργιάδη, δεν έχουν προτίμηση) και το σχεδιάζουμε ως ραβδόγραμμα. Θα χρησιμοποιήσουμε τη συνάρτηση scan για να εισάγουμε τα δεδομένα. Αυτή μπορεί να τα διαβάσει από ένα αρχείο, όπως έχουμε ήδη πει, αλλά μπορεί να τα διαβάσει και κατευθείαν από τη γραμμή εντολών, όπως δείχνουμε εδώ:
Πρώτα γράφουμε:
> retsina=scan()
Μετά γράφουμε τις τιμές (το 1: στην αρχή της σειράς γράφεται από το ίδιο το R, ενώ οι υπόλοιποι αριθμοί είναι τιμές που δίνουμε) και πατάμε ENTER μια φορά και άλλη μία για να τερματιστεί η εισαγωή:

```
1: 3 4 1 1 3 4 3 3 1 3 2 1 2 1 2 3 2 3 1 1 1 1 4 3 1
26: Read 25 items
>
```

Τώρα, αν γράψουμε barplot(retsina), αυτό δε θα ήταν σωστό γιατί η barplot θεωρεί ότι το σύνολο δεδομένων είναι σε μορφή πίνακα και θα "νομίσει" ότι κάθε τιμή που εισάγαμε αντιστοιχεί σε μία κατηγορία. Γράψτε το για να δείτε το διάγραμμα που θα προκύψει:

```
> barplot(retsina)
```

Για να πάρουμε τη σωστή αναπαράσταση των δεδομένων, δηλαδή το ύψος κάθε ράβδου να αντιστοιχεί στη συχνότητα κάθε προτίμησης, θα πρέπει να δώσουμε τα δεδομένα σε μορφή πίνακα. Αυτό γίνεται πολύ απλά ως εξής:

```
> barplot(table(retsina))
```

Το διάγραμμα είναι πιο παραστατικό αν εισάγουμε κατάλληλες "ετικέτες" με περιγραφικά ονόματα για τους άξονες και το διάγραμμα. Γράψτε:

```
> barplot(table(retsina), xlab="ρετσίνα", ylab="συχνότητα", main="Προτιμήσεις ρετσίνας")
```

Τι παρατηρείτε;

Τέλος, θα θέλαμε να βάλουμε ετικέτες με τα ονόματα κάθε κατηγορίας. Αν το σύνολο δεδομένων retsina είχε απευθείας τις συχνότητες των προτιμήσεων, θα μπορούσαμε να χρησιμοποιήσουμε τη συνάρτηση names (προηγούμενο μάθημα) και οι ετικέτες θα εμφανίζονταν αυτόματα. Τώρα, επειδή πρέπει να πινακοποίησουμε τα δεδομένα, θα το κάνουμε λίγο διαφορετικά:
Γράψτε:

```
> retsines = c("Μαλαματίνα", "Κουρτάκη", "Γεωργιάδη", "καμμία")
> barplot(table(retsina), xlab="ρετσίνα", ylab="συχνότητα", main="Προτιμήσεις ρετσίνας", names.arg=retsines)
```

και δείτε το διάγραμμα στην τελική του μορφή.

05 Αριθμητικά δεδομένα – η κεντρική τάση

Για να καταλάβουμε ποια είναι τα κύρια χαρακτηριστικά σε μια κατανομή αριθμητικών δεδομένων, χρησιμοποιούμε στατιστικά όπως η μέση τιμή, ο διάμεσος, η τυπική απόκλιση, τα λεγόμενα *πολλοστημόρια* κλπ.

Η κύρια τάση περιγράφεται από τη μέση τιμή, το διάμεσο και τον αριθμό κορυφών (mode) της κατανομής. Συνεχίζοντας το προηγούμενο παράδειγμα, υπολογίζουμε αυτά τα στατιστικά για το σύνολο δεδομένων που εισάγαμε.

Η μέση τιμή μπορεί να υπολογιστεί τόσο με βάση τον ορισμό της όσο και με τη συνάρτηση mean:

```
> sum(retsina)/length(retsina)
[1] 2.16
> mean(retsina)
[1] 2.16
>
```

Ο διάμεσος βρίσκεται με τη συνάρτηση median:

```
> median(retsina)
[1] 2
>
```

Η χρησιμότητα του διάμεσου στην περιγραφή μιας κατανομής φαίνεται όταν η κατανομή παρουσιάζει κάποια έντονη ασυμμετρία. Για παράδειγμα, έστω ότι σε ένα εστιατόριο βρίσκονται 5 πελάτες με εισοδήματα ενός μέσου εργαζόμενου, π.χ. μεταξύ 800 και 1200 ευρώ. Προφανώς η μέση τιμή των εισοδημάτων των πελατών βρίσκεται κάπου εκεί. Τότε, μπαίνει μέσα ένας πλούσιος με εισόδημα 100000 ευρώ. Η μέση τιμή θα αλλάξει σημαντικά, αλλά αυτό δε μας δίνει μια καλή εικόνα της κατάστασης γιατί δε μας λέει που κυμαίνονται οι περισσότερες τιμές, πράγμα που δίνεται καλύτερα από το διάμεσο.

Με ένα αριθμητικό παράδειγμα αυτό φαίνεται καλύτερα:

Εισάγουμε τιμές για μισθούς των πέντε πελατών και υπολογίζουμε τη μέση τιμή και το διάμεσο:

```
> misthoi=c(850, 1200, 900, 930, 1100)
> mean(misthoi)
[1] 996
> median(misthoi)
[1] 930
>
```

Τώρα, μπαίνει ο πλούσιος με τα 100,000 ευρώ και ξαναϋπολογίζουμε τις ποσότητες:

```
> misthoi.me.plousio=c(misthoi, 100000)
> mean(misthoi.me.plousio)
[1] 17496.67
> median(misthoi.me.plousio)
[1] 1015
>
```

Βλέπουμε ότι η μέση τιμή εκτινάχθηκε στα 17500 ευρώ περίπου, ενώ ο διάμεσος παραμένει κοντά στα 1000 που δείχνει και την πραγματική κατάσταση για τους περισσότερους παρόντες. Λέμε ότι ένα στατιστικό είναι πιο *ανθεκτικό* (resistant) όταν δεν επηρεάζεται σημαντικά από λίγα δεδομένα που βρίσκονται μακριά από τον κύριο όγκο των τιμών. Ο διάμεσος είναι πιο ανθεκτικός από τη μέση τιμή.

Μια τεχνική που κάνει τη μέση τιμή πιο ανθεκτική στις ασυμμετρίες της κατανομής συνίσταται στο να *περικόψουμε* ένα ποσοστό των μεγαλύτερων και μικρότερων τιμών των δεδομένων (trimmed mean). Το σύνολο δεδομένων cfb(UsingR) περιέχει ένα δείγμα από έρευνα για τα οικονομικά του

μέσου καταναλωτή στις ΗΠΑ το 2001. Από αυτό, μπορούμε να βρούμε την κατανομή των εισοδημάτων η οποία είναι πολύ ασύμμετρη εφόσον υπάρχουν λίγοι πολύ πλούσιοι και να εφαρμόσουμε τα παραπάνω, άλλη μια φορά. Γράψτε:

```
> income = cfb$INCOME
> mean(income)
.............. (τι τιμή προκύπτει;)
> median(income)
.............. (τι τιμή προκύπτει;)
> mean(income, trim=.2)
.............. (τι τιμή προκύπτει;)
> sum(income <= mean(income))/length(income)*100
.............. (τι τιμή προκύπτει;)
>
```

Τι παρατηρείτε όσον αφορά τη σχέση μεταξύ μέσης τιμής χωρίς και με αποκοπή (trim) του 20% καθώς και σε σύγκριση με το διάμεσο;
Η τελευταία γραμμή δίνει το ποσοστό των εισοδημάτων που είναι κάτω από τη μέση τιμή. Τι τιμή προκύπτει και τι συμπεραίνετε από αυτό;
Η κορυφή ενός συνόλου δεδομένων είναι η πιο κοινή τιμή αυτού. Δεν υπάρχει συνάρτηση που να τη δίνει απευθείας, αλλά μπορούμε να τη βρούμε με συνδυασμό των προηγούμενων συναρτήσεων. Γράψτε:

```
> x = c(72, 75, 84, 84, 98, 94, 55, 62)
> which(table(x)==max(table(x)))
```

Τι αποτέλεσμα πήρατε;

06 Αριθμητικά δεδομένα – η διακύμανση

Εκτός από τη διασπορά και την τυπική απόκλιση, ένας τρόπος για να δούμε τη διασπορά των δεδομένων είναι τα *πολλοστημόρια* (π.χ. τριτημόρια, τεταρτημόρια κλπ) που γενικεύουν την έννοια του διάμεσου. Αν ο διάμεσος χωρίζει τα δεδομένα στη μέση, το p-πολλοστημόριο χωρίζει τα δεδομένα σε διαστήματα που είναι 100p% μικρότερα και 100(1-p)% μεγαλύτερα από την τιμή αυτή. Το p-πολλοστημόριο, όπου p είναι π.χ. ¼ (τεταρτημόριο) ή 1/8 (ογδοημόριο) ή άλλο ποσοστό, δίνεται από τη σχέση 1+ p(n-1) όπου n ο αριθμός των δεδομένων. Τα πολλοστημόρια που αντιστοιχούν στα ποσοστά 0, 25, 50, 75 και 100%, δίνονται από τη συνάρτηση quantile. Παράδειγμα:

```
> x = 0:5
> x
[1] 0 1 2 3 4 5
> quantile(x)
  0%  25%  50%  75% 100%
0.00 1.25 2.50 3.75 5.00
> quantile(x, 0.25)
 25%
1.25
> quantile(x, c(0.25, 0.5, 0.75))
 25%  50%  75%
1.25 2.50 3.75
>
```

Δυο άλλες ενδιαφέρουσες συναρτήσεις είναι η τυπική απόκλιση sd( ) και η IQR( ). Η τελευταία δίνει την απόσταση ανάμεσα στα πολλοστημόρια 25% και 75% και είναι πιο ανθεκτική από την τυπική απόκλιση. Πειραματιστείτε με δικά σας δεδομένα τα οποία μεταβάλλετε ώστε να είναι όλο και πιο ασύμμετρα και παρατηρείστε πώς συμπεριφέρονται οι δύο συναρτήσεις.