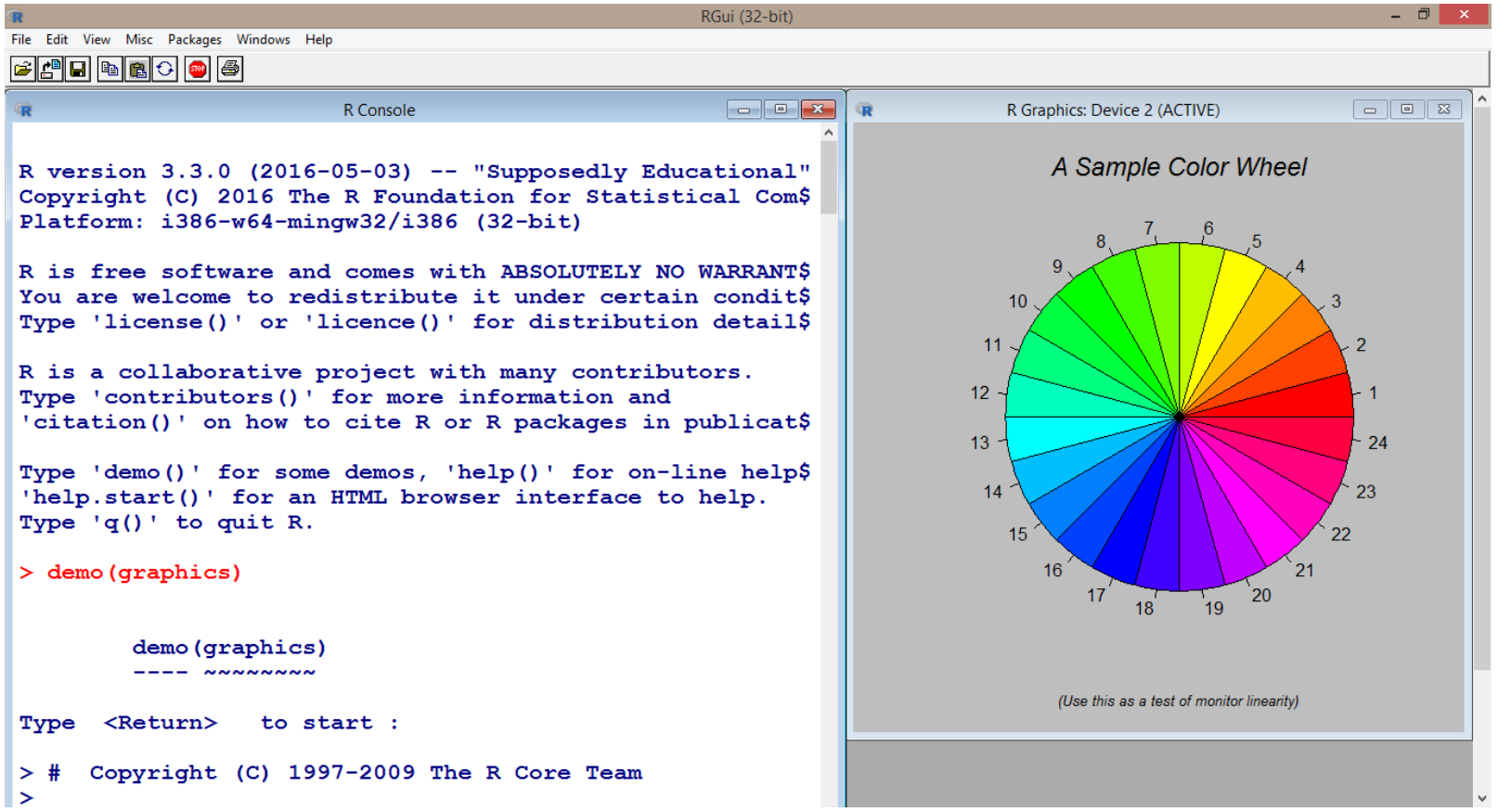
Εισαγωγή στη γλώσσα R

- Απλές πράξεις
- Διανύσματα
- Εισαγωγή δεδομένων
- Στατιστικές Συναρτήσεις
- Γραφικές παραστάσεις

Το περιβάλλον

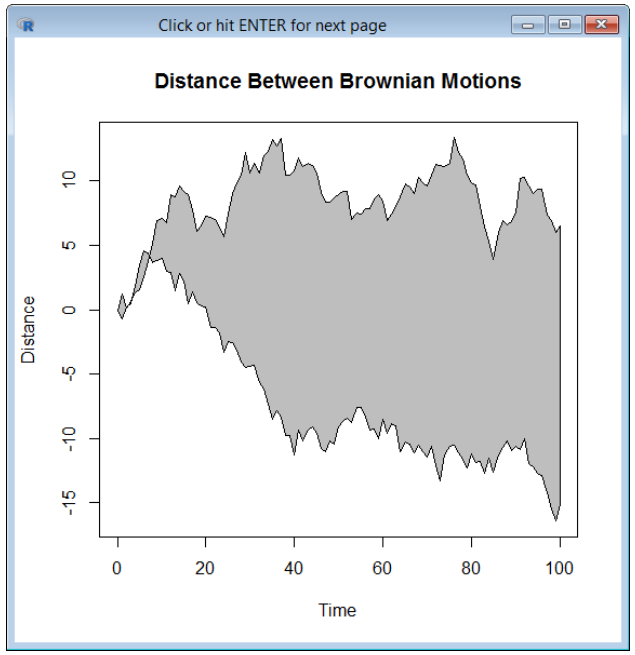
Λογισμικό που αναπτύχθηκε στα Bell Laboratories και εξελίχθηκε στο

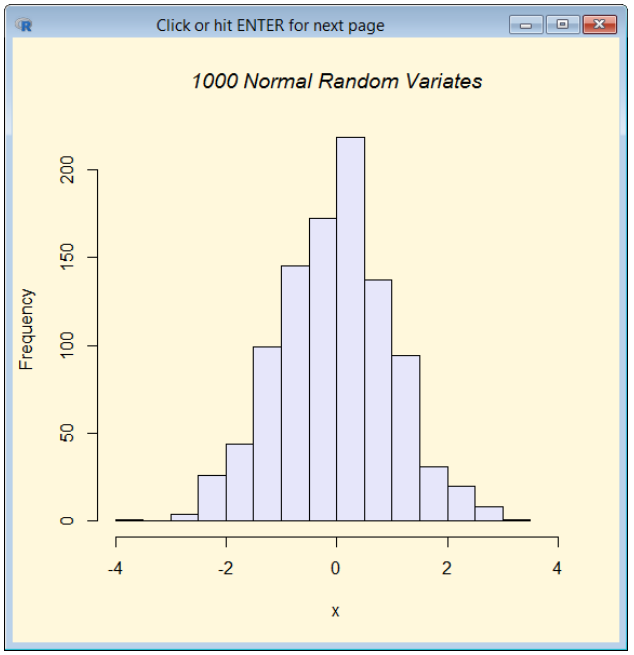
- εμπορικό S-Plus
- ελεύθερο R (περίπου το 1997)
  - ✓ ανάπτυξη διαφόρων packages από πληθώρα ερευνητών

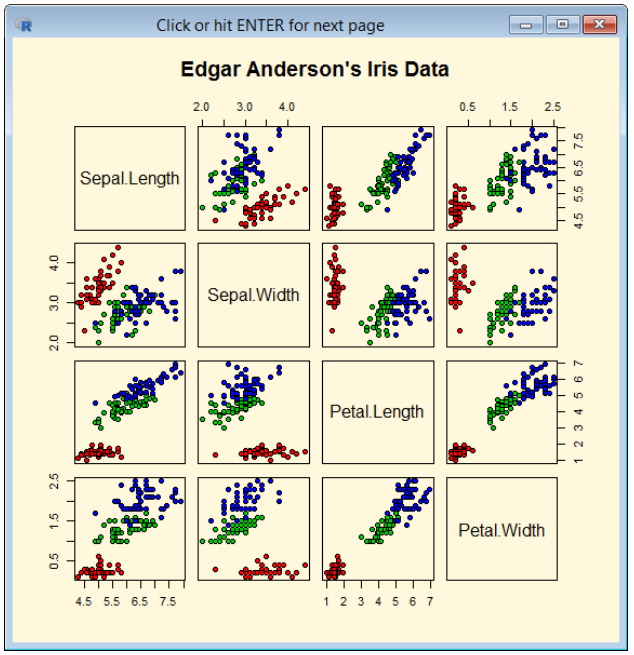


Το περιβάλλον

> demo(graphics)







- θεωρητική μελέτη, με χρήση μεθόδων προσομοίωσης
- παρουσίαση και ανάλυση πραγματικών δεδομένων
  - ✓ εργαλεία περιγραφικής στατιστικής
  - ✓ μέθοδοι ανάλυσης (discriminant analysis)

Το περιβάλλον

> help()

Usage

```
help(topic, package = NULL, lib.loc = NULL,
     verbose = getOption("verbose"),
     try.all.packages = getOption("help.try.all.packages"),
     help_type = getOption("help_type"))
```

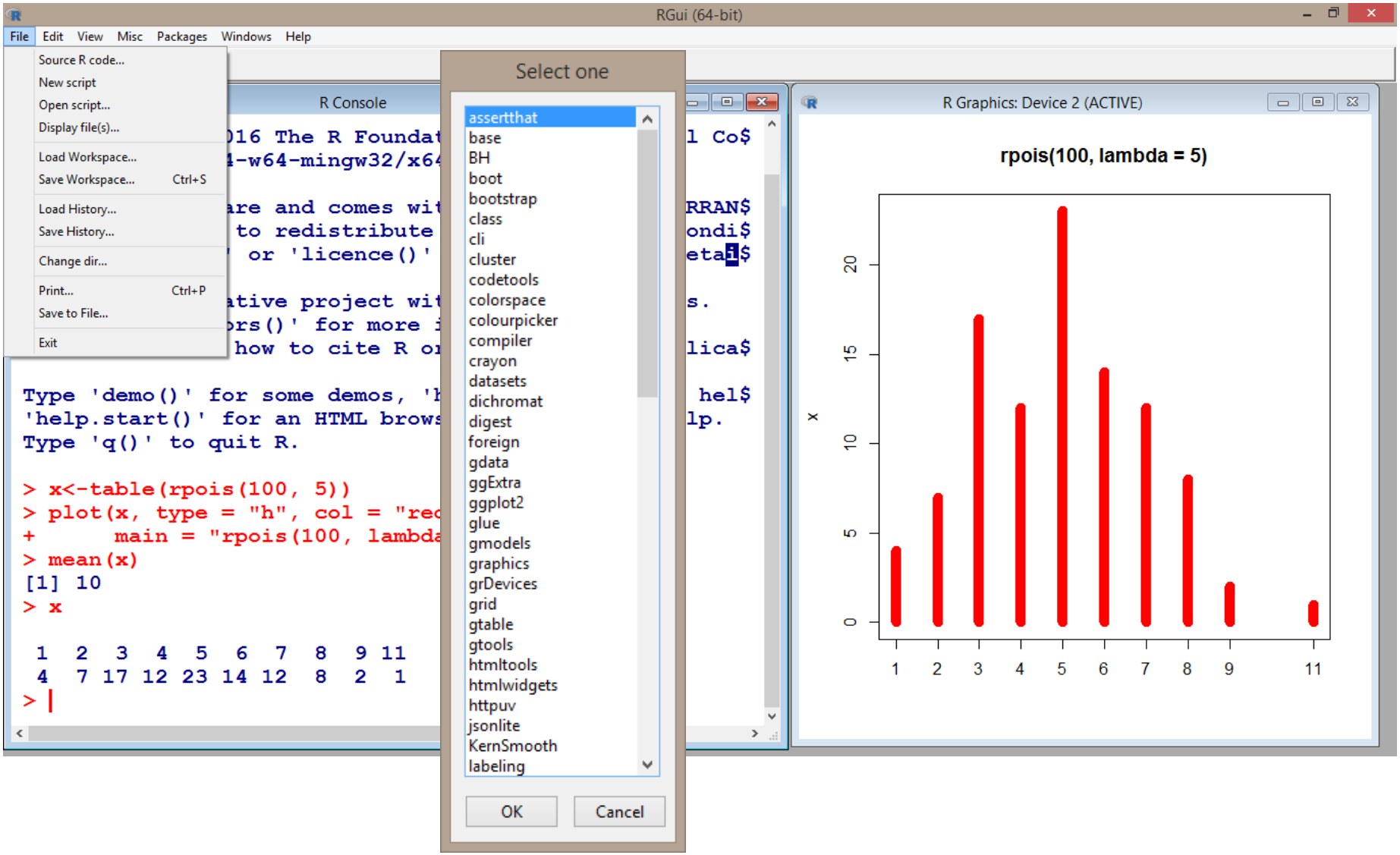
Arguments

topic	usually, a name or character string specifying the topic for which help is sought. A character string (enclosed in explicit single or double quotes) is always taken as naming a topic. If the value of `topic` is a length-one character vector the topic is taken to be the value of the only element. Otherwise `topic` must be a name or a reserved word (if syntactically valid) or character string. See 'Details' for what happens if this is omitted.
package	a name or character vector giving the packages to look into for documentation, or `NULL`. By default, all packages whose namespaces are loaded are used. To avoid a name being deparsed use e.g. `(pkg_ref)` (see the examples).
lib.loc	a character vector of directory names of `R` libraries, or `NULL`. The default value of `NULL` corresponds to all libraries currently known. If the default is used, the loaded packages are searched before the libraries. This is not used for HTML help (see 'Details'.
verbose	logical; if `TRUE`, the file name is reported.
try.all.packages	logical; see `Note`.
help_type	character string: the type of help required. Possible values are `"text"`, `"html"` and `"pdf"`. Case is ignored, and partial matching is allowed.

Examples

```
help()
help(help)              # the same

help(lapply)

help("for")             # or ?"for", but quotes/backticks are needed

help(package = "splines") # get help even when package is not loaded

topi <- "women"
help(topi)
```

Το περιβάλλον



Απλές πράξεις

```
> 3+5
[1] 8
> 3*9
[1] 27
> 27 / 9
[1] 3
> 3^2
[1] 9
> 8^(1/3)
[1] 2
> 3^-2
[1] 0.1111111
> -4^(1/3)
[1] -1.587401
> (-4)^2
[1] 16
> (-4)^2.00000000000001
[1] NaN
```

```
> log(10)
[1] 2.302585
> exp(1)
[1] 2.718282
> log10(6)
[1] 0.7781513
> log(9,3)
[1] 2
> sin(pi/2)
[1] 1
> cos(pi/2)
[1] 6.123032e-17
> cos(pi/2)==0
[1] FALSE
> sin(pi/2)==1
[1] TRUE
> floor(5.7)
[1] 5
> ceiling(5.7)
[1] 6
> round(5.77454,3)
[1] 5.775
> round(5.77456,2)
[1] 5.77
```

Διανύσματα

```
> x<-c(5,7,7,7,9,8)    #  Dnmioupgeia stnlns me 6 stoix$
> x
[1] 5 7 7 7 9 8
> x[2]                 # to stoixeio 2 autns tns stnlns$
[1] 7
> length(x)# to mnkos tns stnlns
[1] 6
> x[length(x)]# to teleutaio stoixeio tns stnlns
[1] 8
> # praxeis me diavusmata
> 2*x
[1] 10 14 14 14 18 16
> x+3
[1]  8 10 10 10 12 11
> x*x
[1] 25 49 49 49 81 64
> x^2
[1] 25 49 49 49 81 64
> y<-x+3
> y
[1]  8 10 10 10 12 11
> x*y
[1]  40  70  70  70 108  88
> sum(x^2)
[1] 317
```

```
> t(x)
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    5    7    7    7    9    8
> x%*%t(x)
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]   25   35   35   35   45   40
[2,]   35   49   49   49   63   56
[3,]   35   49   49   49   63   56
[4,]   35   49   49   49   63   56
[5,]   45   63   63   63   81   72
[6,]   40   56   56   56   72   64
> t(x)%*%x
     [,1]
[1,]  317
```

Διανύσματα

```
> sort(x)
[1] 5 7 7 7 8 9
> rev(sort(x))
[1] 9 8 7 7 7 5
> max(x)
[1] 9
> min(x)
[1] 5
> x[x>=7]# ta stoixeia tou x pou eivai >=7
[1] 7 7 7 9 8
> which(x>=7)
[1] 2 3 4 5 6
> sum(x)
[1] 43
> sum(x>=7)                    # posa stoixeia tou x eivai >=7
[1] 5
> sum(x[x>=7])                 # to a8poisma autwv
[1] 38
>
> #  poses fopes emfavizetai ka8e stoixeio (movo auta $
> table(x)
x
5 7 8 9
1 3 1 1
```

Διανύσματα

```
> x<-c(rep(0,10))# stnln me 10 mndevika
> x
 [1] 0 0 0 0 0 0 0 0 0 0
> x<-c(1:10)# stnln me tous api8mous 1,2,3,...,10
> x
 [1]  1  2  3  4  5  6  7  8  9 10
> #   mndevikos pivakas me 2 grammes kai 5 stnles
> x<-matrix(c(rep(0,10)),nrow=2,ncol=5)
> x
     [,1] [,2] [,3] [,4] [,5]
[1,]    0    0    0    0    0
[2,]    0    0    0    0    0
> x<-matrix(c(1:10),nrow=2,ncol=5) #  1,2,...,10 topo8etn$
> x
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    3    5    7    9
[2,]    2    4    6    8   10
> #   1,2,...,10 topo8etnmeva kata grammes
> x<-matrix(c(1:10),nrow=2,ncol=5,byrow=T)
> x
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    2    3    4    5
[2,]    6    7    8    9   10
```

Εισαγωγή δεδομένων

the speed of light =

299 792.458 kilometer / second

In 1876, the French physicist Cornu reported a value of 299,990 km/sec for c, the speed of light. In 1879, the American physicist A.A. Michelson carried out several experiments to verify and improve on Cornu's value.

Michelson obtained the following 20 measurements of the speed of light:

850	740	900	1070	930	850	950	980	980	880
1000	980	930	650	760	810	1000	1000	960	960

To obtain Michelson's actual measurements in km/sec, add 299,000 km/sec to each of the above values.

Εισαγωγή δεδομένων

```
> mich.df<-scan()
1: 850
2: 740
3: 900
4: 1070
5: 930
6: 850
7: 950
8: 980
9: 980
10: 880
11:
Read 10 items
> mich.df<-c(mich.df,1000,980,930,650,760,810,1000,1000,960,960)
> mich.df
 [1]  850  740  900 1070  930  850  950  980  980  880
[11] 1000  980  930  650  760  810 1000 1000  960  960
> z<-mich.df
> summary(z)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    650     850     940     909     980    1070
> summary(z)[1]
Min.
 650
> summary(z)[5]
3rd Qu.
    980
```

Εισαγωγή δεδομένων

```
> kyphosis<-read.table("C:\\kyphosis.txt",header=TRUE)
> kyphosis
   Kyphosis Age Number Start
1    absent  71      3     5
2    absent 158      3    14
3   present 128      4     5
4    absent   2      5     1
5    absent   1      4    15
6    absent   1      2    16
7    absent  61      2    17
8    absent  37      3    16
9    absent 113      2    16
10  present  59      6    12
11  present  82      5    14
12   absent 148      3    16
13   absent  18      5     2
14   absent   1      4    12
15   absent 168      3    18
16   absent   1      3    16
17   absent  78      6    15
18   absent 175      5    13
19   absent  80      5    16
20   absent  27      4     9
21   absent  22      3    16
```

The four variables in `kyphosis` are defined as follows:

- `Kyphosis`

  A binary variable indicating the presence/absence of a postoperative spinal deformity called *Kyphosis*.

- `Age`

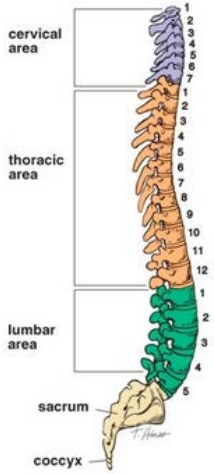
  The age of the child in months.

- `Number`

  The number of vertebrae involved in the spinal operation.

- `Start`

  The beginning of the range of the vertebrae involved in the operation.



Εισαγωγή δεδομένων

```
> kyphosis[1,]
  Kyphosis Age Number Start
1   absent  71      3     5
> kyphosis[,2]
 [1]  71 158 128   2   1   1  61  37 113  59  82 148  18   1 168
[16]   1  78 175  80  27  22 105  96 131  15   9   8 100   4 151
[31]  31 125 130 112 140  93   1  52  20  91  73  35 143  61  97
[46] 139 136 131 121 177  68   9 139   2 140  72   2 120  51 102
[61] 130 114  81 118 118  17 195 159  18  15 158 127  87 206  11
[76] 178 157  26 120  42  36
> kyphosis$Age
 [1]  71 158 128   2   1   1  61  37 113  59  82 148  18   1 168
[16]   1  78 175  80  27  22 105  96 131  15   9   8 100   4 151
[31]  31 125 130 112 140  93   1  52  20  91  73  35 143  61  97
[46] 139 136 131 121 177  68   9 139   2 140  72   2 120  51 102
[61] 130 114  81 118 118  17 195 159  18  15 158 127  87 206  11
[76] 178 157  26 120  42  36
> attach(kyphosis)
> Age
 [1]  71 158 128   2   1   1  61  37 113  59  82 148  18   1 168
[16]   1  78 175  80  27  22 105  96 131  15   9   8 100   4 151
[31]  31 125 130 112 140  93   1  52  20  91  73  35 143  61  97
[46] 139 136 131 121 177  68   9 139   2 140  72   2 120  51 102
[61] 130 114  81 118 118  17 195 159  18  15 158 127  87 206  11
[76] 178 157  26 120  42  36
```

Δεδομένα που υπάρχουν στην R

InsectSprays {datasets} R Documentation

Effectiveness of Insect Sprays

Description

The counts of insects in agricultural experimental units treated with different insecticides.

Usage

```
InsectSprays
```

Format

A data frame with 72 observations on 2 variables.

[,1] count numeric Insect count

[,2] spray factor The type of spray

Source

Beall, G., (1942) The Transformation of data from entomological field experiments, *Biometrika*, **29**, 243–262.

```
> InsectSprays
   count spray
1     10     A
2      7     A
3     20     A
4     14     A
5     14     A
6     12     A
7     10     A
8     23     A
9     17     A
10    20     A
11    14     A
12    13     A
13    11     B
14    17     B
15    21     B
16    11     B
17    16     B
18    14     B
19    17     B
20    17     B
21    19     B
22    21     B
```

Δεδομένα που υπάρχουν στην R

ToothGrowth {datasets} R Documentation

The Effect of Vitamin C on Tooth Growth in Guinea Pigs

Description

The response is the length of odontoblasts (cells responsible for tooth growth) in 60 guinea pigs. Each animal received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods, (orange juice or ascorbic acid (a form of vitamin C and coded as VC).

Usage

```
ToothGrowth
```

Format

A data frame with 60 observations on 3 variables.

[,1] len numeric Tooth length
[,2] supp factor Supplement type (VC or OJ).
[,3] dose numeric Dose in milligrams/day

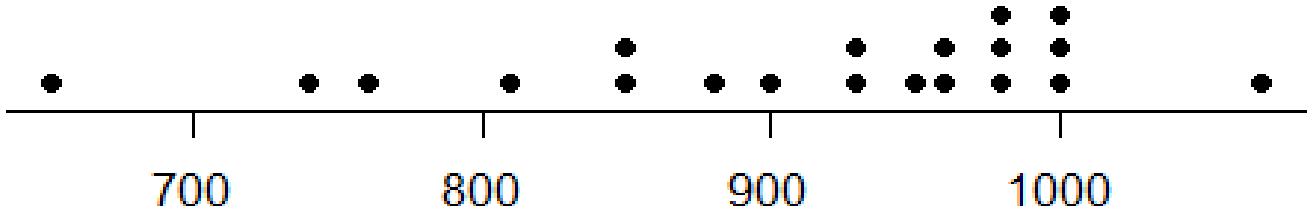
Source

C. I. Bliss (1952) *The Statistics of Bioassay*. Academic Press.

```
> ToothGrowth
    len supp dose
1   4.2   VC  0.5
2  11.5   VC  0.5
3   7.3   VC  0.5
4   5.8   VC  0.5
5   6.4   VC  0.5
6  10.0   VC  0.5
7  11.2   VC  0.5
8  11.2   VC  0.5
9   5.2   VC  0.5
10  7.0   VC  0.5
11 16.5   VC  1.0
12 16.5   VC  1.0
13 15.2   VC  1.0
14 17.3   VC  1.0
15 22.5   VC  1.0
16 17.3   VC  1.0
17 13.6   VC  1.0
18 14.5   VC  1.0
19 18.8   VC  1.0
20 15.5   VC  1.0
21 23.6   VC  2.0
22 18.5   VC  2.0
```

Στατιστικές Συναρτήσεις

```
> mich.df
 [1]  850  740  900 1070  930  850  950  980  980  880
[11] 1000  980  930  650  760  810 1000 1000  960  960
```




```
> quantile(mich.df,probs=c(.25,.50,.75)) # posostimopia
25% 50% 75%
850 940 980
> mean(mich.df)# deigmatikos mesos
[1] 909
> sum((mich.df-mean(mich.df))^2)/(length(mich.df)-1)
[1] 11009.47
> var(mich.df)# diaspora
[1] 11009.47
> sqrt(var(mich.df))
[1] 104.926
> sd(mich.df)# tupikn apoklisn
[1] 104.926
```

Στατιστικές Συναρτήσεις

All sample quantiles are defined as weighted averages of consecutive order statistics. Sample quantiles of type i are defined by:

$$Q[i](p) = (1 - \gamma)\, x[j] + \gamma\, x[j+1],$$

where $1 \le i \le 9$, $(j-m)/n \le p < (j-m+1)/n$, $x[j]$ is the jth order statistic, n is the sample size, the value of γ is a function of $j = floor(np + m)$ and $g = np + m - j$, and m is a constant determined by the sample quantile type.

Continuous sample quantile types 4 through 9

For types 4 through 9, $Q[i](p)$ is a continuous function of p, with $gamma = g$ and m given below. The sample quantiles can be obtained equivalently by linear interpolation between the points $(p[k],x[k])$ where $x[k]$ is the kth order statistic. Specific expressions for $p[k]$ are given below.

Type 6

$m = p$. $p[k] = k / (n + 1)$. Thus $p[k] = E[F(x[k])]$. This is used by Minitab and by SPSS.

Type 7

$m = 1-p$. $p[k] = (k - 1) / (n - 1)$. In this case, $p[k] = mode[F(x[k])]$. This is used by S.

Στατιστικές Συναρτήσεις

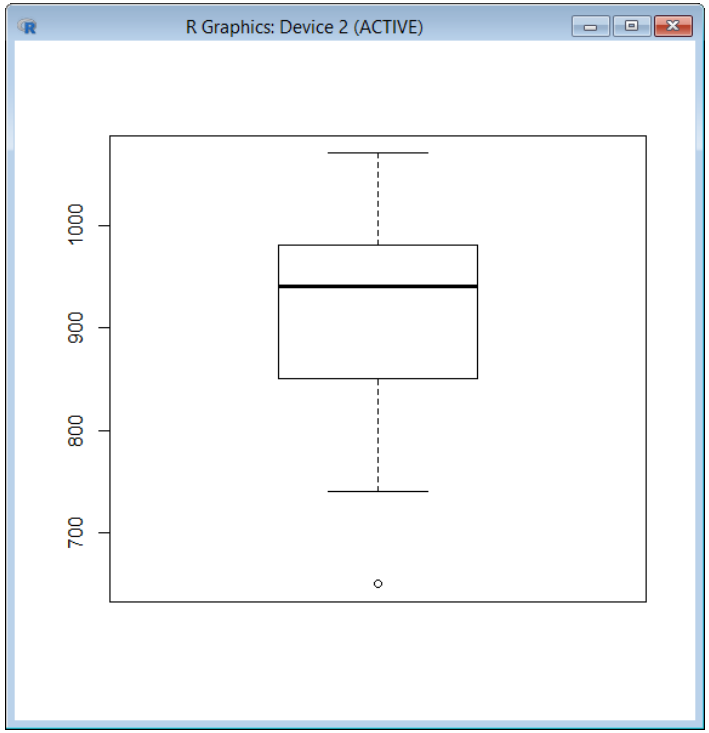
```
var(x, y = NULL, na.rm = FALSE, use)

cov(x, y = NULL, use = "everything",
    method = c("pearson", "kendall", "spearman"))

cor(x, y = NULL, use = "everything",
    method = c("pearson", "kendall", "spearman"))
```

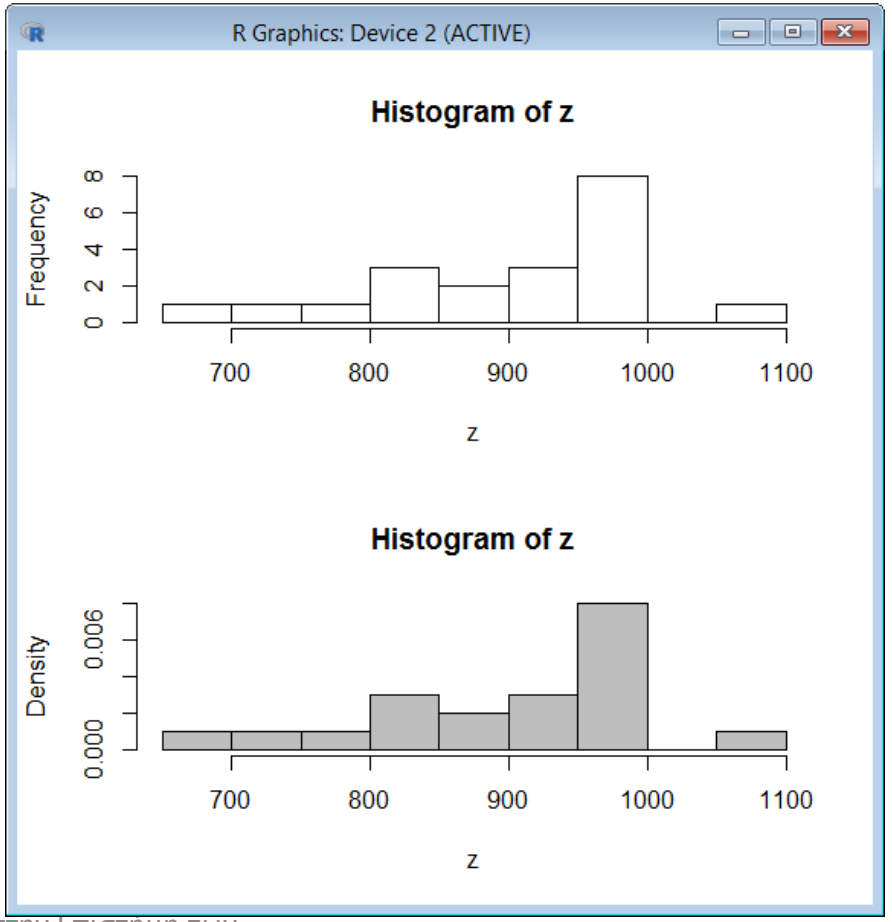
```
> var
function (x, y = NULL, na.rm = FALSE, use)
{
    if (missing(use))
        use <- if (na.rm)
            "na.or.complete"
        else "everything"
    na.method <- pmatch(use, c("all.obs", "complete.obs", "$
        "everything", "na.or.complete"))
    if (is.na(na.method))
        stop("invalid 'use' argument")
    if (is.data.frame(x))
        x <- as.matrix(x)
    else stopifnot(is.atomic(x))
    if (is.data.frame(y))
        y <- as.matrix(y)
    else stopifnot(is.atomic(y))
    .Call(C_cov, x, y, na.method, FALSE)
}
```

Γραφικές παραστάσεις

```
> boxplot(mich.df)
```

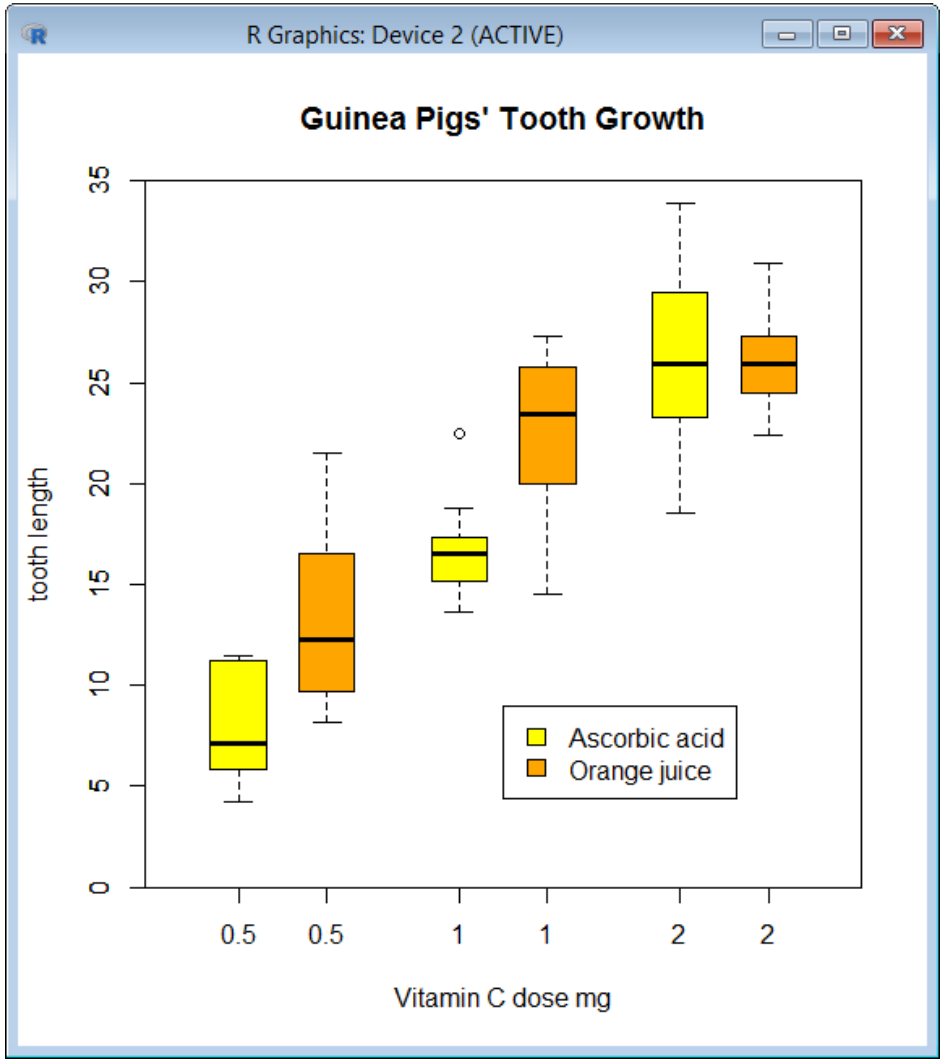
```
> par(mfrow=c(2,1))              # duo grafi
> hist(z)# istogramma suxvotntwv
Waiting to confirm page change...
> hist(z,probability=T,col="grey")
> par(mfrow=c(1,1))
```





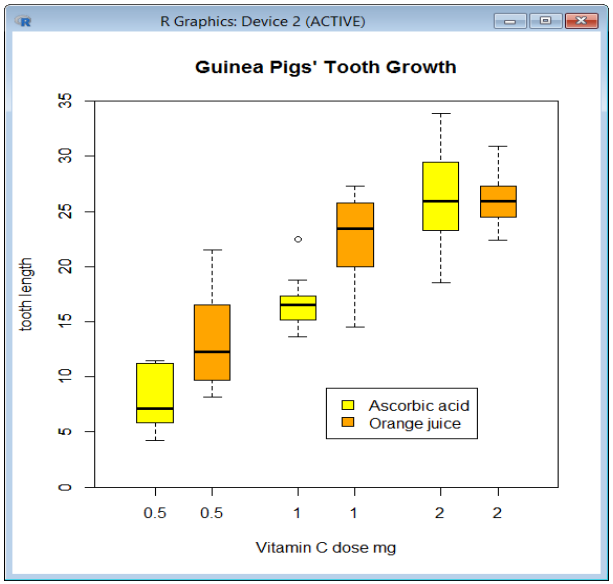
Γραφικές παραστάσεις

```
> devAskNewPage(ask=TRUE)      # perimevei priv ftia3ei ve$
> ToothGrowth
    len supp dose
1   4.2   VC  0.5
2  11.5   VC  0.5
3   7.3   VC  0.5

24 25.5   VC  2.0
25 26.4   VC  2.0
26 32.5   VC  2.0
27 26.7   VC  2.0
28 21.5   VC  2.0
29 23.3   VC  2.0
30 29.5   VC  2.0
31 15.2   OJ  0.5
32 21.5   OJ  0.5
33 17.6   OJ  0.5
34  9.7   OJ  0.5
35 14.5   OJ  0.5
36 10.0   OJ  0.5
```



Γραφικές παραστάσεις

```
> boxplot(len ~ dose, data = ToothGrowth,
+                 boxwex = 0.25, at = 1:3 - 0.2,
+                 subset = supp == "VC", col = "yellow",
+                 main = "Guinea Pigs' Tooth Growth",
+                 xlab = "Vitamin C dose mg",
+                 ylab = "tooth length",
+                 xlim = c(0.5, 3.5), ylim = c(0, 35), yaxs = "i")
Waiting to confirm page change...
>       boxplot(len ~ dose, data = ToothGrowth, add = TRUE,
+                 boxwex = 0.25, at = 1:3 + 0.2,
+                 subset = supp == "OJ", col = "orange")
>       legend(2, 9, c("Ascorbic acid", "Orange juice"),
+              fill = c("yellow", "orange"))
```



Γραφικές παραστάσεις

```
> par(mfrow=c(1,1))
> par(mfrow=c(1,2))
> z<-rnorm(5000) # tuxaio deigma mege8ous 5000 apo N(0,1)
> hist(z,nclass=25,col=4,xlab="x",main="Histogram of N(0,1)$
Waiting to confirm page change...
> plot(dnorm,xlim=c(-4,4))
> par(mfrow=c(1,1))
>
> mean(z)
[1] 0.01848778
> sd(z)
[1] 1.002524
```