

# R1-GettingStart

Hossein Vatani

March 18, 2016

## بنام خدا

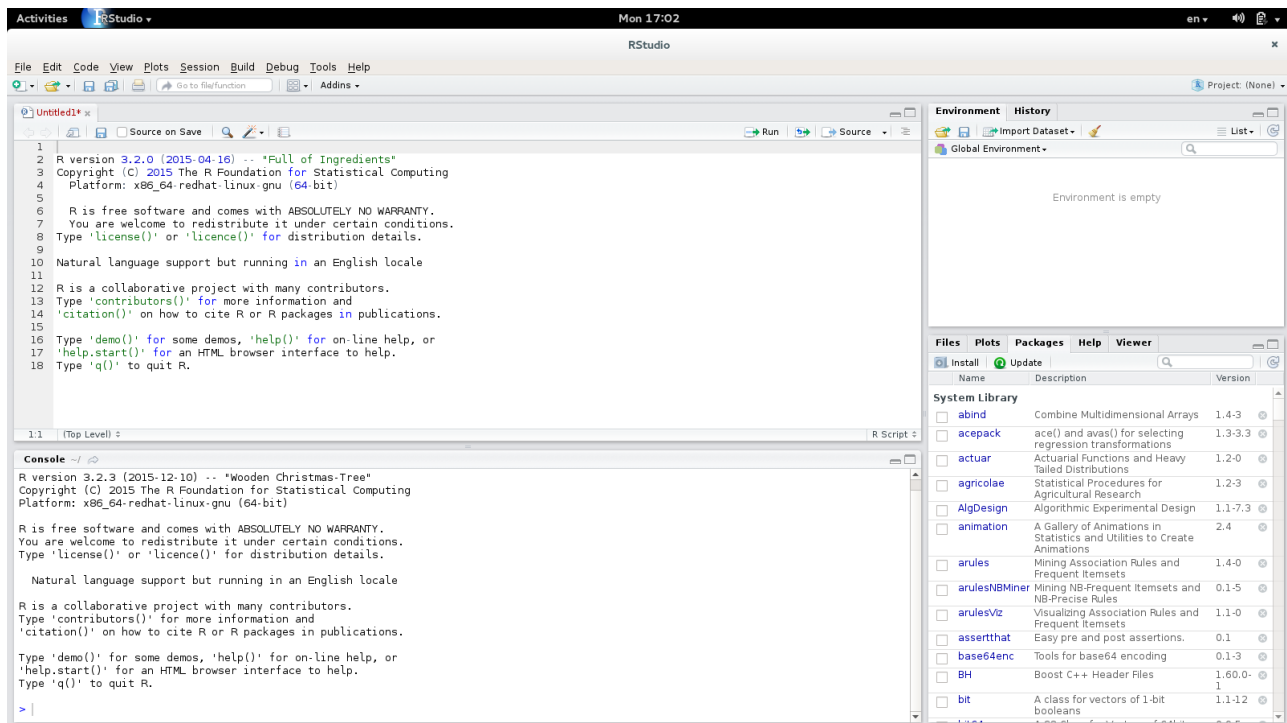
## شروع کار با نرم افزار R

مقدمه متن پیش رو ترجمه ای با اندکی تلخیص از یازده مقاله دانشگاه کلمبیا آمریکا در مورد نرم افزار قدرتمند R می باشد. متن انگلیسی آن نیز در همین سایت موجود است.

## ۱-آماده سازی نرم افزار

نرم افزار R یک زبان برنامه نویسی و محیطی برای انجام عملیات و محاسبات آماری و همینطور نمایش آن می باشد. این زبان در سالهای اخیر بخاطر انعطاف بسیار زیاد خود، جهت توسعه و توضیح مدل‌های جدید آماری بسیار مورد توجه قرار گرفته است. این نرم افزار در تمام آزمایشگاه‌های سراسر دانشگاه کلمبیا آمریکا نصب و در حال استفاده می باشد. نرم افزار R بصورت رایگان و متن باز و تحت موز GNU می باشد که شما می توانید آنرا از آدرس زیر دریافت و نصب نمایید. سایت اصلی نرم افزار (<http://www.r-project.org>)

نرم افزار R بصورت محیط فرمان (comman Line) می باشد که البته برای آن واسط‌های گرافیکی خوبی طراحی شده است. یکی از بهترین و معروفترین آنها واسط گرافیکی RStudio می باشد که از آدرس دریافت واسط گرافیکی (<https://www.rstudio.com/products/rstudio/download>). پس از نصب این رابط گرافیکی و اجرای آن با محیطی مانند تصویر پایین مواجه خواهید شد.



## محیط نرم افزار RStudio

با توجه به این موضوع که در این سری از مقاله ها قصد آشنایی با عملکرد و نحوه استفاده از نرم افزار R می باشد، به همین توضیح که قسمت پایین سمت چپ محیط خط فرمان و محل اجرا کردن دستورات این مطالب می باشد، بسنده می کنیم. در هر محیط خط فرمانی یک علامت جهت مشخص کردن ابتدای خط فرمان وجود دارد که برای محیط R علامت ">" می باشد. برای این نرم افزار مجموعه داده (Datasets) های فراوانی جهت کار و یادگیری ارائه شده است که برخی با نصب خود زبان R در نرم افزار نصب می شوند، برخی به همراه بسته های جدید نصب می شوند و برخی دیگر نیز خود بتهایی قابل نصب بر روی این نرم افزار هستند. جهت دیدن لیست مجموعه داده های موجود از دستور زیر استفاده کنید

جهت نمایش تمام مجموعه داده های موجود در نرم افزار نصب شد `library(help = "datasets")` ده در این سیستم

سه دستور زیر را نیز ملاحظه نمایید

جهت نمایش تمامی اشیاء تعریف شده در نرم افزار، بدیهی است اجرا در ابتدا `objects()`  
ی اجرای برنامه مقدار خالی برمی گرداند  
جهت نمایش صفحه راهنمای دستور `help(command_name)`  
جهت خروج از برنامه `q()`

با اجرای آخرین دستور از شما پرسیده خواهد که آیا تمایل به ذخیره برنامه دارید یا خیر.

## ۲- بیاید شروع کنیم

### الف- عملگرها

شما می توانید تمام عملیات ها، دستورات و تابع های معمولی موجود در ریاضیات را بسادگی اجرا نمایید.

```

2 + 2
9 - 2
10*10
25/5
3 ^ 2      # عدد سه بتوان دو.
100^(1/2)
sqrt(100)  # ریشه عدد ۱۰۰.
log(10)    # لگاریتم طبیعی عدد ۱۰.
log10(1000) # لگاریتم عدد ۱۰۰۰ در مبنای ۱۰.
exp(2)     # بتوان دو e عدد.
cos(pi/4)  # تابع کسینوس.

```

## ب- بردار و ماتریس

نگاه عملگرهای R و ماهیت عمل آنها بر یک مجموعه ای از داده هاست، عبارتی دیگر در R داده ای بعنوان عدد یا رشته نداریم بلکه برداری عددی یا رشته ای ویا... داریم. ساده ترین نوع داده ای بردار می باشد که آرایه ای از عددها ویا حرفها می باشد.

```

x = c(3,4,7,8) # ایجاد یک بردار با اعداد 3,4,7,8
x # نمایش آن

```

```
## [1] 3 4 7 8
```

```

y = 1:4 # ایجاد یک تابع از عدد ۱ تا ۴
y

```

```
## [1] 1 2 3 4
```

```

z = seq(0,1,0.1) # ایجاد یک بردار از عدد ۰ تا ۱ باگام ۰.۱
z

```

```
## [1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
```

اگر تابعی مانند seq نا آشناست می توانید با دستور راهنما توضیحات آنرا مشاهده نمایید

جهت دسترسی به هریک از اعضای بردار

```
x[1]
```

```
## [1] 3
```

```
z[1:3]
```

```
## [1] 0.0 0.1 0.2
```

اکنون نحوه استفاده از عملگرها بروی این بردارها (پایه ای ترین نوع داده) را باهم ملاحظه می کنیم.

```
x+y
```

```
## [1] 4 6 10 12
```

*لطفا توجه نمایید که چون یکی از عملوندها از دیگری کوچکتر است، ادامه ضرب از ابتدای عضو  $x*z$  کوچکتر ادامه می یابد*

```
## Warning in x * z: longer object length is not a multiple of shorter object  
## length
```

```
## [1] 0.0 0.4 1.4 2.4 1.2 2.0 4.2 5.6 2.4 3.6 7.0
```

```
sqrt(x)
```

```
## [1] 1.732051 2.000000 2.645751 2.828427
```

```
y/x
```

```
## [1] 0.3333333 0.5000000 0.4285714 0.5000000
```

```
z^2
```

```
## [1] 0.00 0.01 0.04 0.09 0.16 0.25 0.36 0.49 0.64 0.81 1.00
```

*یا هرچیز دیگری که دوست دارید*  $\cos(z)+x*3/\log(y)$

```
## Warning in cos(z) + x * 3/log(y): longer object length is not a multiple o  
f  
## shorter object length
```

```
## [1]      Inf 18.30734 20.09509 18.26768      Inf 18.18992 19.94036
## [8] 18.07718      Inf 17.93395 19.65533
```

بردارها را می توان از نوع حروف و رشته ای نیز تعریف کرد.

```
Names = c("Mohammad","Ali","Hassan","Hossein")
Names
```

```
## [1] "Mohammad" "Ali"      "Hassan"   "Hossein"
```

اما یک بردار از دو نوع عددی و رشته ای

```
Combine=c(1:3,letters[1:3])
Combine
```

```
## [1] "1" "2" "3" "a" "b" "c"
```

نوع داده بعدی ، ماتریس می باشد که در چند حالت می توان تعریف کرد.

```
M1=matrix(c(x,y), nrow = 2, ncol = 4)
M1
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    3    7    1    3
## [2,]    4    8    2    4
```

```
M1[1,]
```

```
## [1] 3 7 1 3
```

```
M1[,2]
```

```
## [1] 7 8
```

```
M1[1:2,3:4]
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

```
M2=cbind(x,y)
M2
```

```
##      x y
## [1,] 3 1
## [2,] 4 2
## [3,] 7 3
## [4,] 8 4
```

```
M3=rbind(x,y)
M3
```

```
##      [,1] [,2] [,3] [,4]
## x      3    4    7    8
## y      1    2    3    4
```

توضیح آنکه

`nrow` یعنی تعداد سطرها  
`ncol` به معنی تعداد ستونها  
`cbind` یعنی هریک از نوع-داده های آورده شده در رابطه (که در اینجا از نوع بردار می باشند) را یک ستون در نظر بگیر  
`rbind` همان بالایی فقط بصورت سر

عملیات های روی ماتریس

```
M4=M1%%M2 # ضرب ماتریس ها
M4
```

```
##      x y
## [1,] 68 32
## [2,] 90 42
```

```
t(M4) # ترانپاده ماتریس
```

```
##      [,1] [,2]
## x      68   90
## y      32   42
```

```
solve(M4) # معکوس کاتریس
```

```
##      [,1]      [,2]
## x  -1.75   1.333333
## y   3.75  -2.833333
```

## ج-قاب-داده(Data Frame)

پرکاربردترین نوع داده می باشد که تقریباً می توان گفت همان نوع ماتریس است با این تفاوت که نوع داده ستونها می تواند متفاوت باشد. در واقع لازم است که طول ستونها باهم برابر باشد. البته دونهوع داده Na و NaN وجود دارد که در صورت برابر نبودن طول ستونها با توجه به نوع دستورات داده شده، برای یکسان سازی طول داده ها R آنها را در انتهای ستون های کوچکتر اضافه می کند.

```
A=c(TRUE,TRUE,FALSE,FALSE)
B=c("Mohammad","Ali","Hassan","Hossein")
Df=data.frame(x,A,B)
Df
```

```
##   x      A      B
## 1 3  TRUE Mohammad
## 2 4  TRUE      Ali
## 3 7 FALSE  Hassan
## 4 8 FALSE Hossein
```

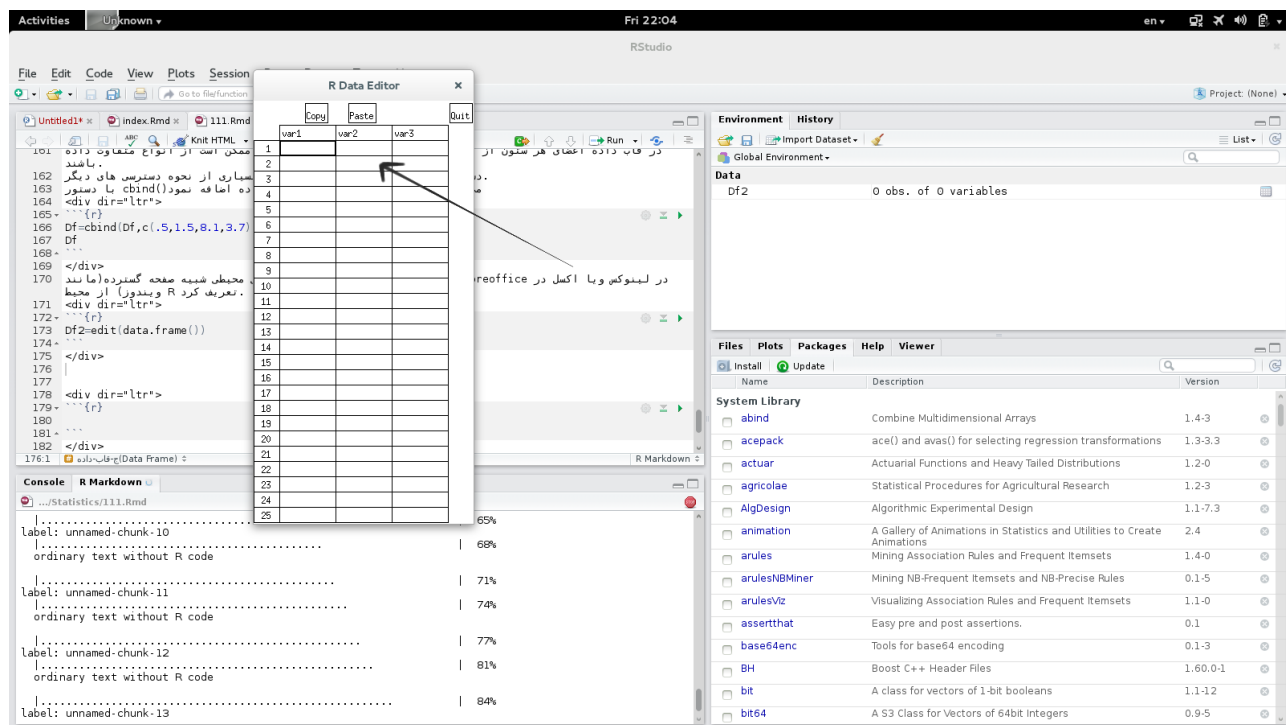
در قاب داده اعضای هر ستون از یک نوع داده می باشد و اعضای هر سطر ممکن است از انواع متفاوت داده باشند. دسترسی به اعضا همانند ماتریس بعلاوه بسیاری از نحوه دسترسی های دیگر. با دستور cbind() می توان ستون های دیگری نیز به قاب-داده اضافه نمود

```
Df=cbind(Df,c(.5,1.5,8.1,3.7))
Df
```

```
##   x      A      B c(0.5, 1.5, 8.1, 3.7)
## 1 3  TRUE Mohammad          0.5
## 2 4  TRUE      Ali          1.5
## 3 7 FALSE  Hassan          8.1
## 4 8 FALSE Hossein          3.7
```

قاب-داده را می توان با فراخوانی محیطی شبیه صفحه گسترده (مانند libreoffice در لینوکس ویا اکسل در ویندوز) از محیط R تعریف کرد.

```
Df2=edit(data.frame())
```



نکته اول: توجه شود با بزرگتر کردن صفحه، ستونهای بیشتری نمایان می شود.

نکته دوم: با کلیک بر روی نام ستون می توانید نام ستون را تغییر دهید.

یک نوع داده ای از نسل قاب داده بانام جدول-داده (Data.Table) تولید شده است که بسیار قوی و کاربردی است، چون مقصد این مباحث نیست از آن عبور می کنیم و اگر عمر و توفیق بود در آینده در مقاله هایی دیگر به آن خواهیم پرداخت، انشاءاله [م]

یک نوع داده دیگر بنام لیست (List) وجود دارد که می توان گفت نوع توسعه یافته قاب-داده است، لذا هر عضو آن می تواند خود یک لیست دیگر و یا قاب داده باشد

### ۳- دسترسی به داده های موجود

توجه شما را به این نکته معطوف می کنم که در قسمت قبل وا خودمان داده را ایجاد کردیم اما در این قسمت می خواهیم به داده هایی که از قبل ایجاد شده اندمانند فایل ها و یا پایگاه های داده دسترسی پیدا کنیم.

#### الف- خواندن از و نوشتن در فایل

اغلب اوقات نیاز است که بتوان اطلاعات را از فایل های ذخیره شده بخوانیم. ساده ترین راه خواندن فایل های متنی و همینطور وشتن در آنها می باشد. یکی از تابعهای تعریف شده برای اینکار read.table() می باشد. با فرض اینکه پنج خط اول این تابع بدین شرح باشد:



ردیف, نام, کد, سمت  
 ۱, "محمد محمدی", 110020, کارشناس  
 ۲, "علی اعلایی", 110021, "کارشناس ارشد"  
 ۳, "حسن حسینی", 110022, مدیر  
 ۴, "حسین محسنی", 110023, "مدیر مسئول"

و فایل مورد نظر در پوشه File در کنار نرم افزار باشد.

```
Personel=read.table("./File/Personel.txt",header = TRUE,sep = ",",quote = '"')
print(Personel)
```

ردیف	نام	کد	سمت
1	محمد محمدی	110020	کارشناس
2	علی اعلایی	110021	کارشناس ارشد
3	حسن حسینی	110022	مدیر
4	حسین محسنی	110023	مدیر مسئول
5	سجاد ساجدی	110024	معاونت
6	باقر علمایی	110025	مسئول دفتر
7	صادق صداقت زاده	110026	رییس دفتر
8	کاظم رضایی	110027	مدیر عامل

header=به تابع می گوید که خط اول فایل را بعنوان نام ستون ها در نظر بگیرد.  
 sep=در واقع مختصر نام separator به معنی جداکننده فیلدها می باشد.  
 quote=یکی از مشکلاتی که در خواندن فایل های متنی وجود دارد استفاده از علامت و مخصصا فاصله در فیلد می باشد که غالب تابع ها و دستورات خواندن فایل های متنی، فاصله را بعنوان جداکننده ستون در نظر می گیرند. جهت جلوگیری از این اتفاق علاوه بر مشخص کردن جداکننده توسط عامل بالایی، لازم است که به تابع بگوییم که هرگاه علامتی خاص را (که بطور غالب علامت می باشد را) بعنوان ابتدا و آغاز یک(تاکید می کنم یک) ستون در نظر بگیرد.

دستور write.table() جهت نوشتن یک قاب-داده در فایل می باشد.

```
('"' = write.table(Df,"/Path",sep=" ",quote
```

## ب-مجموعه داده های ساخته شده در R

همانطور که قبلا صحبت شد در نرم افزار R تعداد زیادی مجموعه داده های خوب و معتبر وجود دارد که توسعه دهندگان جهت تمرین و یادگیری تابعهای نرم افزار آنها را ایجاد کرده اند. در این قسمت به نمایش یکی از آنها و همینطور نحوه جدیدی از دسترسی به اعضای قاب-داده اشاره می کنیم. توجه شود که هریک از مجموعه داده ها را که فراخوانی کنید آنها بصورت یک قاب-داده آماده می کند

```
data("trees")
trees
```

```
##      Girth Height Volume
## 1      8.3      70    10.3
## 2      8.6      65    10.3
## 3      8.8      63    10.2
## 4     10.5      72    16.4
## 5     10.7      81    18.8
## 6     10.8      83    19.7
## 7     11.0      66    15.6
## 8     11.0      75    18.2
## 9     11.1      80    22.6
## 10    11.2      75    19.9
## 11    11.3      79    24.2
## 12    11.4      76    21.0
## 13    11.4      76    21.4
## 14    11.7      69    21.3
## 15    12.0      75    19.1
## 16    12.9      74    22.2
## 17    12.9      85    33.8
## 18    13.3      86    27.4
## 19    13.7      71    25.7
## 20    13.8      64    24.9
## 21    14.0      78    34.5
## 22    14.2      80    31.7
## 23    14.5      74    36.3
## 24    16.0      72    38.3
## 25    16.3      77    42.6
## 26    17.3      81    55.4
## 27    17.5      82    55.7
## 28    17.9      80    58.3
## 29    18.0      80    51.5
## 30    18.0      80    51.0
## 31    20.6      87    77.0
```

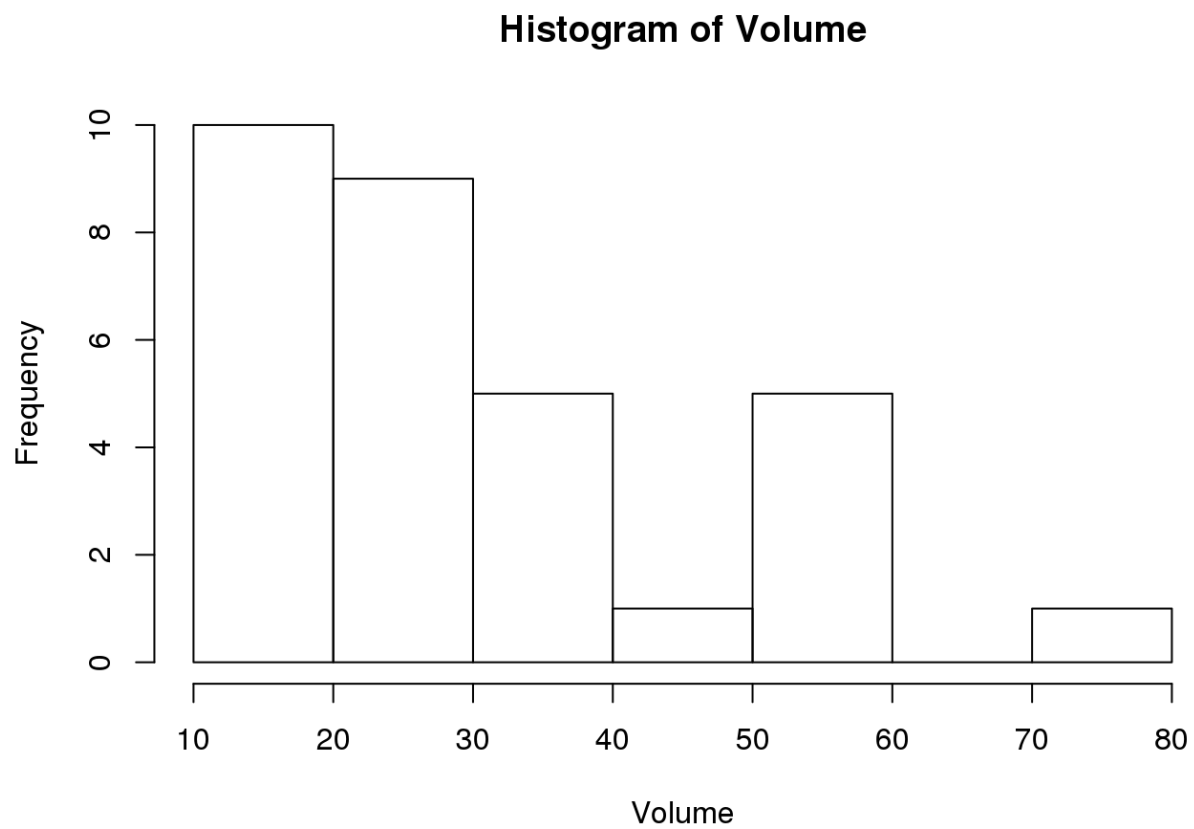
```
trees$Girth
```

```
## [1] 8.3 8.6 8.8 10.5 10.7 10.8 11.0 11.0 11.1 11.2 11.3 11.4 11.4 11.7
## [15] 12.0 12.9 12.9 13.3 13.7 13.8 14.0 14.2 14.5 16.0 16.3 17.3 17.5 17.9
## [29] 18.0 18.0 20.6
```

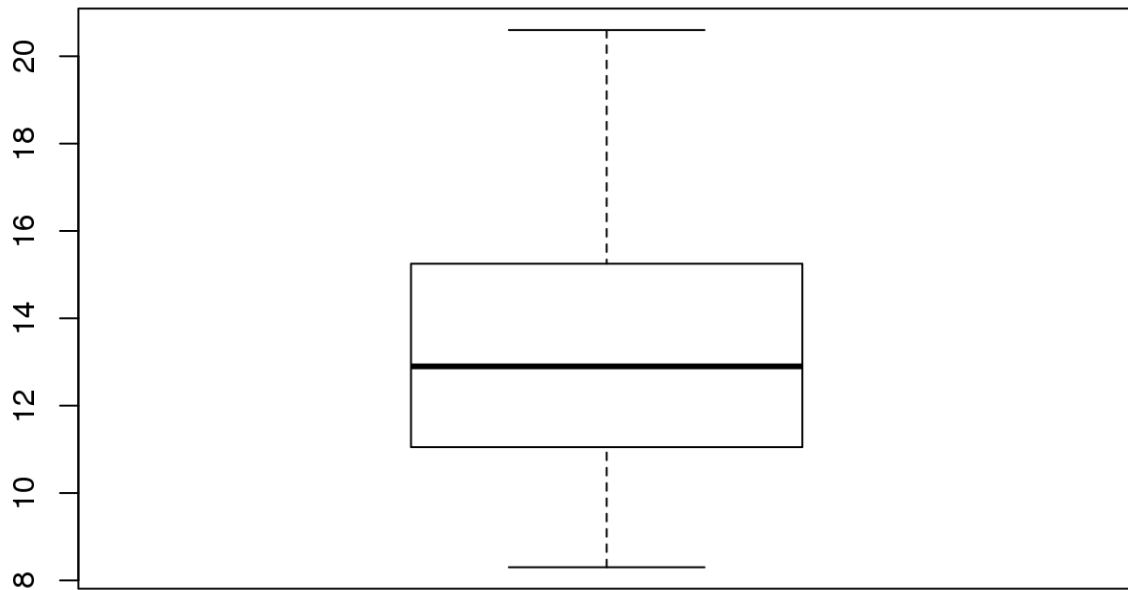
## ۴-نمایش هندسی

یکی از بهترین توصیفات در مورد R ، امکان و انعطاف بسیار بالای آن در نمایش هندسی داده هاست. نمودارهای زیر که از همان مجموعه داده درخت ها می باشد را ملاحظه نمایید.

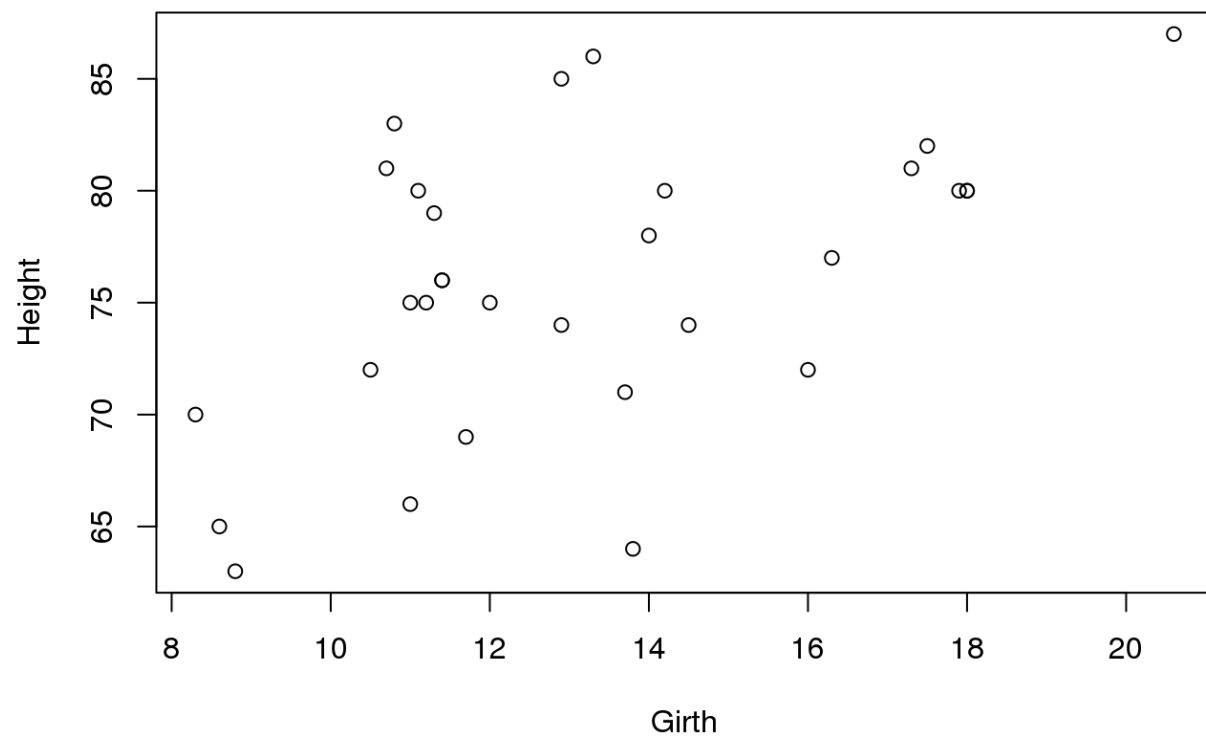
```
attach(trees)
hist(Volume)
```



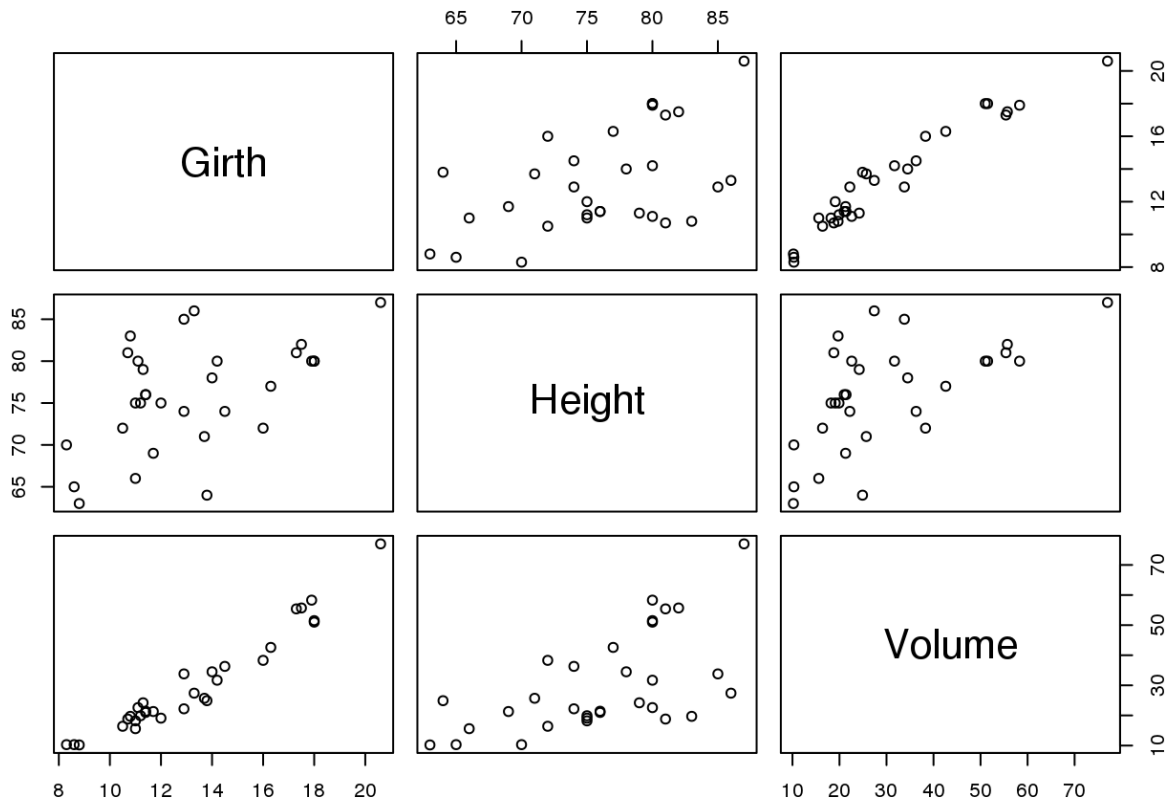
```
boxplot(Girth)
```



```
plot(Girth, Height)
```



```
plot(trees)
```



```
detach(trees)
```

## ۵- خلاصه آماری

چندین تابع بسیار مناسب برای جهت بررسی نوع و خلاصه آماری اشیاء (که بطور معمول مجموعه داده ها می باشند) وجود دارد.

```
class(trees)
```

```
## [1] "data.frame"
```

```
str(trees)
```

```
## 'data.frame':   31 obs. of  3 variables:
## $ Girth : num  8.3 8.6 8.8 10.5 10.7 10.8 11 11 11.1 11.2 ...
## $ Height: num  70 65 63 72 81 83 66 75 80 75 ...
## $ Volume: num  10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 ...
```

```
summary(trees)
```

```
##      Girth      Height      Volume
## Min.   : 8.30   Min.   :63   Min.   :10.20
## 1st Qu.:11.05   1st Qu.:72   1st Qu.:19.40
## Median :12.90   Median :76   Median :24.20
## Mean   :13.25   Mean   :76   Mean   :30.17
## 3rd Qu.:15.25   3rd Qu.:80   3rd Qu.:37.30
## Max.   :20.60   Max.   :87   Max.   :77.00
```

```
summary(trees$Girth)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
##      8.30  11.05   12.90   13.25   15.25   20.60
```

```
c(var(trees$Girth), mean(trees$Girth), min(trees$Girth), max(trees$Girth))
```

```
## [1]  9.847914 13.248387  8.300000 20.600000
```

## ۶- مثال جامع از مباحث بالا

خلاصه ای در مورد داده مورد بحث: سازمان هواشناسی می خواست اطلاعات آماری جاری شدن سیل را با توجه به سطح بارش روزانه در هر شهر را جمع آوری کند. اطلاعات زیر مربوط به یکی از شهرها از سال ۱۳۲۰ الی ۱۳۴۳ می باشد.

```
1 2.76 1.12 1.48 2.86 2.51 1.95 2.14 3.14 2.29 2.94 2.07 2.58 2.23 1.88
                                     1.12 .48
                                     2.12 2.29 4.69 2.12 3.48 2.99 1.50 2.76
```

در اینجا مراحل مروری کوتاه را بر آن باهم مرور می کنیم.

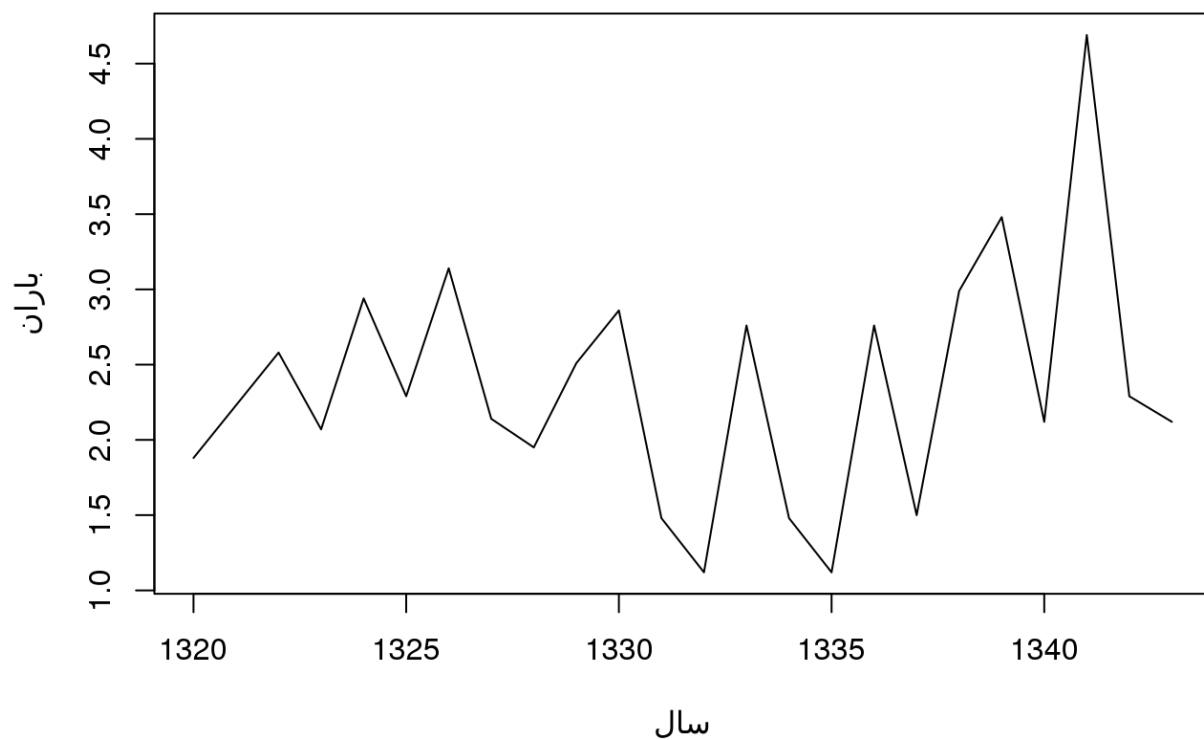
```
باران = c(1.88, 2.23, 2.58, 2.07, 2.94, 2.29, 3.14, 2.14, 1.95, 2.51, 2.86, 1
.48, 1.12, 2.76, 1.48, 1.12, 2.76, 1.50, 2.99, 3.48, 2.12, 4.69, 2.29, 2.12)
سال=1320:1343
اطلاعات=data.frame(باران, سال)
summary(اطلاعات)
```

```
##      سال
## Min.   :1320   Min.   :1.120
## 1st Qu.:1326   1st Qu.:1.933
## Median :1332   Median :2.260
## Mean   :1332   Mean    :2.354
## 3rd Qu.:1337   3rd Qu.:2.785
## Max.   :1343   Max.    :4.690
```

```
sd(اطلاعات$باران)
```

```
## [1] 0.7994994
```

```
plot(اطلاعات,type="l")
```



```
hist(اطلاعات$باران)
```



Histogram of اطلاعات\$باران

