A Brief Introduction to R

```
> 1+1
[1] 2
> 2^3 # Two to the power 3
[1] 8

> 1:30
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
[26] 26 27 28 29 30

> gamma(.5)^2      # Gamma(1/2) = Sqrt(Pi)
[1] 3.141593

> x <- 1           # Assigns the value 1 to x
> y <- 2
> x+y
[1] 3
> z <- x+y
> z
[1] 3
> x <- c(1,2,3,4,5,6)    #  Collect these numbers; x is now a vector

> z # No dynamic updating; it's not a spreadsheet
[1] 3
> x+y
[1] 3 4 5 6 7 8

> y = 1 + 2*x # Another way to do assignment
> cbind(x,y)
     x  y
[1,] 1  3
[2,] 2  5
[3,] 3  7
[4,] 4  9
[5,] 5 11
[6,] 6 13

> z <- y[x>4]           # z gets y such that x > 4
> z
[1] 11 13

> # If you put an array of integers inside the brackets, you get those
> # elements, in the order indicated.

> y[c(6,5,4,3,2,1)] # y in opposite order
[1] 13 11  9  7  5  3
> y[c(2,2,2,3,4)] # Repeats are okay
[1] 5 5 5 7 9
> y[7] # There is no seventh element.  NA is the missing value code
[1] NA
```

```
> # Computing probabilities, etc.
>
> pnorm(0) # Area less than zero for a standard normal
[1] 0.5
>
> pnorm(160,mean=100,sd=15)  # IQ of 160
[1] 0.9999683
>
> pcauchy(4)
[1] 0.9220209
>
> dnorm(0) # height of the curve
[1] 0.3989423
>
> dpois(0,lambda=3) # P(Y=0) for Y ~ Poisson(3)
[1] 0.04978707
>
> qnorm(0.975) # z value with P(Z<z) = 0.975
[1] 1.959964
>
> qf(0.975,df1=6,df2=122) # Critical value for F, not in any table
[1] 2.513606
>
> CriticalValue = qchisq(0.95,df=1:8)
> df=1:8; cbind(df,CriticalValue)
     df CriticalValue
[1,]  1      3.841459
[2,]  2      5.991465
[3,]  3      7.814728
[4,]  4      9.487729
[5,]  5     11.070498
[6,]  6     12.591587
[7,]  7     14.067140
[8,]  8     15.507313
```

```
> # Random number generation
> # Maybe transforming a uniform by inverse CDF
> help(Exponential) # Could also use  help(rexp)
```

Exponential {stats} R Documentation

The Exponential Distribution

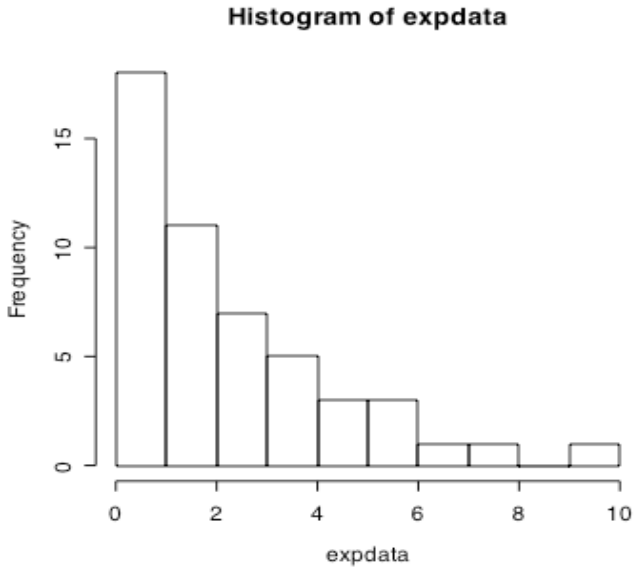
Description

Density, distribution function, quantile function and random generation for the exponential distribution with rate `rate` (i.e., mean `1/rate`).

Usage

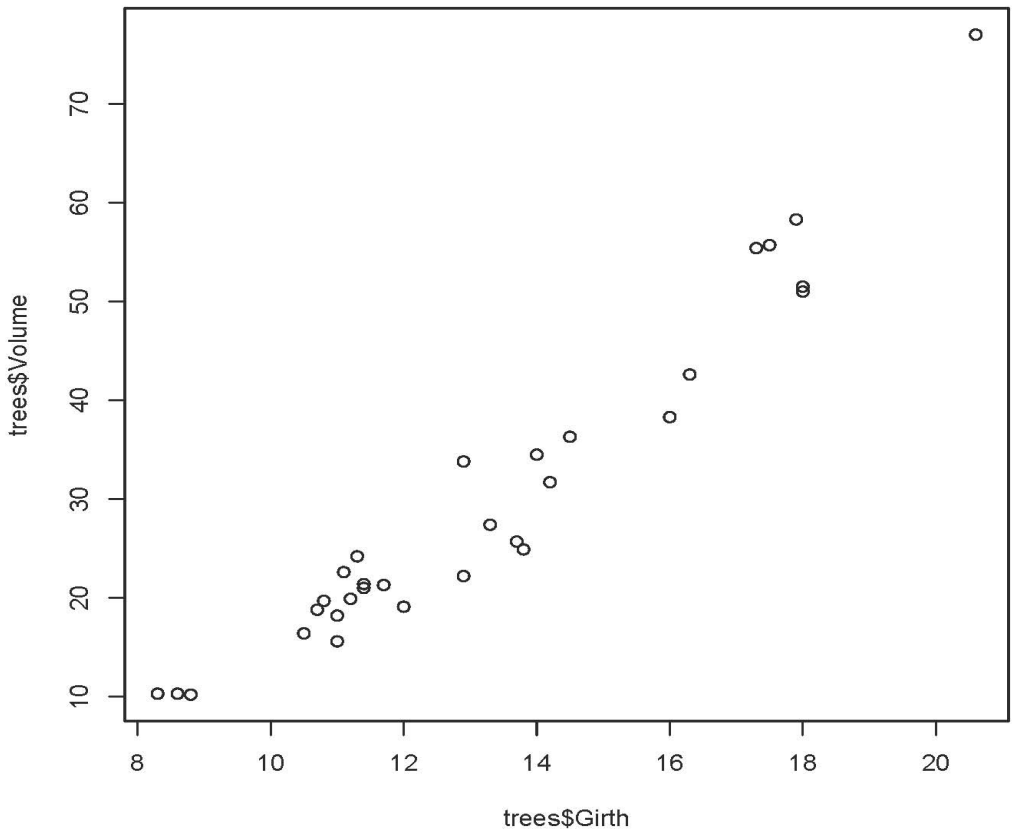
```
dexp(x, rate = 1, log = FALSE)
pexp(q, rate = 1, lower.tail = TRUE, log.p = FALSE)
qexp(p, rate = 1, lower.tail = TRUE, log.p = FALSE)
rexp(n, rate = 1)
```

```
> expdata = rexp(50,rate=1/2) # Random sample from exponential distribution, mean=theta=2
> expdata
 [1] 0.4330015 5.7893762 0.9803759 0.7172530 2.2696433 4.0045302 3.3989651 0.3104736
 [9] 1.2026790 0.8543951 1.0438012 5.5095891 0.7587579 1.9263300 6.0660176 9.3017992
[17] 1.0910204 0.6551285 1.5747176 5.9417700 0.8464761 7.6684436 0.1107589 1.6787699
[25] 2.4744338 3.3470232 0.3209082 4.4307811 4.5510434 1.4316870 0.3457547 0.1302476
[33] 0.5777305 1.0898631 1.4467458 3.2472808 1.8113195 0.5090032 2.4633656 0.8972205
[41] 0.7562905 2.4623634 0.3413955 2.3122374 0.4166320 2.6279765 1.5072294 3.5732947
[49] 3.5449348 2.6472542
> mean(expdata) # The MLE
[1] 2.267962
> hist(expdata)
```

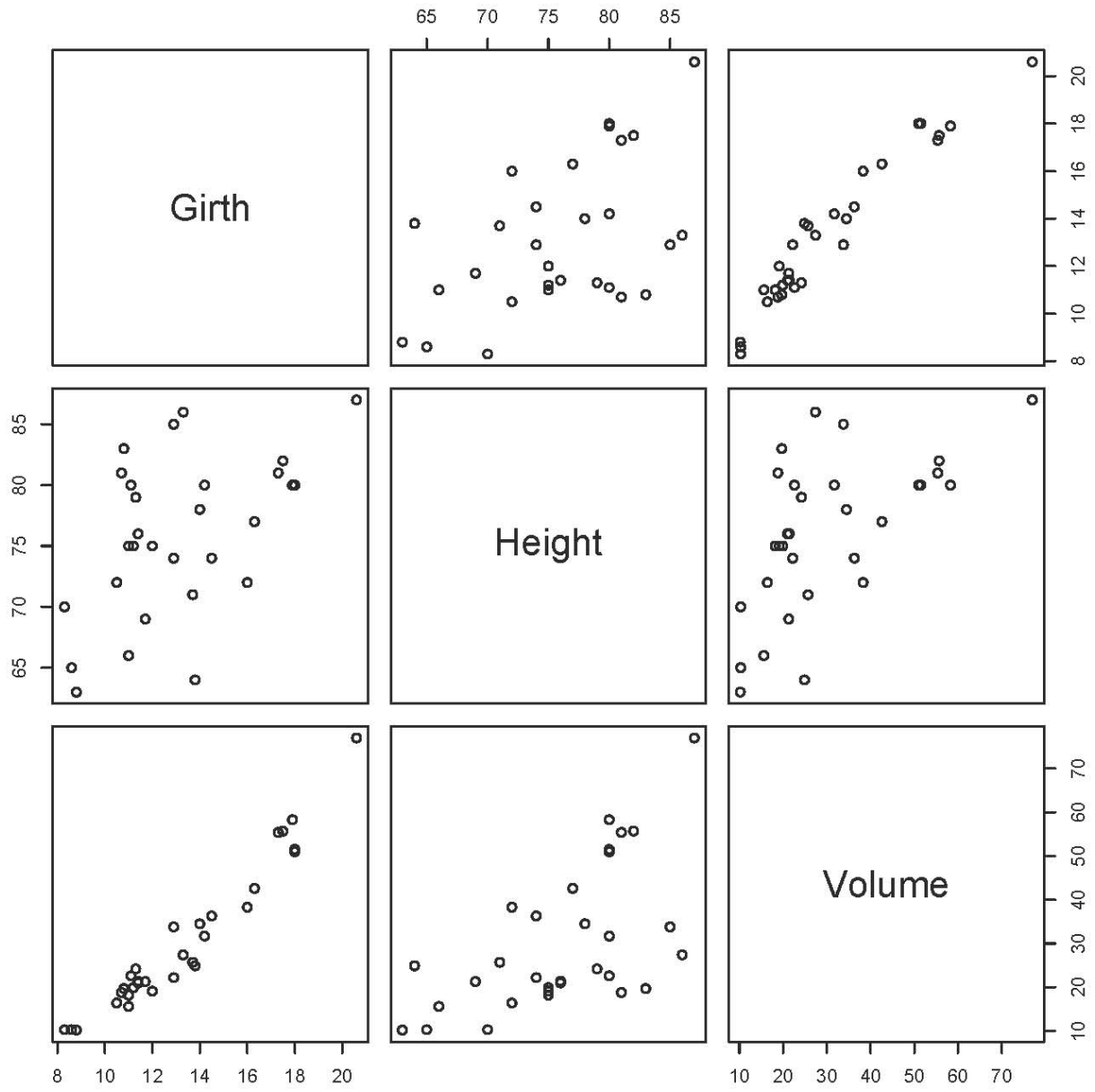


```
> # data() lists available data sets
> trees
   Girth Height Volume
1    8.3     70   10.3
2    8.6     65   10.3
3    8.8     63   10.2
4   10.5     72   16.4
5   10.7     81   18.8
6   10.8     83   19.7
7   11.0     66   15.6
8   11.0     75   18.2
9   11.1     80   22.6
10  11.2     75   19.9
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
> mean(trees$Height); var(trees$Height)
[1] 76
[1] 40.6
> mean(Height)
Error in mean(Height) : object 'Height' not found
> attach(trees)
> mean(Height)
[1] 76
```

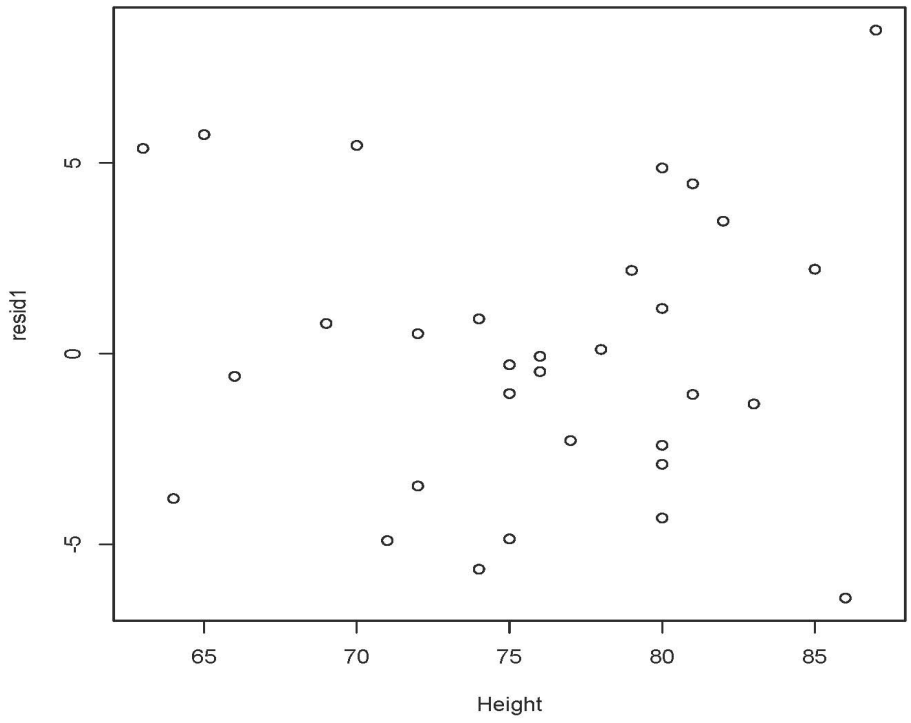
```
> summary(trees)
     Girth           Height       Volume     
 Min.   : 8.30   Min.   :63   Min.   :10.20  
 1st Qu.:11.05   1st Qu.:72   1st Qu.:19.40  
 Median :12.90   Median :76   Median :24.20  
 Mean   :13.25   Mean   :76   Mean   :30.17  
 3rd Qu.:15.25   3rd Qu.:80   3rd Qu.:37.30  
 Max.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

> cor(trees)
           Girth    Height    Volume
Girth  1.0000000 0.5192801 0.9671194
Height 0.5192801 1.0000000 0.5982497
Volume 0.9671194 0.5982497 1.0000000
> plot(trees$Girth,trees$Volume)
```



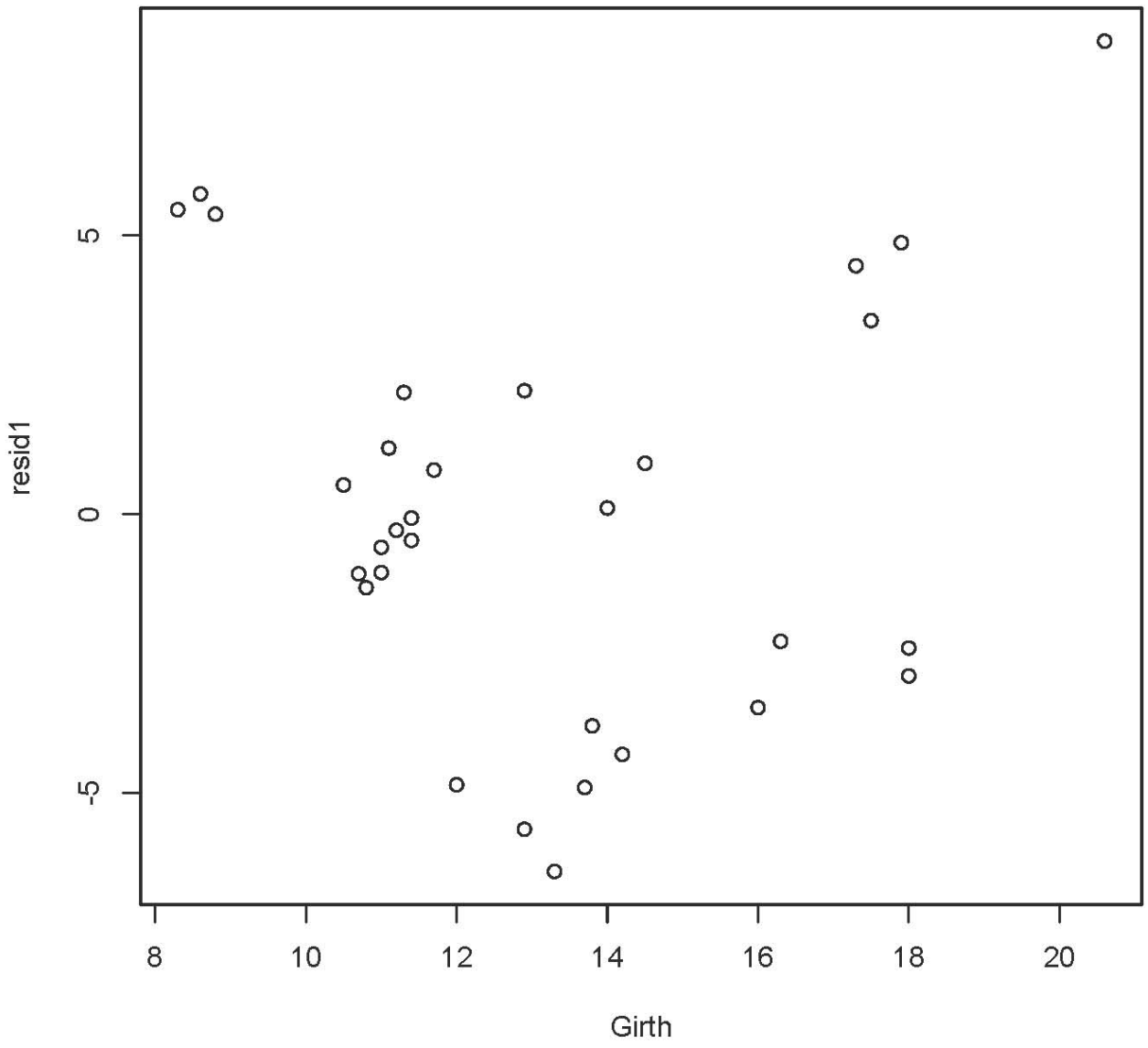
```
> pairs(trees) # Can also give it a matrix, in which the columns are
variables
```



```
> # Linear regression
> treemodel1 = lm(Volume ~ Girth + Height)
> # Could also do    treemodel1 = lm(Volume ~ Girth + Height,data=trees)
> summary(treemodel1)

Call:
lm(formula = Volume ~ Girth + Height)

Residuals:
    Min      1Q  Median      3Q     Max
-6.4065 -2.6493 -0.2876  2.2003  8.4847

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -57.9877     8.6382  -6.713 2.75e-07 ***
Girth         4.7082     0.2643  17.816  < 2e-16 ***
Height        0.3393     0.1302   2.607   0.0145 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.882 on 28 degrees of freedom
Multiple R-squared: 0.948,    Adjusted R-squared: 0.9442
F-statistic:   255 on 2 and 28 DF,  p-value: < 2.2e-16

> resid1 = treemodel1$residuals; mean(resid1)
[1] 1.504453e-16
> plot(Height,resid1)
```



```
> plot(Girth,resid1)
```




```
> girthsq = Girth^2
> treemodel2 = lm(Volume ~ Girth + Height + girthsq)
> summary(treemodel2)
```

Going further on your own

Look at the free mini-book *An introduction to R* by Venables et al. A copy is available at http://www.utstat.toronto.edu/~brunner/help/R-intro.pdf. Or, try

```
> library(help="stats")
```

```
AIC                     Akaike's An Information Criterion
ARMAacf                 Compute Theoretical ACF for an ARMA Process
ARMAtoMA                Convert ARMA Process to Infinite MA Process
Beta                    The Beta Distribution
Binomial                The Binomial Distribution
Box.test                Box-Pierce and Ljung-Box Tests
C                       Sets Contrasts for a Factor
Cauchy                  The Cauchy Distribution
Chisquare               The (non-central) Chi-Squared Distribution
Exponential             The Exponential Distribution
FDist                   The F Distribution
GammaDist               The Gamma Distribution
Geometric               The Geometric Distribution
HoltWinters             Holt-Winters Filtering
```

```
. . . list continues!
```

To see details on one of these packages, type something like

```
> help(t.test)
```

Generally, R's help assumes knowledge of the topic, but they often give references. For a quicker introduction to many statistical topics, try the Wikipedia:

http://en.wikipedia.org

Or you could ask me, but I might have to look in the Wikipedia.
