

# Package ‘UsingR’

January 2, 2012

**Version** 0.1-17

**Title** Data sets for the text ‘‘Using R for Introductory Statistics’’

**Author** John Verzani <verzani@math.csi.cuny.edu>

**Maintainer** John Verzani <verzani@math.csi.cuny.edu>

**Description** A collection of datasets to accompany the textbook ‘‘Using R for Introductory Statistics.’’

**Depends** MASS

**Suggests** zoo

**License** GPL (>= 2)

**URL** <http://www.math.csi.cuny.edu/UsingR>

**LazyData** TRUE

**Repository** CRAN

**Date/Publication** 2011-09-22 05:32:40

## R topics documented:

age.universe . . . . .	4
aid . . . . .	5
alaska.pipeline . . . . .	6
alltime.movies . . . . .	7
aosat . . . . .	7
arctic.oscillations . . . . .	8
babies . . . . .	9
babyboom . . . . .	10
batting . . . . .	11
baycheck . . . . .	12
best.times . . . . .	13
blood . . . . .	14
breakdown . . . . .	15

bright.stars	15
brightness	16
bumpers	17
BushApproval	17
bycatch	18
cabinet	19
camp	20
cancer	21
carbon	22
carsafety	22
central.park	23
central.park.cloud	24
cfb	25
chicken	26
chips	26
co2emiss	27
coins	28
coldvermont	28
corn	29
crime	30
deflection	30
DensityPlot	31
diamond	32
divorce	33
do	34
DOTplot	35
dottodot	35
dowdata	36
dvdsales	37
emissions	37
ewr	38
exec.pay	39
fat	39
father.son	41
female.inc	42
firstchi	43
five.yr.temperature	43
florida	44
galileo	45
galton	46
gap	47
gasprices	47
getAnswer	48
goalspergame	49
google	49
grades	50
grip	50
hall.fame	51

healthy . . . . .	52
heartrate . . . . .	53
home . . . . .	54
homedata . . . . .	54
homeprice . . . . .	55
homework . . . . .	56
HUMMER . . . . .	56
iq . . . . .	57
kid.weights . . . . .	58
KSI . . . . .	59
last.tie . . . . .	59
lawsuits . . . . .	60
malpract . . . . .	61
mandms . . . . .	61
math . . . . .	62
maydow . . . . .	63
midsize . . . . .	64
MLBattend . . . . .	64
movies . . . . .	65
mw.ages . . . . .	66
nba.draft . . . . .	67
normtemp . . . . .	67
npdb . . . . .	68
nym.2002 . . . . .	70
OBP . . . . .	71
oral.lesion . . . . .	71
ozonemonthly . . . . .	72
paradise . . . . .	73
pi2000 . . . . .	73
primes . . . . .	74
puerto . . . . .	74
QQplot . . . . .	75
rat . . . . .	76
reaction.time . . . . .	77
reddrum . . . . .	78
salmon.rate . . . . .	78
salmonharvest . . . . .	79
samhda . . . . .	80
scatter.with.hist . . . . .	81
scrabble . . . . .	82
sim . . . . .	83
simple.chutes . . . . .	83
simple.densityplot . . . . .	84
simple.eda . . . . .	85
simple.eda.ts . . . . .	86
simple.fancy.stripchart . . . . .	87
simple.freqpoly . . . . .	88
simple.hist.and.boxplot . . . . .	88

simple.lag . . . . .	89
simple.lm . . . . .	90
simple.median.test . . . . .	91
simple.scatterplot . . . . .	92
simple.sim . . . . .	93
simple.violinplot . . . . .	94
simple.z.test . . . . .	95
skateranks . . . . .	96
slc . . . . .	97
smokyph . . . . .	97
south . . . . .	98
southernosc . . . . .	98
sp500.excess . . . . .	99
Split.zoo . . . . .	100
squareplot . . . . .	101
stud.recs . . . . .	102
student.expenses . . . . .	102
superbarplot . . . . .	103
tastesgreat . . . . .	104
tcm1y . . . . .	105
tempsalinity . . . . .	106
too.young . . . . .	107
twins . . . . .	107
u2 . . . . .	108
urchin.growth . . . . .	109
UsingR . . . . .	109
vacation . . . . .	110
violinplot . . . . .	111
watertemp . . . . .	112
yahoo.get.hist.quote . . . . .	112
yellowfin . . . . .	113

**Index****115**


---

age.universe	<i>Best estimate of the age of the universe</i>
--------------	---

---

**Description**

For years people have tried to estimate the age of the universe. This data set collects a few estimates starting with lower bounds using estimates for the earth's age.

**Usage**

```
data(age.universe)
```

**Format**

A data frame with 15 observations on the following 4 variables.

**lower** a numeric vector

**upper** a numeric vector

**year** a numeric vector

**source** Short description of source

**Details**

In the last two decades estimates for the age of the universe have been greatly improved. As of 2003, the best guess is 13.7 billion years with a margin of error of 1 percent. This last estimate is found by WMAP using microwave background radiation. Previous estimates were also based on estimates of Hubble's constant, and dating of old stars.

**Source**

This data was collected from the following web sites: <http://www.cwru.edu/pubaff/univcomm/2003/1-03/kraussuniverse.htm>, <http://www.astro.ucla.edu/~wright/age.html>, <http://www.lhup.edu/~dsimanek/cutting/ageuniv.htm>, and [http://map.gsfc.nasa.gov/m\\_uni/uni\\_101age.html](http://map.gsfc.nasa.gov/m_uni/uni_101age.html).

**Examples**

```
data(age.universe)
n = nrow(age.universe)
x = 1:15; names(x) = age.universe$year
plot(x, age.universe$upper, ylim=c(0,20))
points(x, age.universe$lower)
with(age.universe, sapply(x, function(i) lines(c(i,i), c(lower[i], upper[i])))))
```

---

aid	<i>monthly payment for federal program</i>
-----	--

---

**Description**

monthly payment for federal program

**Usage**

```
data(aid)
```

**Format**

The format is: Named num [1:51] 57.2 253.5 114.2 68.2 199.6 ... - attr(\*, "names")= chr [1:51] "Alabama" "Alaska" "Arizona" "Arkansas" ...

**Source**

From Kitchen's Exploring Statistics

**Examples**

```
data(aid)
hist(aid)
```

---

alaska.pipeline

*Comparison of in-field and laboratory measurement of defects*

---

**Description**

The Alaska pipeline data consists of in-field ultrasonic measurements of the depths of defects in the Alaska pipeline. The depth of the defects were then re-measured in the laboratory. These measurements were performed in six different batches.

**Usage**

```
data(alaska.pipeline)
```

**Format**

A data frame with 107 observations on the following 3 variables.

**field.defect** Depth of defect as measured in field

**lab.defect** Depth of defect as measured in lab

**batch** One of 6 batches

**Source**

From an example in *Engineering Statistics Handbook* from <http://www.itl.nist.gov/div898/handbook/>

**Examples**

```
data(alaska.pipeline)
res = lm(lab.defect ~ field.defect, alaska.pipeline)
plot(lab.defect ~ field.defect, alaska.pipeline)
abline(res)
plot(fitted(res), resid(res))
```

---

alltime.movies	<i>Top movies of all time</i>
----------------	-------------------------------

---

**Description**

The top 79 all-time movies as of 2003 by domestic (US) gross receipts.

**Usage**

```
data(alltime.movies)
```

**Format**

A data frame with 79 observations on the following 2 variables.

**Gross** a numeric vector

**Release.Year** a numeric vector

The row names are the titles of the movies.

**Source**

This data was found on <http://movieweb.com/movie/alltime.html> on June 17, 2003. The source of the data is attributed to (partially) Exhibitor Relations Co. .

**Examples**

```
data(alltime.movies)
hist(alltime.movies$Gross)
```

---

aosat	<i>Artic Oscillation data based on SAT data</i>
-------	---

---

**Description**

A time series of January, February, and March measurements of the annular modes from January 1851 to March 1997.

**Usage**

```
data(aosat)
```

**Format**

The format is: first column is date in years with fraction to indicate month. The second column is the measurement.

**Details**

See <http://jisao.washington.edu/ao/> for more details on the importance of this time series.

**Source**

This data came from the file AO\\_SATindex\\_JFM\\_Jan1851March1997.ascii at [http://www.atmos.colostate.edu/ao/Data/ao\\_index.html](http://www.atmos.colostate.edu/ao/Data/ao_index.html)

**Examples**

```
data(aosat)
## Not run:
library(zoo)
z = zoo(aosat[,2], order.by=aosat[,1])
plot(z)
## yearly
plot(aggregate(z, floor(index(z)), mean))
## decade-long means
plot(aggregate(z, 10*floor(index(z)/10), mean))

## End(Not run)
```

---

arctic.oscillations     *Measurement of sea-level pressure at the arctic*

---

**Description**

A monthly time series from January 1899 to June 2002 of sea-level pressure measurements relative to some baseline.

**Usage**

```
data(arctic.oscillations)
```

**Format**

The format is: chr "arctic.oscillations"

**Details**

See <http://jisao.washington.edu/ao/> for more details on the importance of this time series.

**Source**

The data came from the file AO\\_TREN\\_NCEP\\_Jan1899Current.ascii found at [http://www.atmos.colostate.edu/ao/Data/ao\\_index.html](http://www.atmos.colostate.edu/ao/Data/ao_index.html).

**Examples**

```
data(arctic.oscillations)
x = ts(arctic.oscillations, start=c(1899,1), frequency=12)
plot(x)
```

babies

*Mothers and their babies data***Description**

A collection of variables taken for each new mother in a Child and Health Development Study.

**Usage**

```
data(babies)
```

**Format**

A data frame with 1,236 observations on the following 23 variables.

Variables in data file

**id** identification number

**plurality** 5= single fetus

**outcome** 1= live birth that survived at least 28 days

**date** birth date where 1096=January1,1961

**gestation** length of gestation in days

**sex** infant's sex 1=male 2=female 9=unknown

**wt** birth weight in ounces (999 unknown)

**parity** total number of previous pregnancies including fetal deaths and still births, 99=unknown

**race** mother's race 0=white 6=mex 7=black 8=asian 9=mixed 99=unknown

**age** mother's age in years at termination of pregnancy, 99=unknown

**ed** mother's education 0= less than 8th grade, 1 = 8th -12th grade - did not graduate, 2= HS graduate-no other schooling , 3= HS+trade, 4=HS+some college 5= College graduate, 6&7 Trade school HS unclear, 9=unknown

**ht** mother's height in inches to the last completed inch 99=unknown

**wt1** mother prepregnancy wt in pounds, 999=unknown

**drace** father's race, coding same as mother's race.

**dage** father's age, coding same as mother's age.

**ded** father's education, coding same as mother's education.

**dht** father's height, coding same as for mother's height

**dwt** father's weight coding same as for mother's weight

**marital** 1=married, 2= legally separated, 3= divorced, 4=widowed, 5=never married

**inc** family yearly income in \\$/2500 increments 0 = under 2500, 1=2500-4999, ..., 8= 12,500-14,999, 9=15000+, 98=unknown, 99=not asked

**smoke** does mother smoke? 0=never, 1= smokes now, 2=until current pregnancy, 3=once did, not now, 9=unknown

**time** If mother quit, how long ago? 0=never smoked, 1=still smokes, 2=during current preg, 3=within 1 yr, 4= 1 to 2 years ago, 5= 2 to 3 yr ago, 6= 3 to 4 yrs ago, 7=5 to 9yrs ago, 8=10+yrs ago, 9=quit and don't know, 98=unknown, 99=not asked

**number** number of cigs smoked per day for past and current smokers 0=never, 1=1-4,2=5-9, 3=10-14, 4=15-19, 5=20-29, 6=30-39, 7=40-60, 8=60+, 9=smoke but don't know,98=unknown, 99=not asked

### Source

This dataset is found from <http://www.stat.berkeley.edu/users/statlabs/labs.html>. It accompanies the excellent text *Stat Labs: Mathematical Statistics through Applications* Springer-Verlag (2001) by Deborah Nolan and Terry Speed.

### Examples

```
data(babies)
plot(wt ~ factor(smoke), data=babies)
plot(wt1 ~ dwt, data=babies, subset=wt1 < 800 & dwt < 800)
```

---

babyboom

*Babyboom: data for 44 babies born in one 24-hour period.*

---

### Description

The babyboom dataset contains the time of birth, sex, and birth weight for 44 babies born in one 24-hour period at a hospital in Brisbane, Australia.

### Usage

```
data(babyboom)
```

### Format

A data frame with 44 observations on the following 4 variables.

**clock.time** Time on clock

**gender** a factor with levels girl boy

**wt** weight in grams of child

**running.time** minutes after midnight of birth

**Source**

This data set was submitted to the *Journal of Statistical Education*, <http://www.amstat.org/publications/jse/secure/v7n3/datasets.dunn.cfm>, by Peter K. Dunn.

**Examples**

```
data(babyboom)
hist(babyboom$wt)
hist(diff(babyboom$running.time))
```

---

batting

*Batting statistics for 2002 baseball season*


---

**Description**

This dataset contains batting statistics for the 2002 baseball season. The data allows you to compute batting averages, on base percentages, and other statistics of interest to baseball fans. The data only contains players with more than 100 atbats for a team in the year. The data is excerpted with permission from the Lahman baseball database at <http://www.baseball11.com/>.

**Usage**

```
data(batting)
```

**Format**

A data frame with 438 observations on the following 22 variables.

**playerID** This is coded, but those familiar with the players should be able to find their favorites.

**yearID** a numeric vector. Always 2002 in this dataset.

**stintID** a numeric vector. Player's stint (order of appearances within a season)

**teamID** a factor with Team

**lgID** a factor with levels AL NL

**G** number of games played

**AB** number of at bats

**R** number of runs

**H** number of hits

**DOUBLE** number of doubles. "2B" in original data base.

**TRIPLE** number of triples. "3B" in original data base

**HR** number of home runs

**RBI** number of runs batted in

**SB** number of stolen bases

**CS** number of times caught stealing

**BB** number of base on balls (walks)  
**SO** number of strikeouts  
**IBB** number of intentional walks  
**HBP** number of hit by pitches  
**SH** number of sacrifice hits  
**SF** number of sacrifice flies  
**GIDP** number of grounded into double plays

### Details

Baseball fans are “statistics” crazy. They love to talk about things like RBIs, BAs and OBPs. In order to do so, they need the numbers. This data comes from the Lahman baseball database at <http://www.baseball1.com/>. The complete dataset includes data for all of baseball not just the year 2002 presented here.

### Source

Lahman baseball database, <http://www.baseball1.com/>

### References

In addition to the data set above, the book *Curve Ball*, by Albert, J. and Bennett, J., Copernicus Books, gives an extensive statistical analysis of baseball.

See <http://www.baseball-almanac.com/stats.shtml> for definitions of common baseball statistics.

### Examples

```
data(batting)
attach(batting)
BA = H/AB # batting average
OBP = (H + BB + HBP) / (AB + BB + HBP + SF) # On base "percentage"
```

---

baycheck

*Population estimate of type of Bay Checkerspot butterfly*

---

### Description

Estimates of the population of a type of Bay Checkerspot butterfly near San Francisco.

### Usage

```
data(baycheck)
```

**Format**

A data frame with 27 observations on the following 2 variables.

**year** a numeric vector

**Nt** estimated number

**Source**

From chapter 4 of Morris and Doak, *Quantitative Conservation Biology: Theory and Practice of Population Viability Analysis*, Sinauer Associates, 2003.

**Examples**

```
data(baycheck)
plot(Nt ~ year, baycheck)
## fit Ricker model  $N_{t+1} = N_t e^{-rt} W_t$ 
n = length(baycheck$year)
yt = with(baycheck, log(Nt[-1]/Nt[-n]))
nt = with(baycheck, Nt[-n])
lm(yt ~ nt, baycheck)
```

---

best.times

*Best track and field times by age and distance*

---

**Description**

A dataset giving world records in track and field running events for various distances and different age groups.

**Usage**

```
data(best.times)
```

**Format**

A data frame with 113 observations on the following 6 variables.

**Dist** Distance in meters (42195 is a marathon)

**Name** Name of record holder

**Date** Date of record

**Time** Time in seconds

**Time.1** Time as character

**age** Age at time of record

**Details**

Age-graded race results allow competitors of different ages to compare their race performances. This data set allows one to see what the relationship is based on peak performances.

**Source**

The data comes from <http://www.personal.rdg.ac.uk/~snsgrubb/athletics/agegroups.html> which includes a calculator to compare results.

**Examples**

```
data(best.times)
attach(best.times)
by.dist = split(best.times,as.factor(Dist))
lm(scale(Time) ~ age, by.dist[['400']])
dists = names(by.dist)
apply(dists, function(n) print(lm(scale(Time) ~ age, by.dist[[n]])))
```

---

blood

*blood pressure readings*

---

**Description**

blood pressure of 15 males taken by machine and expert

**Usage**

```
data(blood)
```

**Format**

This data frame contains the following columns:

**Machine** a numeric vector

**Expert** a numeric vector

**Source**

Taken from Kitchen's Exploring Statistics.

**References**

~~ possibly secondary sources and usages ~~

**Examples**

```
data(blood)
attach(blood)
t.test(Machine,Expert)
detach(blood)
```

---

breakdown	<i>Time of insulating fluid to breakdown</i>
-----------	--

---

**Description**

The time in minutes for an insulating fluid to break down under varying voltage loads

**Usage**

```
data(breakdown)
```

**Format**

A data frame with 75 observations on the following 2 variables.

**voltage** Number of kV

**time** time in minutes

**Details**

An example from industry where a linear model is used with replication and transformation of variables.

**Source**

Data is from Display 8.3 of Ramsay and Shafer, *The Statistical Sleuth* Duxbury Press, 1997.

**Examples**

```
data(breakdown)
plot(log(time) ~ voltage, data = breakdown)
```

---

bright.stars	<i>List of bright stars with Hipparcos catalog number</i>
--------------	---

---

**Description**

List of bright stars with Hipparcos catalog number.

**Usage**

```
data(bright.stars)
```

**Format**

A data frame with 96 observations on the following 2 variables.

**name** Common name of star

**hip** HIP number for identification

**Details**

The source of star names goes back to the Greeks and Arabs. Few are modern. This is a list of 96 common stars.

**Source**

Form the Hipparcos website <http://astro.estec.esa.nl/Hipparcos/ident6.html>.

**Examples**

```
data(bright.stars)
all.names = paste(bright.stars$name, sep="", collapse="")
x = unlist(strsplit(tolower(all.names), ""))
letter.dist = sapply(letters, function(i) sum(x == i))
data(scrabble) # for frequency info
p = scrabble$frequency[1:26];p=p/sum(p)
chisq.test(letter.dist, p=p) # compare with English
```

---

brightness

*Brightness of 966 stars*

---

**Description**

The Hipparcos Catalogue has information on over 100,000 stars. Listed in this dataset are brightness measurements for 966 stars from a given sector of the sky.

**Usage**

```
data(brightness)
```

**Format**

A univariate dataset of 966 numbers.

**Details**

This is field H5 in the catalog measuring the magnitude, V , in the Johnson UBV photometric system. The smaller numbers are for brighter stars.

**Source**

<http://astro.estec.esa.nl/hipparcos>

**Examples**

```
data(brightness)
hist(brightness)
```

---

bumpers

*Bumper repair costs for various automobiles*

---

**Description**

bumper repair costs

**Usage**

```
data(bumpers)
```

**Format**

Price in dollars to repair a bumper.

**Source**

From *Exploring Statistics*, Duxbury Press, 1998, L. Kitchens.

**Examples**

```
data(bumpers)
stem(bumpers)
```

---

BushApproval

*U.S. President George Bush approval ratings*

---

**Description**

Approval ratings as reported by six different polls.

**Usage**

```
data(BushApproval)
```

**Format**

A data frame with 323 observations on the following 3 variables.

**date** The date poll was begun (some take a few days)

**approval** a numeric number between 0 and 100

**who** a factor with levels fox gallup newsweek time.cnn upenn zogby

## Details

A data set of approval ratings of George Bush over the time of his presidency, as reported by several agencies. Most polls were of size approximately 1,000 so the margin of error is about 3 percentage points.

## Source

This is from data found at <http://www.pollingreport.com/BushJob.htm>. The idea came from an article in *Salon* [http://salon.com/opinion/feature/2004/02/09/bush\\_approval/index.html](http://salon.com/opinion/feature/2004/02/09/bush_approval/index.html) by James K. Galbraith.

## Examples

```
data(BushApproval)
attach(BushApproval)

## Plot data with confidence intervals. Each poll gets different line type
## no points at first
plot(strptime(date,"%m/%d/%y"),approval,type="n",
     ylab = "Approval Rating",xlab="Date",
     ylim=c(30,100)
     )

## plot line for CI. Margin or error about 3
## matlines has trouble with dates from strptime()
colors = rainbow(6)

for(i in 1:nrow(BushApproval)) {
  lines(rep(strptime(date[i],"%m/%d/%y"),2),
        c(approval[i]-3,approval[i]+3),
        lty=as.numeric(who[i]),
        col=colors[as.numeric(who[i])])
}

## plot points
points(strptime(date,"%m/%d/%y"),approval,pch=as.numeric(who))

## add legend
legend((2003-1970)*365*24*60*60,90,legend=as.character(levels(who)),lty=1:6,col=1:6)
detach(BushApproval)
```

---

bycatch

*Number of Albatrosses accidentally caught during a fishing haul*

---

## Description

This data set from Hillborn and Mangel contains data on the number of Albatrosses accidentally caught while fishing by commercial fisheries.

**Usage**

```
data(bycatch)
```

**Format**

A data frame with 18 observations on the following 2 variables.

**no.albatross** The number of albatross caught

**no.hauls** Number of hauls with this many albatross caught

**Details**

During fishing operations non-target species are often captured. These are called “incidental catch”. In some cases, large-scale observer programs are used to monitor this incidental catch.

When fishing for squid, albatrosses are caught while feeding on the squid at the time of fishing. This feeding is encouraged while the net is being hauled in, as the squid are clustered making it an opportunistic time for the albatross to eat.

**Source**

This is from Hilborn and Mangel, *The Ecological Detective*, Princeton University Press, 1997. Original source of data is Bartle.

**Examples**

```
data(bycatch)
hauls = with(bycatch,rep(no.albatross,no.hauls))
```

---

cabinet

*Estimated tax savings for US President Bush's cabinet*

---

**Description**

Estimated savings from a repeal of the tax on capital gains and dividends for Bush's cabinet members.

**Usage**

```
data(cabinet)
```

**Format**

A data frame with 19 observations on the following 4 variables.

**name** Name of individual

**position** Position of individual

**est.dividend.cg** Estimated amount of dividend and capital gain income

**est.tax.savings** Estimated tax savings

**Details**

Quoting from the data source [http://www.house.gov/reform/min/pdfs\\_108/pdf\\_inves/pdf\\_admin\\_tax\\_law\\_cabinet\\_june\\_3\\_rep.pdf](http://www.house.gov/reform/min/pdfs_108/pdf_inves/pdf_admin_tax_law_cabinet_june_3_rep.pdf) (From Henry Waxman, congressional watchdog.)

“On May 22, 2003, the House of Representatives and the Senate passed tax legislation that included \$320 billion in tax cuts. The final tax cut bill was signed into law by President Bush on May 28, 2003. The largest component of the new tax law is the reduction of tax rates on both capital gains and dividend income. The law also includes the acceleration of future tax cuts, as well as new tax reductions for businesses.

This capital gains and dividend tax cut will have virtually no impact on the average American. The vast majority of Americans (88%) have no capital gains on their tax returns. These taxpayers will receive no tax savings at all from the reduction in taxes on capital gains. Similarly, most Americans (75%) have no dividends on their tax returns. The reduction of taxes on dividends.

While the average American will derive little, if any, benefit from the cuts in dividend and capital gains taxes, the law offers significant benefits to the wealthy. For example, the top 1% receive an average tax cut of almost \$21,000 each. In particular, some of the major beneficiaries of this plan will be Vice President Cheney, President Bush, and other members of the cabinet. Based on 2001 and 2002 dividends and capital gains income, Vice President Cheney, President Bush, and the cabinet are estimated to receive an average tax cut of at least \$42,000 per year. Their average tax savings equals the median household income in the United States.”

**Source**

[http://www.house.gov/reform/min/pdfs\\_108/pdf\\_inves/pdf\\_admin\\_tax\\_law\\_cabinet\\_june\\_3\\_rep.pdf](http://www.house.gov/reform/min/pdfs_108/pdf_inves/pdf_admin_tax_law_cabinet_june_3_rep.pdf)

**Examples**

```
data(cabinet)
attach(cabinet)
median(est.dividend.cg)
mean(est.dividend.cg)
detach(cabinet)
```

---

camp

*Mount Campito Yearly Treering Data, -3435-1969.*

---

**Description**

Contains annual tree-ring measurements from Mount Campito from 3426 BC through 1969 AD.

**Usage**

```
data(camp)
```

**Format**

A univariate time series with 5405 observations. The object is of class "ts".

**Details**

This series is a standard example for the concept of long memory time series.

The data was produced and assembled at the Tree Ring Laboratory at the University of Arizona, Tuscon.

**Source**

Time Series Data Library:<http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/>

**References**

This data set is in the tseries package. It is repackaged here for convenience only.

**Examples**

```
data(camp)
acf(camp)
```

---

cancer

*cancer survival times*

---

**Description**

cancer survival times

**Usage**

```
data(cancer)
```

**Format**

The format is: The format is: List of 5 numeric components stomach, bronchus, colon, ovary and breast

**Source**

Taken from L. Kitchens, *Exploring Statistics*, Duxbury Press, 1997.

**Examples**

```
data(cancer)
boxplot(cancer)
```

carbon

*Carbon Monoxide levels at different sites*

---

**Description**

Carbon Monoxide levels at different sites

**Usage**

```
data(carbon)
```

**Format**

This data frame contains the following columns:

**Monoxide** a numeric vector

**Site** a numeric vector

**Source**

Borrowed from Kitchen's Exploring Statistics

**Examples**

```
data(carbon)
boxplot(Monoxide ~ Site,data=carbon)
```

---

carsafety

*Fatality information in U.S. for several popular cars*

---

**Description**

Safety statistics appearing in a January 12th, 2004 issue of the *New Yorker* showing fatality rates per million vehicles both for drivers of a car, and drivers of other cars that are hit.

**Usage**

```
data(carsafety)
```

**Format**

A data frame with 33 observations on the following 4 variables.

**Make.model** The make and model of the car

**type** Type of car

**Driver.deaths** Number of drivers deaths per year if 1,000,000 cars were on the road

**Other.deaths** Number of deaths in other vehicle caused by accidents involving these cars per year if 1,000,000 cars were on the road

**Details**

The article this data came from wishes to make the case that SUVs are not safer despite a perception among the U.S. public that they are.

**Source**

From "Big and Bad" by Malcolm Gladwell. *New Yorker*, Jan. 12 2004 pp28-33. Data attributed to Tom Wenzel and Marc Ross who have written <http://www.lbl.gov/Science-Articles/Archive/assets/images/2002/Aug-26-2002/SUV-report.pdf>.

**Examples**

```
data(carsafety)
plot(Driver.deaths + Other.deaths ~ type, data = carsafety)
plot(Driver.deaths + Other.deaths ~ type, data = carsafety)
```

---

central.park

*Weather in Central Park NY in May 2003*

---

**Description**

A listing of various weather measurements made at Central Park in New York City during the month of May 2003.

**Usage**

```
data(central.park)
```

**Format**

A data frame with 31 observations on the following 19 variables.

**DY** the day

**MAX** maximum temperature (temperatures in Farenheit)

**MIN** minimum temperature

**AVG** average temperature

**DEP** departure from normal

**HDD** heating degree days

**CDD** cooling degree days

**WTR** Water fall. A factor as "T" is a trace.

**SNW** Amount of snowfall

**DPTH** Depth of snow

**SPD** Average wind speed

**SPD1** Max wind speed

**DIR** 2 minimum direction

**MIN2** Sunshine measurement a factor with two levels 0 M

**PSBL** Sunshine measurement a factor with levels 0 M

**S.S** Sunshine measurement. 0-3 = Clear, 4-7 partly cloudy, 8-10 is cloudy

**WX** (This is not as documented in the data source. Ignore this variable. It should be: 1 = FOG, 2 = FOG REDUCING VISIBILITY TO 1/4 MILE OR LESS, 3 = THUNDER, 4 = ICE PELLETS, 5 = HAIL, 6 = GLAZE OR RIME, 7 = BLOWING DUST OR SAND: VSBY 1/2 MILE OR LESS, 8 = SMOKE OR HAZE, 9 = BLOWING SNOW, X = TORNADO)

**SPD3** peak wind speed

**DR** direction of peak wind

### Details

This datasets summarizes the weather in New York City during the merry month of May 2003. This data set comes from the daily climate report issued by the National Weather Service Office.

### Source

This data is published by <http://www.noah.gov>.

### Examples

```
data(central.park)
attach(central.park)
barplot(rbind(MIN,MAX-MIN),ylim=c(0,80))
```

---

central.park.cloud      *Type of day in Central Park, NY May 2003*

---

### Description

The type of day in May 2003 in Central Park, NY

### Usage

```
data(central.park.cloud)
```

### Format

A factor with levels clear,partly.cloudy and cloudy.

### Source

This type of data, and much more, is available from <http://www.noaa.gov>.

### Examples

```
data(central.park.cloud)
table(central.park.cloud)
```

---

cfb

*Bootstrap sample from the Survey of Consumer Finances*

---

**Description**

A bootstrap sample from the “Survey of Consumer Finances”.

**Usage**

```
data(cfb)
```

**Format**

A data frame with 1000 observations on the following 14 variables.

**WGT** Weights to compensate for undersampling. Not applicable

**AGE** Age of participants

**EDUC** Education level (number of years) of participant

**INCOME** Income in year 2001 of participant

**CHECKING** Amount in checking account for participant

**SAVING** Amount in savings accounts

**NMMF** Total directly-held mutual funds

**STOCKS** Amount held in stocks

**FIN** Total financial assets

**VEHIC** Value of all vehicles (includes autos, motor homes, RVs, airplanes, boats)

**HOMEEQ** Total home equity

**OTHNFIN** Other financial assets

**DEBT** Total debt

**NETWORTH** Total net worth

**Details**

The SCF dataset is a comprehensive survey of consumer finances sponsored by the United States Federal Reserve, <http://www.federalreserve.gov/pubs/oss/oss2/2001/scf2001home.html>.

The data is oversampled to compensate for low response in the upper brackets. To compensate, weights are assigned. By bootstrapping the data with the weights, we get a “better” version of a random sample from the population.

**Source**

<http://www.federalreserve.gov/pubs/oss/oss2/2001/scf2001home.html>

**Examples**

```
data(cfb)
attach(cfb)
mean(INCOME)
```

---

chicken	<i>weight gain of chickens fed 3 different rations</i>
---------	--

---

**Description**

weight gain of chickens fed 3 different rations

**Usage**

```
data(chicken)
```

**Format**

This data frame contains the following columns:

**Ration1** a numeric vector

**Ration2** a numeric vector

**Ration3** a numeric vector

**Source**

From Kitchens' Exploring Statistics.

**Examples**

```
data(chicken)
boxplot(chicken)
```

---

chips	<i>Measurements of chip wafers</i>
-------	------------------------------------

---

**Description**

The chips data frame has 30 rows and 8 columns.

**Usage**

```
data(chips)
```

**Format**

This data frame contains the following columns:

**wafer11** a numeric vector

**wafer12** a numeric vector

**wafer13** a numeric vector

**wafer14** a numeric vector

**wafer21** a numeric vector

**wafer22** a numeric vector

**wafer23** a numeric vector

**wafer24** a numeric vector

**Source**

From Kitchens' Exploring Statistics

**Examples**

```
data(chips)
boxplot(chips)
```

---

co2emiss

*Carbon Dioxide Emissions from the U.S.A. from fossil fuel*

---

**Description**

Carbon Dioxide Emissions from the U.S.A. from fossil fuel

**Usage**

```
data(co2emiss)
```

**Format**

The format is: Time-Series [1:276] from 1981 to 2004: -30.5 -30.4 -30.3 -29.8 -29.6 ...

**Details**

Monthly estimates of <sup>13</sup>C/<sup>12</sup>C in fossil-fuel CO<sub>2</sub> emissions. For write up see [http://cdiac.esd.ornl.gov/trends/emis\\_mon/emis\\_mon\\_co2.html](http://cdiac.esd.ornl.gov/trends/emis_mon/emis_mon_co2.html). In particular find:

"An annual cycle, peaking during the winter months and reflecting natural gas consumption, and a semi-annual cycle of lesser amplitude, peaking in summer and winter and reflecting coal consumption, comprise the dominant features of the annual pattern. The relatively constant emissions until 1987, followed by an increase from 1987-1989, a decrease in 1990-1991 and record highs during the late 1990s, are also evident in the annual data of Marland et al. However, emissions have declined somewhat since 2000."

**Source**

[http://cdiac.esd.ornl.gov/ftp/trends/emis\\_mon/emis\\_mon\\_c13.dat](http://cdiac.esd.ornl.gov/ftp/trends/emis_mon/emis_mon_c13.dat)

**Examples**

```
data(co2emiss)
monthplot(co2emiss)
stl(co2emiss, s.window="periodic")
```

---

coins	<i>The coins in my change bin</i>
-------	-----------------------------------

---

**Description**

The coins in author's change bin with year and value.

**Usage**

```
data(coins)
```

**Format**

A data frame with 371 observations on the following 2 variables.

**year** Year of coin

**value** Value of coin: quarter, dime, nickel, or penny

**Examples**

```
data(coins)
years = cut(coins$year, seq(1920, 2010, by=10), include.lowest=TRUE,
  labels = paste(192:200, "*", sep=""))
table(years)
```

---

coldvermont	<i>Daily minimum temperature in Woodstock Vermont</i>
-------------	---

---

**Description**

Recordings of daily minimum temperature in Woodstock Vermont from January 1 1980 through 1985.

**Usage**

```
data(coldvermont)
```

**Format**

A ts object with daily frequency

**Source**

Extracted from [http://www.ce.washington.edu/pub/HYDRO/edm/met\\_thru\\_97/vttmin.dly.gz](http://www.ce.washington.edu/pub/HYDRO/edm/met_thru_97/vttmin.dly.gz). Errors were possibly introduced.

**Examples**

```
data(coldvermont)
plot(coldvermont)
```

---

corn

*Comparison of corn for new and standard variety*

---

**Description**

Comparison of corn for new and standard variety

**Usage**

```
data(corn)
```

**Format**

This data frame contains the following columns:

**New** a numeric vector

**Standard** a numeric vector

**Source**

From Kitchens' Exploring Statitistics

**Examples**

```
data(corn)
t.test(corn)
```

---

crime	<i>violent crime rates in 50 states of US</i>
-------	---

---

**Description**

crime rates for 50 states in 1983 and 1993

**Usage**

```
data(crime)
```

**Format**

This data frame contains the following columns:

**y1983** a numeric vector

**y1993** a numeric vector

**Source**

from Kitchens' Exploring Statistics

**Examples**

```
data(crime)
boxplot(crime)
t.test(crime[,1],crime[,2],paired=TRUE)
```

---

deflection	<i>Deflection under load</i>
------------	------------------------------

---

**Description**

The data collected in a calibration experiment consisting of a known load, applied to the load cell, and the corresponding deflection of the cell from its nominal position.

**Usage**

```
data(deflection)
```

**Format**

A data frame with 40 observations on the following 2 variables.

**Deflection** a numeric vector

**Load** a numeric vector

**Source**

From an example in *Engineering Statistics Handbook* from <http://www.itl.nist.gov/div898/handbook/>

**Examples**

```
data(deflection)
res = lm(Deflection ~ Load, data = deflection)
plot(Deflection ~ Load, data = deflection)
abline(res) # looks good?
plot(res)
```

---

DensityPlot

*Plots densities of data*

---

**Description**

Allows one to compare empirical densities of different distributions in a simple manner. The density is used as graphs with multiple histograms are too crowded. The usage is similar to side-by-side boxplots.

**Usage**

```
DensityPlot(x, ...)
```

**Arguments**

x	x may be a sequence of data vectors (eg. x,y,z), a data frame with numeric column vectors or a model formula
...	You can pass in a bandwidth argument such as bw="SJ". See density for details. A legend will be placed for you automatically. To override the positioning set do.legend="manual". To skip the legend, set do.legend=FALSE.

**Value**

Makes a plot

**Author(s)**

John Verzani

**References**

Basically a modified boxplot function. As well it should be as it serves the same utility: comparing distributions.

**See Also**

[boxplot](#), [violinplot](#), [density](#)

**Examples**

```
## taken from boxplot
## using a formula
data(InsectSprays)
DensityPlot(count ~ spray, data = InsectSprays)
## on a matrix (data frame)
mat <- cbind(Uni05 = (1:100)/21, Norm = rnorm(100),
             T5 = rt(100, df = 5), Gam2 = rgamma(100, shape = 2))
DensityPlot(data.frame(mat))
```

---

diamond

*Price by size for diamond rings*

---

**Description**

A data set on 48 diamond rings containing price in Singapore dollars and size of diamond in carats.

**Usage**

```
data(diamond)
```

**Format**

A data frame with 48 observations on the following 2 variables.

**carat** A measurement of a diamond's size

**price** Price in Singapore dollars

**Details**

This data comes from a collection of the *Journal of Statistics Education*. The accompanying documentation says:

“Data presented in a newspaper advertisement suggest the use of simple linear regression to relate the prices of diamond rings to the weights of their diamond stones. The intercept of the resulting regression line is negative and significantly different from zero. This finding raises questions about an assumed pricing mechanism and motivates consideration of remedial actions.”

**Source**

This comes from <http://www.amstat.org/publications/jse/datasets/diamond.txt>. Data set is contributed by Singfat Chu.

**Examples**

```
data(diamond)
plot(price ~ carat, diamond, pch=5)
```

---

divorce	<i>Time until divorce for divorced women (by age)</i>
---------	---

---

**Description**

The divorce data frame has 25 rows and 6 columns.

**Usage**

```
data(divorce)
```

**Format**

This data frame contains the following columns:

**time of divorce** a factor

**all ages** a numeric vector

**0-17** a numeric vector

**18-19** a numeric vector

**20-24** a numeric vector

**25-100** a numeric vector

**Source**

Forgot source

**Examples**

```
data(divorce)
apply(divorce[,2:6],2,sum) # percent divorced by age of marriage
```

---

do *Function for repeating commands.*

---

**Description**

A function to facilitate performing of simulations

**Usage**

```
do(.n)
```

**Arguments**

.n                    A positive integer

**Details**

This function facilitates simulations. It is called in two steps. The first sets up a function which will to .n simulations. This function is then called with a block of commands and returns the simulation.

**Value**

Returns a function which will repeat a block of commands for purposes of simulation.

**Note**

do is due to Daniel Kaplan.

**Author(s)**

Daniel Kaplan

**See Also**

[replicate](#)

**Examples**

```
aFew <- do(5)
aLot <- do(100)
## simple call
aFew(mean(rnorm(100)))
## t-statistic simulation
aFew({x <- rexp(10); t.test(x)$stat})
## view correlation of regression coefficients. Returns a data.frame
x <- 1:10
res <- aLot({y <- rnorm(1+2*x); coef(lm(y~x))})
plot(res)
```

---

DOTplot	<i>Make big DOT plot like stripchart</i>
---------	--

---

**Description**

A variant of the `stripchart` using big dots as the default.

**Usage**

```
DOTplot(x, ...)
```

**Arguments**

<code>x</code>	May be a vector, data frame, matrix (each column a variable), list or model formula. Treats each variable or group as a univariate dataset and makes corresponding DOTplot.
<code>...</code>	arguments passed onto points.

**Value**

Returns the graphic only.

**Author(s)**

John Verzani

**See Also**

See also as [stripchart](#), [dotplot](#)

**Examples**

```
x = c(1,1,2,3,5,8)
DOTplot(x,main="Fibonacci",cex=2)
```

---

dottodot	<i>Dot-to-dot puzzle</i>
----------	--------------------------

---

**Description**

A set of points to make a dot-to-dot puzzle

**Usage**

```
data(dottodot)
```

**Format**

A data frame with 49 observations on the following 4 variables.

**x** x position

**y** y position

**pos** where to put label

**ind** number for label

**Details**

Points to make a dot to dot puzzle to illustrate, [text](#), [points](#), and the argument `pos`.

**Source**

Illustration by Noah Verzani.

**Examples**

```
data(dottodot)
# make a blank graph
plot(y~x,data=dottodot,type="n",bty="n",xaxt="n",xlab="",yaxt="n",ylab="")
# add the points
points(y~x,data=dottodot)
# add the labels using pos argument
with(dottodot, text(x,y,labels=ind,pos=pos))
# solve the puzzle
lines(y~x, data=dottodot)
```

---

dowdata

*The Dow Jones average from Jan 1999 to October 2000*

---

**Description**

The dowdata data frame has 443 rows and 5 columns.

**Usage**

```
data(dowdata)
```

**Format**

This data frame contains the following columns:

**Open** a numeric vector

**High** a numeric vector

**Date** a numeric vector

**Low** a numeric vector

**Close** a numeric vector

**Source**

this data comes from the site <http://www.forecasts.org/>

**Examples**

```
data(dowdata)
the.close <- dowdata$Close
n <- length(the.close)
plot(log(the.close[2:n]/the.close[1:(n-1)]))
```

---

dvdsales

*Monthly DVD player sales since introduction to May 2004*

---

**Description**

Monthly DVD player sales since introduction of DVD format to May 2004

**Usage**

```
data(dvdsales)
```

**Format**

Matrix with rows recording the year, and columns the month.

**Source**

<http://www.thedigitalbits.com/articles/cemadvdsales.html>

**Examples**

```
data(dvdsales)
barplot(t(dvdsales[7:1,]), beside=TRUE)
```

---

emissions

*CO2 emissions data and gross domestic product for 26 countries*

---

**Description**

The emissions data frame has 26 rows and 3 columns.

A data set listing GDP, GDP per capita, and CO2 emissions for 1999.

**Usage**

```
data(emissions)
```

**Format**

This data frame contains the following columns:

**GDP** a numeric vector

**perCapita** a numeric vector

**CO2** a numeric vector

**Source**

<http://www.grida.no> for CO2 data and <http://www.mrdowling.com> for GDP data.

Prompted by a plot appearing in a June 2001 issue of the *New York Times*.

**Examples**

```
data(emissions)
plot(emissions)
```

---

 ewr

*Taxi in and taxi out times at EWR (Newark) airport for 1999-2001*

---

**Description**

The ewr data frame has 46 rows and 11 columns.

Gives taxi in and taxi out times for 8 different airlines and several months at EWR airport.

Airline codes are AA (American Airlines), AQ (Aloha Airlines), AS (Alaska Airlines), CO (Continental Airlines), DL (Delta Airlines), HP (America West Airlines), NW (Northwest Airlines), TW (Trans World Airlines), UA (United Airlines), US (US Airways), and WN (Southwest Airlines)

**Usage**

```
data(ewr)
```

**Format**

This data frame contains the following columns:

**Year** a numeric vector

**Month** a factor for months

**AA** a numeric vector

**CO** a numeric vector

**DL** a numeric vector

**HP** a numeric vector

**NW** a numeric vector

**TW** a numeric vector

**UA** a numeric vector

**US** a numeric vector

**inorout** a factor with levels in or out

**Source**

<http://www.bts.gov/oai/taxitime/html/ewrtaxi.html>

**Examples**

```
data(ewr)
boxplot(ewr[3:10])
```

---

exec.pay

*Direct compensation for 199 United States CEOs in the year 2000*

---

**Description**

Direct compensation for 199 United States CEOs in the year 2000 in units of \\$/10,000.

**Usage**

```
data(exec.pay)
```

**Format**

A numeric vector with 199 entries each measuring compensation in 10,000s of dollars.

**Source**

*New York Times* Business section 04/01/2001. See also <http://www.aflcio.org>.

**Examples**

```
data(exec.pay)
hist(exec.pay)
```

---

fat

*Body measurements to predict percentage of body fat in males*

---

**Description**

A data set containing many physical measurements of 252 males. Most of the variables can be measured with a scale or tape measure. Can they be used to predict the percentage of body fat? If so, this offers an easy alternative to an underwater weighing technique.

**Usage**

```
data(fat)
```

**Format**

A data frame with 252 observations on the following 19 variables.

**case** Case Number

**body.fat** Percent body fat using Brozek's equation,  $457/\text{Density} - 414.2$

**body.fat.siri** Percent body fat using Siri's equation,  $495/\text{Density} - 450$

**density** Density (gm/cm<sup>2</sup>)

**age** Age (yrs)

**weight** Weight (lbs)

**height** Height (inches)

**BMI** Adiposity index =  $\text{Weight}/\text{Height}^2$  (kg/m<sup>2</sup>)

**ffweight** Fat Free Weight =  $(1 - \text{fraction of body fat}) * \text{Weight}$ , using Brozek's formula (lbs)

**neck** Neck circumference (cm)

**chest** Chest circumference (cm)

**abdomen** Abdomen circumference (cm) "at the umbilicus and level with the iliac crest"

**hip** Hip circumference (cm)

**thigh** Thigh circumference (cm)

**knee** Knee circumference (cm)

**ankle** Ankle circumference (cm)

**bicep** Extended biceps circumference (cm)

**forearm** Forearm circumference (cm)

**wrist** Wrist circumference (cm) "distal to the styloid processes"

**Details**

From the source:

"The data are as received from Dr. Fisher. Note, however, that there are a few errors. The body densities for cases 48, 76, and 96, for instance, each seem to have one digit in error as can be seen from the two body fat percentage values. Also note the presence of a man (case 42) over 200 pounds in weight who is less than 3 feet tall (the height should presumably be 69.5 inches, not 29.5 inches)! The percent body fat estimates are truncated to zero when negative (case 182)."

**Source**

This data set comes from the collection of the *Journal of Statistics Education* at <http://www.amstat.org/publications/jse/datasets/fat.txt>. The data set was contributed by Roger W. Johnson.

**References**

The source of the data is attributed to Dr. A. Garth Fisher, Human Performance Research Center, Brigham Young University, Provo, Utah 84602,

**Examples**

```
data(fat)
f = body.fat ~ age + weight + height + BMI + neck + chest + abdomen +
hip + thigh + knee + ankle + bicep + forearm + wrist
res = lm(f, data=fat)
summary(res)
```

---

father.son

*Pearson's data set on heights of fathers and their sons*

---

**Description**

1078 measurements of a father's height and his son's height.

**Usage**

```
data(father.son)
```

**Format**

A data frame with 1078 observations on the following 2 variables.

**fheight** Father's height in inches

**sheight** Son's height in inches

**Details**

Data set used by Pearson to investigate regression. See data set galton for data set used by Galton.

**Source**

Read into R by the command

```
read.table("http://stat-www.berkeley.edu/users/juliab/141C/pearson.dat", sep=" ")[, -1],
```

as mentioned by Chuck Cleland on the r-help mailing list.

**Examples**

```
data(father.son)
## like cover of Freedman, Pisani, and Purves
plot(sheight ~ fheight, data=father.son, bty="l", pch=20)
abline(a=0, b=1, lty=2, lwd=2)
abline(lm(sheight ~ fheight, data=father.son), lty=1, lwd=2)
```

---

`female.inc`*Income distribution for females in 2001*

---

### Description

A data set containing incomes for 1,000 females along with race information. The data is sampled from data provided by the United States Census Bureau.

### Usage

```
data(female.inc)
```

### Format

A data frame with 1,000 observations on the following 2 variables.

**income** Income for 2001 in dollars

**race** a factor with levels black, hispanic or white

### Details

The United States Census Bureau provides a lot of data on income distributions. This data comes from the Current Population Survey (CPS) for the year 2001. The raw data appears in table format. This data is sampled from the data in that table.

### Source

The original table is found at [http://ferret.bls.census.gov/macro/032002/perinc/new11\\_002.htm](http://ferret.bls.census.gov/macro/032002/perinc/new11_002.htm).

### Examples

```
data(female.inc)
boxplot(income ~ race, female.inc)
boxplot(log(income,10) ~ race, female.inc)
sapply(with(female.inc,split(income,race)),median)
```

---

firstchi	<i>Age of mother at birth of first child</i>
----------	--

---

**Description**

Age of mother at birth of first child

**Usage**

```
data(firstchi)
```

**Format**

The format is: num [1:87] 30 18 35 22 23 22 36 24 23 28 ...

**Source**

From *Exploring Statistics*, L. Kitchens, Duxbury Press, 1998.

**Examples**

```
data(firstchi)
hist(firstchi)
```

---

five.yr.temperature	<i>Five years of weather in New York City</i>
---------------------	---

---

**Description**

Five years of maximum temperatures in New York City

**Usage**

```
data(five.yr.temperature)
```

**Format**

A data frame with 2,439 observations on the following 3 variables.

**days** Which day of the year

**years** The year

**temps** Maximum temperature

**Source**

Dataset found on the internet, but original source is lost.

### Examples

```
data(five.yr.temperature)
attach(five.yr.temperature)
scatter.smooth(temps ~ days,col=gray(.75))
lines(smooth.spline(temps ~ days), lty=2)
lines(supsmu(days, temps), lty=3)
```

---

florida	<i>County-by-county results of year 2000 US presidential election in Florida</i>
---------	--

---

### Description

The florida data frame has 67 rows and 13 columns.

Gives a county by county accounting of the US elections in the state of Florida.

### Usage

```
data(floriga)
```

### Format

This data frame contains the following columns:

**County** Name of county

**GORE** Votes for Gore

**BUSH** Votes for Bush

**BUCHANAN** Votes for Buchanan

**NADER** Votes for Nader

**BROWN** a numeric vector

**HAGELIN** a numeric vector

**HARRIS** a numeric vector

**MCREYNOLDS** a numeric vector

**MOOREHEAD** a numeric vector

**PHILLIPS** a numeric vector

**Total** a numeric vector

### Source

Found in the excellent notes *Using R for Data Analysis and Graphics* by John Maindonald. (As of 2003 a book published by Cambridge University Press.)

**Examples**

```

data(flórida)
attach(flórida)
result.lm <- lm(BUCHANAN ~ BUSH)
plot(BUSH, BUCHANAN)
abline(result.lm) ## can you find Palm Beach and Miami Dade counties?

```

galileo

*Galileo data on falling bodies***Description**

Data recorded by Galileo in 1609 during his investigations of the trajectory of a falling body.

**Usage**

```
data(galileo)
```

**Format**

A data frame with 7 observations on the following 2 variables.

**init.h** Initial height of ball

**h.d** Horizontal distance traveled

**Details**

A simple ramp 500 punti above the ground was constructed. A ball was placed on the ramp at an indicated height from the ground and released. The horizontal distance traveled is recorded (in punti). (One punto is 169/180 millimeter, not a car by FIAT.)

**Source**

This data and example come from the *Statistical Sleuth* by Ramsay and Schafer, Duxbury (2001), section 10.1.1. They attribute an article in *Scientific American* by Drake and MacLachlan.

**Examples**

```

data(galileo)
polynomial = function(x,coefs) {
  sum = 0
  for(i in 0:(length(coefs)-1)) {
    sum = sum + coefs[i+1]*x^i
  }
  sum
}
res.lm = lm(h.d ~ init.h, data = galileo)
res.lm2 = update(res.lm, . ~ . + I(init.h^2), data=galileo)
res.lm3 = update(res.lm2, . ~ . + I(init.h^3), data=galileo)

```

```
plot(h.d ~ init.h, data = galileo)
curve(polynomial(x,coef(res.lm)),add=TRUE)
curve(polynomial(x,coef(res.lm2)),add=TRUE)
curve(polynomial(x,coef(res.lm3)),add=TRUE)
```

---

galton

*Galton's height data for parents and children*


---

### Description

Data set from tabulated data set used by Galton in 1885 to study the relationship between a parent's height and their childrens.

### Usage

```
data(galton)
```

### Format

A data frame with 928 observations on the following 2 variables.

**child** The child's height

**parent** The "midparent" height

### Details

The midparent's height is an average of the fathers height and 1.08 times the mother's. In the data there are 205 different parents and 928 children. The data here is truncated at the ends for both parents and children so that it can be treated as numeric data. The data were tabulated and consequently made discrete. The father . son data set is similar data used by Galton and is continuous.

### Source

This data was found at <http://www.bun.kyoto-u.ac.jp/~suchii/galton86.html>.

See also the data.set [father.son](http://stat-www.berkeley.edu/users/juliab/141C/pearson.dat) which was found from <http://stat-www.berkeley.edu/users/juliab/141C/pearson.dat>.

### Examples

```
data(galton)
plot(galton)
## or with some jitter.
plot(jitter(child,5) ~ jitter(parent,5),galton)
## sunflowerplot shows flowers for multiple plots (Thanks MM)
sunflowerplot(galton)
```

---

gap

*Sales data for the Gap*

---

**Description**

Sales data for the Gap from Jan

**Usage**

```
data(gap)
```

**Format**

The format is a ts object storing data from June 2002 through June 2005.

**Source**

<http://home.businesswire.com>

**Examples**

```
data(gap)
monthplot(gap)
```

---

gasprices

*Monthly average gasoline prices in the United States*

---

**Description**

Average retail gasoline prices per month in the United States from January 2000 through February 2006. The hurricane Katrina caused a percentage loss of refinery capability leading to rapidly increasing prices.

**Usage**

```
data(gasprices)
```

**Format**

The format is: Time-Series [1:74] from 2000 to 2006: 129 138 152 146 148 ...

**Source**

From the Department of Energy web site: <http://tonto.eia.doe.gov/oog/info/gdu/gaspump.html>

**Examples**

```
data(gasprices)
plot(gasprices)
```

---

`getAnswer`*Display answers to selected problems*

---

**Description**

Displays answers to selected problems in the system's web browser .

**Usage**

```
getAnswer(chapter = NULL, problem = NULL)
errata()
```

**Arguments**

chapter	The chapter number
problem	The problems number Not all answers are available.

**Details**

Some selected answers from the problems in *Using R for Introductory Statistics* are available from the **UsingR** package The `getAnswer` function will display them one-by-one in the browser.

The `errata` function will display the list of errata.

**Value**

If available, opens web browser to the requested answer or errata page

**Author(s)**

John Verzani

**Examples**

```
getAnswer()
```

---

goalspergame	<i>Goals per game in NHL</i>
--------------	------------------------------

---

**Description**

Goals per game in NHL

**Usage**

```
data(goalspergame)
```

**Format**

The format is: mts [1:53, 1:4] 6 6 6 6 6 6 6 6 6 ... - attr(\*, "dimnames")=List of 2 ..\$ : NULL ..\$ : chr [1:4] "n.teams" "n.games" "n.goals" "gpg" - attr(\*, "tsp")= num [1:3] 1946 1998 1 - attr(\*, "class")= chr [1:2] "mts" "ts"

**Source**

Off internet site. Forgot which.

**Examples**

```
data(goalspergame)
```

---

google	<i>Google stock values during 2005-02-07 to 2005-07-07</i>
--------	--

---

**Description**

Closing stock price of a share of Google stock during 2005-02-07 to 2005-07-07

**Usage**

```
data(google)
```

**Format**

A data vector of numeric values with names attribute giving the dates.

**Source**

finance.yahoo.com

**Examples**

```
data(google)
plot(google, type="l")
```

---

grades	<i>Current and previous grades</i>
--------	------------------------------------

---

**Description**

A dataframe of a students grade and their grade in their previous class. Graded on American A-F scale.

**Usage**

```
data(grades)
```

**Format**

A dataframe of 122 rows with 2 columns

**prev** The grade in the previous class in the subject matter

**grade** The grade in the current class

**Examples**

```
data(grades)
table(grades)
```

---

grip	<i>Effects of cross-country ski-pole grip</i>
------	---

---

**Description**

Simulated data set investigating effects of cross-country ski-pole grip.

**Usage**

```
data(grip)
```

**Format**

A data frame with 36 observations on the following 4 variables.

**UBP** Measurement of upper-body power

**person** One of four skiers

**grip.type** Either classic, modern, or integrated.

**replicate** a numeric vector

## Details

Based on a study done at <http://www.montana.edu/wwwhhd/movementscilab/> mentioned at <http://www.xskiworld.com/>. The study investigates the effect of grip type on upper body power. As this influences performance in races, presumably a skier would prefer the grip that provides the best power output.

## Examples

```
data(grip)
ftable(xtabs(UBP ~ person + replicate + grip.type,grip))
```

---

hall.fame	<i>Data frame containing baseball statistics including Hall of Fame membership</i>
-----------	--

---

## Description

A data frame containing baseball statistics for several players.

## Usage

```
data(hall.fame)
```

## Format

A data frame with 1340 observations on the following 28 variables.

**first** first name  
**last** last name  
**seasons** Seasons played  
**games** Games played  
**AB** Official At Bats  
**runs** Runs scored  
**hits** hits  
**doubles** doubles  
**triples** triples numeric vector  
**HR** Home runs  
**RBI** Runs batted in  
**BB** Base on balls  
**SO** Strike outs  
**BA** Batting Average  
**OBP** On Base percentage  
**SP** Slugging Percentage

**AP** Adjusted productions  
**BR** batting runs  
**ABRuns** adjusted batting runs  
**Runs.Created** Runs created  
**SB** Stolen Bases  
**CS** Caught stealing  
**Stolen.Base.Runs** Runs scored by stealing  
**Fielding.Average** Fielding average  
**Fielding.Runs** Fielding runs  
**Primary.Position.Played** C = Catcher, 1 = First Base, 2 = Second Base, 3 = Third Base, S = Shortstop, O = Outfield, and D = Designated hitter  
**Total.Player.Rating** a numeric vector  
**Hall.Fame.Membership** Not a member, Elected by the BBWAA, or Chosen by the Old Timers Committee or Veterans Committee

### Details

The sport of baseball lends itself to the collection of data. This data set contains many variables used to assess a players career. The Hall of Fame is reserved for outstanding players as judged initially by the Baseball Writers Association and subsequently by the Veterans Committee.

### Source

This data set was submitted to the *Journal of Statistical Education*, <http://www.amstat.org/publications/jse/secure/v8n2/datasets.cochran.new.cfm>, by James J. Cochran.

### Examples

```
data(hall.fame)
hist(hall.fame$OBP)
with(hall.fame,last[Hall.Fame.Membership != "not a member"])
```

---

healthy

*Healthy or not?*

---

### Description

Data on whether a patient is healthy with two covariates.

### Usage

```
data(healthy)
```

**Format**

A data frame with 32 observations on the following 3 variables.

**p** One covariate

**g** Another covariate

**healthy** 0 is healthy, 1 is not

**Details**

Data on health with information from two unspecified covariates.

**Examples**

```
data(healthy)
library(MASS)
stepAIC(glm(healthy ~ p + g, healthy, family=binomial))
```

---

heartrate

*Simulated data of age vs. max heart rate*

---

**Description**

Simulated data of age vs. max heart rate

**Usage**

```
data(heartrate)
```

**Format**

This data frame contains the following columns:

**age** a numeric vector

**maxrate** a numeric vector

**Details**

Does this fit the workout room value of 220 - age?

**Source**

Simulated based on “Age-predicted maximal heart rate revisited” Hirofumi Tanaka, Kevin D. Monahan, Douglas R. Seals *Journal of the American College of Cardiology*, 37:1:153-156.

**Examples**

```
data(heartrate)
plot(heartrate)
abline(lm(maxrate ~ age,data=heartrate))
```

---

home	<i>Maplewood NJ homedata</i>
------	------------------------------

---

**Description**

The home data frame has 15 rows and 2 columns.

**Usage**

```
data(home)
```

**Format**

This data frame contains the following columns:

**old** a numeric vector

**new** a numeric vector

**Details**

See full dataset homedata

**Source**

See full dataset homedata

**Examples**

```
data(home)
## compare on the same scale
boxplot(data.frame(scale(home)))
```

---

homedata	<i>Maplewood NJ assessed values for years 1970 and 2000</i>
----------	---

---

**Description**

The homedata data frame has 6841 rows and 2 columns.

Data set containing assessed values of homes in Maplewood NJ for the years 1970 and 2000. The properties were not officially assessed during that time and it is interesting to see the change in percentage appreciation.

**Usage**

```
data(homedata)
```

**Format**

This data frame contains the following columns:

**y1970** a numeric vector

**y2000** a numeric vector

**Source**

Maplewood Reval

**Examples**

```
data(homedata)
plot(homedata)
```

---

homeprice

*Sale price of homes in New Jersey in the year 2001*

---

**Description**

The homeprice data frame has 29 rows and 7 columns.

**Usage**

```
data(homeprice)
```

**Format**

This data frame contains the following columns:

**list** list price of home (in thousands)

**sale** actual sale price

**full** Number of full bathrooms

**half** number of half bathrooms

**bedrooms** number of bedrooms

**rooms** total number of rooms

**neighborhood** Subjective assessment of neighborhood on scale of 1-5

**Details**

This dataset is a random sampling of the homes sold in Maplewood, NJ during the year 2001. Of course the prices will either seem incredibly high or fantastically cheap depending on where you live, and if you have recently purchased a home.

**Source**

Source Burgdorff Realty.

**Examples**

```
data(homeprice)
plot(homeprice$sale,homeprice$list)
abline(lm(homeprice$list~homeprice$sale))
```

---

homework	<i>Homework averages for Private and Public schools</i>
----------	---

---

**Description**

Homework averages for Private and Public schools

**Usage**

```
data(homework)
```

**Format**

This data frame contains the following columns:

**Private** a numeric vector

**Public** a numeric vector

**Source**

This is from Kitchens Exploring Statistics

**Examples**

```
data(homework)
boxplot(homework)
```

---

HUMMER	<i>Deliveries of new HUMMER vehicles</i>
--------	--

---

**Description**

Gives monthly delivery numbers for new HUMMER vehicles from June 2003 through February 2006. During July, August, and September 2005 there was an Employee Pricing Incentive.

**Usage**

```
data(HUMMER)
```

**Format**

The format is: Time-Series [1:33] from 2003 to 2006: 2493 2654 2987 2837 3157 2837 3157 1927 2141 2334 ...

**Source**

Compiled from delivery data available at [http://www.gm.com/company/investor\\_information/sales\\_prod/hist\\_sales.html](http://www.gm.com/company/investor_information/sales_prod/hist_sales.html)

**Examples**

```
data(HUMMER)
plot(HUMMER)
```

---

iq	<i>IQ scores</i>
----	------------------

---

**Description**

simulated IQ scores

**Usage**

```
data(iq)
```

**Format**

The format is: num [1:100] 72 75 77 77 81 82 83 84 84 86 ...

**Source**

From Kitchens Exploring Statistics

**Examples**

```
data(iq)
qqnorm(iq)
```

---

`kid.weights`*Weight and height measurement for a sample of U.S. children*

---

## Description

A sample from the data presented in the NHANES III survey (<http://www.cdc.gov/nchs/nhanes.htm>). This survey is used to form the CDC Growth Charts (<http://www.cdc.gov/growthcharts>) for children.

## Usage

```
data(kid.weights)
```

## Format

A data frame with 250 observations on the following 4 variables.

**age** Age in months

**weight** weight in pounds

**height** height in inches

**gender** Male of Female

## Source

This data is extracted from the NHANES III survey: <http://www.cdc.gov/nchs/nhanes.htm>.

## Examples

```
data(kid.weights)
attach(kid.weights)
plot(weight,height,pch=as.character(gender))
## find the BMI -- body mass index
m.ht = height*2.54/100      # 2.54 cm per inch
m.wt = weight / 2.2046     # 2.2046 lbs. per kg
bmi = m.wt/m.ht^2
hist(bmi)
```

---

KSI

*Data set on automobile deaths and injuries in Great Britain*

---

### Description

Data on car drivers killed, car drivers killed or seriously injured (KSI), and light goods drivers killed during the years 1969 to 1984 in Great Britain. In February 1982 a compulsory seat belt law was introduced.

### Usage

```
data(KSI)
```

### Format

The data is stored as a multi-variate zoo object.

### Source

Data copied from Appendix 2 "Forecasting, structural time series, models and the Kalman Filter" by Andrew Harvey. The lg.k data is also found in the vandrivers dataset contained in the sspir package.

### References

Source: HMSO: Road Accidents in Great Britain 1984.

### Examples

```
data(KSI)
plot(KSI)
seatbelt = time(KSI) < 1983 + (2-1)/12
```

---

last.tie

*Last tie in 100 coin tosses*

---

### Description

Toss a coin 100 times and keep a running count of the number of heads and the number of tails. Record the times when the number is tied and report the last one. The distribution will have an approximate "arc-sine" law or well-shaped distribution.

### Usage

```
data(last.tie)
```

**Format**

200 numbers between 0 and 100 indicating when the last tie was.

**Details**

This data comes from simulating the commands: `x = cumsum(sample(c(-1,1),100,replace=T))`  
and then finding the last tie with

```
last.tie[i]<-max(0,max(which(!sign(x) == sign(x[length(x)]))))).
```

**Examples**

```
data(last.tie)
hist(last.tie)
```

---

lawsuits

*Law suit settlements*

---

**Description**

A simulated dataset on the settlement amount of 250 lawsuits based on values reported by Class Action Reports.

**Usage**

```
data(lawsuits)
```

**Format**

The format is: num [1:250] 16763 10489 17693 14268 442 ...

**Details**

Class Action Reports completed an extensive survey of attorney fee awards from 1,120 common fund class actions (Volume 24, No. 2, March/April 2003). The full data set is available for a fee. This data is simulated from the values published in an excerpt.

**Source**

<http://www.classactionreports.com/classactionreports/attorneyfee.htm>

**References**

See also "Study Disputes View of Costly Surge in Class-Action Suits" by Jonathan D. Glater in the January 14, 2004 New York Times which cites a Jan. 2004 paper in the *Journal of Empirical Legal Studies* by Eisenberg and Miller.

**Examples**

```
data(lawsuits)
mean(lawsuits)
median(lawsuits)
```

---

malpract	<i>malpractice settlements</i>
----------	--------------------------------

---

**Description**

malpractice settlements

**Usage**

```
data(malpract)
```

**Format**

The format is: num [1:17] 760 380 125 250 2800 450 100 150 2000 180 ...

**Source**

From Kitchens Exploring Statistics

**Examples**

```
data(malpract)
boxplot(malpract)
```

---

mandms	<i>Proportions of colors in various M and M's varieties</i>
--------	---

---

**Description**

A bag of the candy M and M's has many different colors. Each large production batch is blended to the ratios given in this data set. The batches are thoroughly mixed and then the individual packages are filled by weight using high-speed equipment, not by count.

**Usage**

```
data(mandms)
```

**Format**

A data frame with 5 observations on the following 6 variables.

**blue** percentage of blue

**brown** percentage of brown

**green** percentage of green

**orange** percentage of orange

**red** percentage of red

**yellow** percentage of yellow

**Source**

This data is attributed to an email sent by Masterfoods USA, A Mars, Incorporated Company. This email was archived at the Math Forum, <http://www.mathforum.org>.

**Examples**

```
data(mandms)
bagfull = c(15,34,7,19,29,24)
names(bagfull) = c("blue", "brown", "green", "orange", "red", "yellow")
prop = function(x) x/sum(x)
chisq.test(bagfull,p = prop(mandms["milk chocolate",]))
chisq.test(bagfull,p = prop(mandms["Peanut",]))
```

---

math

*Standardized math scores*


---

**Description**

Standardized math scores

**Usage**

```
data(math)
```

**Format**

The format is: num [1:30] 44 49 62 45 51 59 57 55 70 64 ...

**Source**

From Larry Kitchens, *Exploring Statistics*, Duxbury Press.

**Examples**

```
data(math)
hist(math)
```

---

maydow

*Dow Jones industrial average and May maximum temperature*

---

### Description

A data set of both the Dow Jones industrial average and the maximum daily temperature in New York City for May 2003.

### Usage

```
data(maydow)
```

### Format

A data frame with 21 observations on the following 3 variables.

**Day** Day of the month

**DJA** The daily close of the DJIQ

**max.temp** Daily maximum temperature in Central Park

### Details

Are stock traders influenced by the weather? This dataset looks briefly at this question by comparing the daily close of the Dow Jones industrial average with the maximum daily temperature for the month of May 2003. This month was rainy and unseasonably cool weather wise, yet the DJIA did well.

### Source

The DJIA data is from <http://finance.yahoo.com> the temperature data from <http://www.noaa.gov>.

### Examples

```
data(maydow)
attach(maydow)
plot(max.temp,DJA)
plot(max.temp[-1],diff(DJA))
```

midsize

*Price of new and used of three mid-sized cars*

---

**Description**

New and used prices of three popular mid-sized cars.

**Usage**

```
data(midsize)
```

**Format**

A data frame with 15 observations on the following 4 variables.

**Year** 2004 is new car price, others are for used car

**Accord** Honda Accord

**Camry** Toyota Camry

**Taurus** Ford Taurus

**Details**

The value of a car depreciates over time. This data gives the price of a new car and values of similar models for previous years as reported by <http://www.edmunds.com>.

**Examples**

```
data(midsize)
plot(Accord ~ I(2004-Year), data = midsize)
```

---

MLBattend*Major league baseball attendance data*

---

**Description**

Data on home-game attendance in Major League Baseball for the years 1969-2000.

**Usage**

```
data(MLBattend)
```

**Format**

A data frame with 838 observations on the following 10 variables.

**franchise** Which team

**league** American or National league

**division** Which division

**year** The year (the year 2000 is recorded as 0)

**attendance** Actual attendance

**runs.scored** Runs scored by the team during year

**runs.allowed** Runs allows by the team during year

**wins** Number of wins for season

**losses** Number of losses for season

**games.behind** A measure of how far from division winner the team was. Higher numbers are worse.

**Source**

This data was submitted to *The Journal of Statistical Education* by James J. Cochran, <http://www.amstat.org/publications/jse/v10n2/datasets.cochran.html>.

**Examples**

```
data(MLBattend)
boxplot(attendance ~ franchise, MLBattend)
with(MLBattend, cor(attendance,wins))
```

---

movies

*Data frame on top 25 movies for some week, many weeks ago*

---

**Description**

Data on 25 top movies

**Usage**

```
data(movies)
```

**Format**

A data frame with 26 observations on the following 5 variables.

**title** Titles

**current** Current week

**previous** Previous weel

**gross** Total

**Source**

Some movie website, sorry lost the url.

**Examples**

```
data(movies)
boxplot(movies$previous)
```

---

mw.ages

*Age distribution in year 2000 in Maplewood New Jersey*

---

**Description**

Age distribution in Maplewood New Jersey, a suburb of New York City. Data is broken down by Male and Female.

**Usage**

```
data(mw.ages)
```

**Format**

A data frame with 103 observations on the following 2 variables.

**Male** Counts per age group. Most groups are 1 year, except for 100-104, 105-110, 110+

**Female** Same

**Source**

US Census 2000 data from <http://factfinder.census.gov/>

**Examples**

```
data(mw.ages)
barplot(mw.ages$Male + mw.ages$Female)
```

---

nba.draft	<i>NBA draft lottery odds for 2002</i>
-----------	--

---

**Description**

The NBA draft in 2002 has a lottery

**Usage**

```
data(nba.draft)
```

**Format**

A data frame with 13 observations on the following 2 variables.

**Team** Team name

**Record** The team won-loss record

**Balls** The number of balls (of 1000) that this team has in the lottery selection

**Details**

The NBA draft has a lottery to determining the top 13 placings. The odds in the lottery are determined by the won-loss record of the team, with poorer records having better odds of winning.

**Source**

Data is taken from [http://www.nba.com/news/draft\\_ties\\_020424.html](http://www.nba.com/news/draft_ties_020424.html).

**Examples**

```
data(nba.draft)
top.pick = sample(row.names(nba.draft),1,prob = nba.draft$Balls)
```

---

normtemp	<i>Body temperature and heart rate of 130 health individuals</i>
----------	--

---

**Description**

A data set used to investigate the claim that “normal” temperature is 98.6 degrees.

**Usage**

```
data(normtemp)
```

**Format**

A data frame with 130 observations on the following 3 variables.

**temperature** normal body temperature

**gender** Gender 1 = male, 2 = female

**hr** Resting heart rate

**Details**

Is normal body temperature 98.6 degrees Fahrenheit? This dataset was constructed to match data presented in an article intending to establish the true value of “normal” body temperature.

**Source**

This data set was contributed by Allen L. Shoemaker to the *Journal of Statistics Education*, <http://www.amstat.org/publications/jse/datasets/normtemp.txt>.

**References**

Data set is simulated from values contained in Mackowiak, P. A., Wasserman, S. S., and Levine, M. M. (1992), "A Critical Appraisal of 98.6 Degrees F, the Upper Limit of the Normal Body Temperature, and Other Legacies of Carl Reinhold August Wunderlich," *Journal of the American Medical Association*, 268, 1578-1580.

**Examples**

```
data(normtemp)
hist(normtemp$temperature)
t.test(normtemp$temperature, mu=98.2)
summary(lm(temperature ~ factor(gender), normtemp))
```

---

npdb

*National Practitioner Data Bank*

---

**Description**

Selected variables from the publicly available data from the National Practitioner Data Bank (NPDB).

**Usage**

```
data(npdb)
```

## Format

A data frame with 6797 observations on the following 6 variables.

**state** 2 digit abbreviation of state

**field** Field of practice

**age** Age of practitioner (rounded down to 10s digit)

**year** Year of claim

**amount** Dollar amount of reward

**ID** a practioner ID, masked for anonymity

The variable names do not match the original. The codings for field come from a document on <http://63.240.212.200/publicdata.html>.

## Details

This dataset excerpts some interesting variables from the NPDB for the years 2000-2003. The question of capping medical malpractice awards to lower insurance costs is currently being debated nationwide (U.S.). This data is a primary source for determining this debate.

A quotation from <http://www.npdb-hipdb.com/npdb.html>:

“The legislation that led to the creation of the NPDB was enacted the U.S. Congress believed that the increasing occurrence of medical malpractice litigation and the need to improve the quality of medical care had become nationwide problems that warranted greater efforts than any individual State could undertake. The intent is to improve the quality of health care by encouraging State licensing boards, hospitals and other health care entities, and professional societies to identify and discipline those who engage in unprofessional behavior; and to restrict the ability of incompetent physicians, dentists, and other health care practitioners to move from State to State without disclosure or discovery of previous medical malpractice payment and adverse action history. Adverse actions can involve licensure, clinical privileges, professional society membership, and exclusions from Medicare and Medicaid.”

## Source

This data came from <http://www.npdb-hipdb.com/npdb.html>

## Examples

```
data(npdb)
table(table(npdb$ID)) # big offenders
hist(log(npdb$amount)) # log normal?
```

---

nym.2002

*Random sample of 2002 New York City Marathon finishers*

---

### **Description**

A random sample of finishers from the New York City Marathon.

### **Usage**

```
data(nym.2002)
```

### **Format**

A data frame with 1000 observations on the following 5 variables.

**place** Place in the race

**gender** What gender

**age** Age on day of race

**home** Indicator of hometown or nation

**time** Time in minutes to finish

### **Details**

Each year thousands of participants line up to run the New York City Marathon. This list is a random sample from the finishers.

### **Source**

From the New York City Road Runners web site <http://www.nyrrc.org>

### **Examples**

```
data(nym.2002)
with(nym.2002, cor(time,age))
```

---

 OBP

*On base percentage for 2002 major league baseball season*


---

**Description**

The on base percentage, OBP, is a measure of how often a players gets on base. It differs from the more familiar batting average, as it include bases on balls (BB) and hit by pitches (HBP). The exact formula is  $OBP = (H + BB + HBP) / (AB + BB + HBP + SF)$ .

**Usage**

```
data(OBP)
```

**Format**

438 numbers between 0 and 1 corresponding the on base “percentage” for the 438 players who had 100 or more at bats in the 2002 baseball season. The "outlier" is Barry Bonds.

**Source**

This data came from the interesting Lahman baseball data base <http://www.baseball11.com/>. The names attribute uses the playerID from this database. Unfortunately there were some errors in the extraction from the original data set. Consult the original for accurate numbers.

**Examples**

```
data(OBP)
hist(OBP)
OBP[OBP>.5] # who is better than 50%? (only Barry Bonds)
```

---

 oral.lesion

*Oral lesion location by town*


---

**Description**

A data set on oral lesion location for three Indian towns.

**Usage**

```
data(oral.lesion)
```

**Format**

A data frame with 9 observations on the following 3 variables.

**Kerala** a numeric vector

**Gujarat** a numeric vector

**Andhra** a numeric vector

**Source**

"Exact Inference for Categorical Data", by Cyrus R. Mehta and Nitin R. Patel. Found at <http://www.cytel.com/papers/sxpaper.pdf>.

**Examples**

```
data(oral.lesion)
chisq.test(oral.lesion)$p.value
chisq.test(oral.lesion,simulate.p.value=TRUE)$p.value ## exact is.0269
```

---

ozonemonthly	<i>Monthly mean ozone values at Halley Bay Antarctica</i>
--------------	---

---

**Description**

A time series showing ozone values at Halley Bay Antarctica

**Usage**

```
data(ozonemonthly)
```

**Format**

The format is: Time-Series [1:590] from 1957 to 2006: 313 311 370 359 334 296 288 274 NA NA ... - attr(\*, "names")= chr [1:590] "V5" "V6" "V7" "V8" ...

**Details**

Provisional monthly mean ozone values for Halley Bay Antarctica between 1956 and 2005. Data comes from <http://www.antarctica.ac.uk/met/jds/ozone/>.

**Source**

<http://www.antarctica.ac.uk/met/jds/ozone/data/ZNOZ.DAT>

**References**

See [www.meteohistory.org/2004proceedings1.1/pdfs/11christie.pdf](http://www.meteohistory.org/2004proceedings1.1/pdfs/11christie.pdf) for a discussion of data collection and the Ozone hole.

**Examples**

```
data(ozonemonthly)
## notice decay in the 80s
plot(ozonemonthly)
## October plot shows dramatic swing
monthplot(ozonemonthly)
```

---

paradise

*Annual snowfall at Paradise Ranger Station, Mount Ranier*

---

**Description**

Annual snowfall (from July 1 to June 30th) measured at Paradise ranger station at Mount Ranier Washington.

**Usage**

```
data(paradise)
```

**Format**

The data is stored as a zoo class object. The time index refers to the year the snowfall begins.

**Details**

Due to its rapid elevation gain, and proximity to the warm moist air of the Pacific Northwest record amounts of snow can fall on Mount Ranier. This data set shows the fluctuations.

**Source**

<http://www.nps.gov/mora/current/weather.htm>

**Examples**

```
require(zoo)
data(paradise)
range(paradise, na.rm=TRUE)
plot(paradise)
```

---

pi2000

*first 2000 digits of pi*

---

**Description**

first 2000 digits of pi

**Usage**

```
data(pi2000)
```

**Format**

The format is: num [1:2000] 3 1 4 1 5 9 2 6 5 3 ...

**Source**

Generated by *Mathematica*, <http://www.wolfram.com>.

**Examples**

```
data(pi2000)
chisq.test(table(pi2000))
```

---

primes

*Primes numbers less than 2003*

---

**Description**

Prime numbers between 1 and 2003.

**Usage**

```
data(primes)
```

**Format**

The format is: num [1:304] 2 3 5 7 11 13 17 19 23 29 ...

**Source**

Generated using <http://www.rsok.com/~jrm/printprimes.html>.

**Examples**

```
data(primes)
diff(primes)
```

---

puerto

*Incomes for Puerto Rican immigrants to Miami*

---

**Description**

Incomes for Puerto Rican immigrants to Miami

**Usage**

```
data(puerto)
```

**Format**

The format is: num [1:50] 150 280 175 190 305 380 290 300 170 315 ...

**Source**

From Kitchens Exploring Statistics

**Examples**

```
data(puerto)
hist(puerto)
```

---

**QQplot***Creates a qqplot with shaded density estimate*

---

**Description**

Creates a qqplot of two variables along with graphs of their densities, shaded so that the corresponding percentiles are clearly matched up.

**Usage**

```
QQplot(x, y, n = 20, xsf = 4, ysf = 4, main = "qqplot", xlab = deparse(substitute(x)), ylab = deparse(substitute(y)))
```

**Arguments**

x	The x variable
y	The y variable
n	number of points to plot in qqplot.
xsf	scale factor to adjust size of x density graph
ysf	scale factor to adjust size of y density graph
main	title
xlab	label for x axis
ylab	label for y axis
pch	plot character for points in qqplot
pcol	color of plot character
shade	shading color
...	extra arguments passed to plot.window

**Details**

Shows density estimates for the two samples in a qqplot. Meant to make this useful plot more transparent to first-time users of quantile-quantile plots.

This function has some limitations: the scale factor may need to be adjusted; the code to shade only shaded trapezoids, and does not completely follow the density.

**Value**

Produces a graphic

**Author(s)**

John Verzani

**See Also**

[qqplot](#), [qqnorm](#)

**Examples**

```
x = rnorm(100)
y = rt(100, df=3)
QQplot(x,y)
```

---

rat

*Survival times of 20 rats exposed to radiation*

---

**Description**

Survival times of 20 rats exposed to radiation

**Usage**

```
data(rat)
```

**Format**

The format is: num [1:20] 152 152 115 109 137 88 94 77 160 165 ...

**Source**

From Kitchens Exploring Statistics

**Examples**

```
data(rat)
hist(rat)
```

---

reaction.time	<i>Reaction time with cell phone usage</i>
---------------	--

---

**Description**

A simulated dataset on reaction time to an external event for subject using cell phones.

**Usage**

```
data(reaction.time)
```

**Format**

A data frame with 60 observations on the following 4 variables.

**age** Age of participant coded as 16-24 or 25+

**gender** Male or Female

**control** Code to indicate if subject is using a cell phone "T" or is in the control group "C"

**time** Time in seconds to react to external event

**Details**

Several studies indicate that cell phone usage while driving can effect reaction times to external events. This dataset uses simulated data based on values from the NHTSA study "The Influence of the Use of Mobile Phones on Driver Situation Awareness".

**Source**

The NHTSA study may be found at <http://www-nrd.nhtsa.dot.gov/departments/nrd-13/driver-distraction/PDF/2.PDF>

**References**

This study and others are linked from the web page <http://www.accidentreconstruction.com/research/cellphones/>.

**Examples**

```
data(reaction.time)
boxplot(time ~ control, data = reaction.time)
```

---

reddrum	<i>Growth of red drum</i>
---------	---------------------------

---

**Description**

Simulated length-at-age data for the red drum.

**Usage**

```
data(reddrum)
```

**Format**

A data frame with 100 observations on the following 2 variables.

**age** age

**length** a numeric vector

**Details**

This data is simulated from values reported in a paper by Porch, Wilson and Nieland titled "A new growth model for red drum (*Sciaenops ocellatus*) that accommodates seasonal and ontogenic changes in growth rates" which appeared in *Fishery Bulletin* 100(1) (<http://fishbull.noaa.gov/1001/por.pdf>). They attribute the data to Beckman et. al and say it comes from measurements in the Northern Gulf of Mexico, between September 1985 and October 1998.

**Examples**

```
data(reddrum)
plot(length ~ age, reddrum)
```

---

salmon.rate	<i>Simulated Data on Rate of Recruitment for Salmon</i>
-------------	---

---

**Description**

The Ricker model is used to model the relationship of recruitment of a salmon species versus the number of spawners. The model has two parameters, a rate of growth at small numbers and a decay rate at large numbers. This data set is simulated data for 83 different recordings using parameters found in a paper by Chen and Holtby.

**Usage**

```
data(salmon.rate)
```

**Format**

The format is: 83 numbers on decay rates.

**Details**

The Ricker model for recruitment modeled by spawner count

$$R_t = S_t e^{a-bS_t}$$

The parameter  $b$  is a decay rate for large values of  $S$ . In the paper by Chen and Holtby, they studied 83 datasets and found that  $b$  is log-normally distributed. The data is simulated from their values to illustrate a log normal distribution.

**Source**

These values are from D.G. Chen and L. Blair Holtby, "A regional meta-model for stock recruitment analysis using an empirical Bayesian approach", found at <http://www.iphc.washington.edu/>.

**Examples**

```
data(salmon.rate)
hist(log(salmon.rate))
```

---

salmonharvest

*Salmon harvest in Alaska from 1980 to 1998*

---

**Description**

A data set of unofficial tallies of salmon harvested in Alaska between the years 1980 and 1998. The units are in thousands of fish.

**Usage**

```
data(salmonharvest)
```

**Format**

A multiple time series object with yearly sampling for the five species Chinook, Sockeye, Coho, Pink, and Chum.

**Source**

This data was found at [http://seamarkets.alaska.edu/ak\\_harv\\_fish.htm](http://seamarkets.alaska.edu/ak_harv_fish.htm)

**Examples**

```
data(salmonharvest)
acf(salmonharvest)
```

---

samhda

*Substance Abuse and Mental Health Data for teens*

---

### Description

A data frame containing data on health behaviour for school-aged children.

### Usage

```
data(samhda)
```

### Format

A data frame with 600 observations on the following 9 variables.

**wt** A numeric weight used in sampling

**gender** 1=Male, 2=Female, 7=not recorded

**grade** 1 = 6th, 2 = 8th, 3 = 10th

**live.with.father** 1 = Y, 2 = N

**amt.smoke** Amount of days you smoked cigarettes in last 30. 1 = all 30, 2= 20-29, 3 = 10-19, 4 = 6-9, 5= 3-5, 6 = 1-2, 7=0

**alcohol** Have you ever drank alcohol, 1 = Y, 2 = N

**amt.alcohol** Number of days in last 30 in which you drank alcohol

**marijuana** Ever smoke marijuana. 1 = Y, 2= N

**amt.marijuana** Number of days in last 30 that marijuana was used. 1 = Never used, 2 = all 30, 3 = 20-29, 4 = 10-19, 5 = 6-9, 6 = 3-5, 7 = 1-2, 8 =Used, but not in last 30 days

### Details

A data frame containing data on health behaviour for school-aged children.

### Source

This data is sampled from the data set "Health Behavior in School-Aged Children, 1996: [United States]" collected by the World Health Organization, <http://www.icpsr.umich.edu/>. It is available at the Substance Abuse and Mental Health Data Archive (SAMHDA). Only complete cases are given.

### Examples

```
data(samhda)
attach(samhda)
table(amt.smoke)
```

---

scatter.with.hist      *Scatterplot with histograms*

---

**Description**

Draws a scatterplot of the data, and histogram in the margins. A trend line can be added, if desired.

**Usage**

```
scatter.with.hist(x, y,  
  hist.col = gray(0.95),  
  trend.line = "lm",  
  ...)
```

**Arguments**

x	numeric predictor
y	numeric response variables
hist.col	color for histogram
trend.line	Draw a trend line using <a href="#">lm</a> , <a href="#">supsmu</a> or <a href="#">lowess</a> . Use NULL for none.
...	Passed to <a href="#">plot</a> command for scatterplot

**Value**

Draws the graphic. No return value.

**Author(s)**

John Verzani

**References**

This example comes from the help page for [layout](#).

**See Also**

[layout](#)

**Examples**

```
data(emissions)  
attach(emissions)  
scatter.with.hist(perCapita,C02)
```

---

`scrabble`*Distribution of Scrabble pieces*

---

**Description**

Distribution and point values of letters in Scrabble.

**Usage**

```
data(scrabble)
```

**Format**

A data frame with 27 observations on the following 3 variables.

**piece** Which piece

**points** point value

**frequency** Number of pieces

**Details**

Scrabble is a popular board game based on forming words from the players' pieces. These consist of letters drawn from a pile at random. The game has a certain frequency of letters given by this data. These match fairly well with the letter distribution of the English language.

**Examples**

```
data(scrabble)
## perform chi-squared analysis on long string. Is it in English?
quote = " R is a language and environment for statistical computing \
and graphics. It is a GNU project which is similar to the S language \
and environment which was developed at Bell Laboratories (formerly \
AT&T, now Lucent Technologies) by John Chambers and colleagues. R \
can be considered as a different implementation of S. There are \
some important differences, but much code written for S runs \
unaltered under R."
quote.lc = tolower(quote)
quote = unlist(strsplit(quote.lc,""))
ltr.dist = sapply(c(letters," "),function(x) sum(quote == x))
chisq.test(ltr.dist, ,scrabble$freq)
```

---

sim *Function to simulate sampling distributions*

---

### Description

A convenient function for simulating sampling distributions

### Usage

```
sim(n = 10, m = 250, statistic = "mean", family = "norm", ...)
```

### Arguments

n	size of sample
m	number of simulations
statistic	character name of statistic. For example "mean" or "median".
family	family name of distribution such as "norm", "exp", "unif",...
...	named parameter values for the "r" function to find the random samples.

### Value

Returns a data vector of length m.

### Author(s)

John Verzani

### Examples

```
qqnorm(sim(n=10, family="exp"))
qqnorm(sim(n=10, family="t", df=3))
qqnorm(sim(n=10, family="t", statistic="median", df=3))
```

---

simple.chutes *simulate a chutes and ladder game*

---

### Description

This function will simulate a chutes and ladder game. It returns a trajectory for a single player. Optionally it can return the transition matrix which can be used to speed up the simulation.

### Usage

```
simple.chutes(sim=FALSE, return.cl=FALSE, cl=make.cl())
```

**Arguments**

sim                   Set to TRUE to return a trajectory.  
return.cl            Set to TRUE to return a transition matrix  
cl                    set to the chutes and ladders transition matrix

**Details**

To make a chutes and ladders trajectory

```
simple.chutes(sim=TRUE)
```

To return the game board

```
simple.chutes(return.cl=TRUE)
```

when doing a lot of simulations, it may be best to pass in the game board

```
cl <- simple.chutes(return.cl=TRUE) simple.chutes(sim=TRUE,cl)
```

**Value**

returns a trajectory as a vector, or a matrix if asked to return the transition matrix

**Author(s)**

John Verzani

**References**

board from <http://www.ahs.uwaterloo.ca/~musuem/vexhibit/Whitehill/snakes/snakes.gif>

**Examples**

```
plot(simple.chutes(sim=TRUE))
```

---

simple.densityplot     *Plots densities of data*

---

**Description**

Allows one to compare empirical densities of different distributions in a simple manner. The density is used as graphs with multiple histograms are too crowded. The usage is similar to side-by-side boxplots.

**Usage**

```
simple.densityplot(x, ...)
```

**Arguments**

x x may be a sequence of data vectors (eg. x,y,z), a data frame with numeric column vectors or a model formula

... You can pass in a bandwidth argument such as bw="SJ". See density for details. A legend will be placed for you automatically. To override the positioning set do.legend="manual". To skip the legend, set do.legend=FALSE.

**Value**

Makes a plot

**Author(s)**

John Verzani

**References**

Basically a modified boxplot function. As well it should be as it serves the same utility: comparing distributions.

**See Also**

[boxplot](#), [simple.violinplot](#), [density](#)

**Examples**

```
## taken from boxplot
## using a formula
data(InsectSprays)
simple.densityplot(count ~ spray, data = InsectSprays)
## on a matrix (data frame)
mat <- cbind(Uni05 = (1:100)/21, Norm = rnorm(100),
             T5 = rt(100, df = 5), Gam2 = rgamma(100, shape = 2))
simple.densityplot(data.frame(mat))
```

---

simple.eda

*Simple function to plot histogram, boxplot and normal plot*

---

**Description**

Simply plots histogram, boxplot and normal plot for experimental data analysis.

**Usage**

```
simple.eda(x)
```

**Arguments**

x                    a vector of data

**Value**

Just does the plots. No return value

**Author(s)**

John Verzani

**References**

Inspired by S-Plus documentation

**See Also**

hist,boxplot,qnorm

**Examples**

```
x<- rnorm(100,5,10)
simple.eda(x)
```

---

simple.eda.ts

*Makes 3 useful graphs for eda of times series*

---

**Description**

This makes 3 graphs to check for serial correlation in data. The graphs are a sequential plot ( $i$  vs  $X_i$ ), a lag plot (plotting  $X_i$  vs  $X_i$  where  $k=1$  by default) and an autocorrelation plot from the times series ("ts") package.

**Usage**

```
simple.eda.ts(x, lag=1)
```

**Arguments**

x                    a univariate vector of data

lag                  a lag to give to the lag plot

**Value**

Makes the graph with 1 row, 3 columns

**Author(s)**

John Verzani

**References**

see <http://www.itl.nist.gov/div898/handbook/eda/section3/eda34.htm> for more information on these and other plots for exploratory data analysis.

**Examples**

```
## The function is currently defined as

## look for no correlation
x <- rnorm(100);simple.eda.ts(x)
## you will find correlation here
simple.eda.ts(cumsum(x))
```

---

`simple.fancy.stripchart`

*Makes a fancier strip chart: plots means and a line*

---

**Description**

Not much, just hides some ugly code

**Usage**

```
simple.fancy.stripchart(l)
```

**Arguments**

l                    A list with each element to be plotted with a stripchart

**Value**

Creates the plot

**Author(s)**

John Verzani

**See Also**

`stripchart`

**Examples**

```
x = rnorm(10);y=rnorm(10,1)
simple.fancy.stripchart(list(x=x,y=y))
```

---

simple.freqpoly      *Simply plot histogram and frequency polygon*

---

**Description**

Simply plot histogram and frequency polygon. Students do not need to know how to add lines to a histogram, and how to extract values.

**Usage**

```
simple.freqpoly(x, ...)
```

**Arguments**

x	a vector of data
...	arguments passed onto histogram

**Value**

returns just the plot

**Author(s)**

John Verzani

**See Also**

hist,density

**Examples**

```
x <- rt(100,4)
simple.freqpoly(x)
```

---

simple.hist.and.boxplot  
*A function to plot both a histogram and a boxplot*

---

**Description**

Simple function to plot both histogram and boxplot to compare

**Usage**

```
simple.hist.and.boxplot(x, ...)
```

**Arguments**

x                    vector of univariate data  
...                   Arguments passed to the hist function

**Value**

Just prints the two graphs

**Author(s)**

John Verzani

**See Also**

hist, boxplot, layout

**Examples**

```
x<-rnorm(100)
simple.hist.and.boxplot(x)
```

---

simple.lag

*applies function to moving subsets of a data vector*

---

**Description**

Used to apply a function to subsets of a data vector. In particular, it is used to find moving averages over a certain "lag" period.

**Usage**

```
simple.lag(x, lag, FUN = mean)
```

**Arguments**

x                    a data vector  
lag                   the lag amount to use.  
FUN                   a function to apply to the lagged data. Defaults to mean

**Details**

The function FUN is applied to the data  $x[(i-lag):i]$  and assigned to the (i-lag)th component of the return vector. Useful for finding moving averages.

**Value**

returns a vector.

**Author(s)**

Provided to R help list by Martyn Plummer

**See Also**

filter

**Examples**

```
## find a moving average of the dow daily High
data(dowdata)
lag = 50; n = length(dowdata$High)
plot(simple.lag(dowdata$High,lag),type="l")
lines(dowdata$High[lag:n])
```

---

simple.lm

*Simplify usage of lm*

---

**Description**

Simplify usage of lm by avoiding model notation, drawing plot, drawing regression line, drawing confidence intervals.

**Usage**

```
simple.lm(x, y, show.residuals=FALSE, show.ci=FALSE, conf.level=0.95,pred=)
```

**Arguments**

x	The predictor variable
y	The response variable
show.residuals	set to TRUE to plot residuals
show.ci	set to TRUE to plot confidence intervals
conf.level	if show.ci=TRUE will plot these CI's at this level
pred	values of the x-variable for prediction

**Value**

returns plots and an instance of lm, as though it were called `lm(y ~ x)`

**Author(s)**

John Verzani

**See Also**

lm

**Examples**

```
## on simulated data
x<-1:10
y<-5*x + rnorm(10,0,1)
tmp<-simple.lm(x,y)
summary(tmp)

## predict values
simple.lm(x,y,pred=c(5,6,7))
```

---

simple.median.test      *Do simple sign test for median – no ranks*

---

**Description**

Do simple sign test like wilcox.test without ranking. Just computes two-sided p-value, no confidence interval is given.

**Usage**

```
simple.median.test(x, median=NA)
```

**Arguments**

x	A data vector
median	The value of median under the null hypothesis

**Details**

Unlike wilcox.test, this tests the null hypothesis that the median is specified against the two-sided alternative. For illustration purposes only.

**Value**

Returns the p value.

**Author(s)**

John Verzani

**See Also**

wilcox.test

**Examples**

```
x<-c(12,2,17,25,52,8,1,12)
simple.median.test(x,20)
```

---

simple.scatterplot      *Simple scatter plot of x versus y with histograms of each*

---

### Description

Shows scatterplot of x vs y with histograms of each on sides of graph. As in the example from layout.

### Usage

```
simple.scatterplot(x, y, ...)
```

### Arguments

x	data vector
y	data vector
...	passed to plot command

### Value

Returns the plot

### Author(s)

John Verzani

### See Also

layout

### Examples

```
x<-sort(rnorm(100))
y<-sort(rt(100,3))
simple.scatterplot(x,y)
```

---

`simple.sim`*Simplify the process of simulation*

---

### Description

'simple.sim' is intended to make it a little easier to do simulations with R. Instead of writing a for loop, or dealing with column or row sums, a student can use this "simpler" interface.

### Usage

```
simple.sim(no.samples, f, ...)
```

### Arguments

<code>no.samples</code>	How many samples do you wish to generate
<code>f</code>	A function which generates a single random number from some distributions. <code>simple.sim</code> generates the rest.
<code>...</code>	parameters passed to <code>f</code> . It does not like named parameters.

### Details

This is simply a wrapper for a for loop that uses the function `f` to create random numbers from some distribution.

### Value

returns a vector of size `no.samples`

### Note

There must be a 1000 better ways to do this. See [replicate](#) or [sapply](#) for example.

### Author(s)

John Verzani

### Examples

```
## First shows trivial (and very unnecessary usage)
## define a function f and then simulate
f<-function() rnorm(1) # create a single random real number
sim <- simple.sim(100,f) # create 100 random normal numbers
hist(sim)

## what does range look like?
f<- function (n,mu=0,sigma=1) {
  tmp <- rnorm(n,mu,sigma)
  max(tmp) - min(tmp)
}
```

```
}  
sim <- simple.sim(100,f,5)  
hist(sim)
```

---

simple.violinplot      *Plots violinplots instead of boxplots*

---

### Description

This function serves the same utility as side-by-side boxplots, only it provides more detail about the different distribution. It plots violinplots instead of boxplots. That is, instead of a box, it uses the density function to plot the density. For skewed distributions, the results look like "violins". Hence the name.

### Usage

```
simple.violinplot(x, ...)
```

### Arguments

x	Either a sequence of variable names, or a data frame, or a model formula
...	You can pass arguments to polygon with this. Notably, you can set the color to red with col='red', and a border color with border='blue'

### Value

Returns a plot.

### Author(s)

John Verzani

### References

This is really the boxplot function from R/base with some minor adjustments

### See Also

boxplot, simple.densityplot

**Examples**

```
## make a "violin"
x <- rnorm(100) ;x[101:150] <- rnorm(50,5)
simple.violinplot(x,col="brown")
f<-factor(rep(1:5,30))
## make a quintet. Note also choice of bandwidth
simple.violinplot(x~f,col="brown",bw="SJ")
```

---

`simple.z.test`*Implement basic z-test for illustrative purposes*

---

**Description**

Implements a z-test similar to the t.test function

**Usage**

```
simple.z.test(x, sigma, conf.level=0.95)
```

**Arguments**

x	A data vector
sigma	the known variance
conf.level	Confidence level for confidence interval

**Value**

Returns a confidence interval for the mean

**Author(s)**

Joh Verzani

**See Also**

t.test, prop.test

**Examples**

```
x<-rnorm(10,0,5)
simple.z.test(x,5)
```

---

skateranks

*Judges scores for disputed ice skating competition*

---

### Description

Judges scores from the disputed ice skating competition at the 2002 Winter olympics

### Usage

```
data(skateranks)
```

### Format

A data frame with 20 observations on the following 11 variables.

**Name** a factor with levels Berankova/Diabola Berezhnaya/Sikharulidze Bestnadigova/Bestandif Chuvaeva/Palamarchuk Cobisi/DePra Ina/Zimmerman Kautz/Jeschke Krasitseva/Znachkov Langlois/Archetto Lariviere/Faustino Pang/Tong Petrova/Tikhonov Ponomareva/SWviridov Savchenko/Morozov Scott/Dulebohn Sele/Pelletier Shen/Zhao Totmianina/Marinin Zagorska/Siudek Zhang/Zhang

**Country** a factor with levels Armenia Canada China Czech Germany Italy Poland Russia Slovakia US Ukraine Uzbekistan

**Russia** a numeric vector

**China** a numeric vector

**US** a numeric vector

**France** a numeric vector

**Poland** a numeric vector

**Canada** a numeric vector

**Ukraine** a numeric vector

**Germany** a numeric vector

**Japan** a numeric vector

### Examples

```
data(skateranks)
```

---

slc	<i>Sodium-Lithium countertransport</i>
-----	--

---

**Description**

Sodium-Lithium countertransport

**Usage**

```
data(slc)
```

**Format**

The format is: num [1:190] 0.467 0.430 0.192 0.192 0.293 ...

**Source**

From Kitchens' Exploring Statistics

**Examples**

```
data(slc)
hist(slc)
```

---

smokyph	<i>Water pH levels at 75 water samples in the Great Smoky Mountains</i>
---------	---

---

**Description**

Water pH levels at 75 water samples in the Great Smoky Mountains

**Usage**

```
data(smokyph)
```

**Format**

This data frame contains the following columns:

**waterph** a numeric vector

**elev** a numeric vector

**code** a numeric vector

**Source**

From Kitchens' Exploring Statistics

**Examples**

```
data(smokyph)
plot(smokyph$elev, smokyph$waterph)
```

---

south	<i>Murder rates for 30 Southern US cities</i>
-------	---

---

**Description**

Murder rates for 30 Southern US cities

**Usage**

```
data(south)
```

**Format**

The format is: num [1:30] 12 10 10 13 12 12 14 7 16 18 ...

**Source**

From Kitchens' Exploring Statistics

**Examples**

```
data(south)
hist(south)
```

---

southernosc	<i>Southern Oscillations</i>
-------------	------------------------------

---

**Description**

The southern oscillation is defined as the barometric pressure difference between Tahiti and the Darwin Islands at sea level. The southern oscillation is a predictor of el nino which in turn is thought to be a driver of world-wide weather. Specifically, repeated southern oscillation values less than -1 typically defines an el nino.

**Usage**

```
data(southernosc)
```

**Format**

The format is: Time-Series [1:456] from 1952 to 1990: -0.7 1.3 0.1 -0.9 0.8 1.6 1.7 1.4 1.4 1.5 ...

**Source**

<http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4412.htm>

**References**

See <http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4461.htm> for description.

**Examples**

```
data(southernosc)
plot(southernosc)
```

---

sp500.excess	<i>Excess returns of S&amp;P 500</i>
--------------	--------------------------------------

---

**Description**

Excess returns of S&P 500. These are defined as the difference between the series and some riskless asset.

**Usage**

```
data(sp500.excess)
```

**Format**

The format is: Time-Series [1:792] from 1929 to 1995: 0.0225 -0.044 -0.0591 0.0227 0.0077 0.0432 0.0455 0.0171 0.0229 -0.0313 ...

**Source**

This data set is used in Tsay, Analysis of Financial Time Series. It was downloaded from [www.gsb.uchicago.edu/fac/ruey.tsay/teaching/fts](http://www.gsb.uchicago.edu/fac/ruey.tsay/teaching/fts). The fSeries package may also contain this data set.

**Examples**

```
data(sp500.excess)
plot(sp500.excess)
```

---

`Split.zoo`*Add split method for zoo objects*

---

**Description**

Splits zoo objects by a grouping variable ala `split()`. Each univariate series is turned into a multivariate zoo object. If the original series is multivariate, the output is a list of multivariate zoo objects.

**Usage**

```
Split.zoo(x, f)
```

**Arguments**

<code>x</code>	an univariate or multivariate zoo object
<code>f</code>	A grouping variable of the same length of <code>x</code> . A warning is given if <code>length(f)</code> is not the same as index size of <code>x</code>

**Value**

Returns a multivariate zoo object, or list of such.

**Author(s)**

John Verzani

**See Also**

[split](#)

**Examples**

```
if(require(zoo)) {  
  split.zoo = Split.zoo ## make generic  
  x = zoo(1:30,1:30)  
  f = sample(letters[1:5],30, replace=TRUE)  
  split(x,f)  
}
```

---

`squareplot`*Create a squareplot alternative to a segmented barplot*

---

### Description

Create a squareplot as an alternative to a segmented barplot. Useful when the viewer is interested in exact counts in the categories. A squareplot is often used by the *New York Times*. A grid of squares is presented with each color representing a different category. The colors appear contiguously reading top to bottom, left to right. The colors segment the graph as a segmented bargraph, but the squares allow an interested reader to easily tally the counts.

### Usage

```
squareplot(x, col = gray(seq(0.5, 1, length = length(x))),
border =NULL, nrows = ceiling(sqrt(sum(x))), ncols =
ceiling(sum(x)/nrows),
...)
```

### Arguments

<code>x</code>	a vector of counts
<code>col</code>	a vector of colors
<code>border</code>	border color passed to <a href="#">polygon</a>
<code>nrows</code>	number of rows
<code>ncols</code>	number of columns
<code>...</code>	passed to title

### Value

Creates the graph, but has no return value.

### Author(s)

John Verzani

### References

The *New York Times*, [www.nytimes.com](http://www.nytimes.com). In particular, Sports page 6, June 15, 2003.

### Examples

```
## A Roger Clemens Cy Young year -- roids?
squareplot(c(21,7,6),col=c("blue","green","white"))
```

---

`stud.recs`*Student records*

---

**Description**

A simulation of student records used for placement purposes

**Usage**

```
data(stud.recs)
```

**Format**

A data frame with 160 observations on the following 6 variables.

**seq.1** Score on sequential 1 test

**seq.2** Score on sequential 2 test

**seq.3** Score on sequential 3 test

**sat.v** SAT verbal score

**sat.m** SAT math score

**num.grade** grade on first math class

**letter.grade** grade on first math class

**Details**

Some simulated student records for placement purposes

**Examples**

```
data(stud.recs)
hist(stud.recs$sat.v)
with(stud.recs, cor(sat.v, sat.m))
```

---

`student.expenses`*Some simulated data on student expenses*

---

**Description**

Some data for possible student expenses

**Usage**

```
data(student.expenses)
```

**Format**

A data frame of 5 variables for 10 students. All answers are coded "Y" for yes, "N" for no.

**cell.phone** Does student have cell phone.

**cable.tv** Does student have cable TV.

**dial.up** Does student pay for dial-up internet access.

**cable.modem** Does student pay for high-speed or cable modem access to internet.

**car** Does student own a car.

**Details**

Sample dataset of students expenses.

**Examples**

```
data(student.expenses)
attach(student.expenses)
table(dial.up, cable.modem)
```

---

superbarplot

*super segmented barplot*

---

**Description**

Plot a barplot, with bars nested and ranging from a max to a minimum value. A similar graphic is used on the weather page of the *New York Times*.

**Usage**

```
superbarplot(x, names = 1:dim(x)[2], names_height = NULL,
             col = gray(seq(0.8, 0.5, length = dim(x)[1]/2)), ...
             )
```

**Arguments**

x	A matrix with each pair of rows representing a min and max for the bar.
names	Place a name in each bar.
names_height	Where the names should go
col	What colors to use for the bars. There should be half as many specified as rows of x
...	passed to <code>plot.window</code> .

## Details

A similar graphic on the weather page of the *New York Times* shows bars for record highs and lows, normal highs and lows and actual (or predicted) highs or lows for 10 days of weather. This graphic succinctly and elegantly displays a wealth of information. Intended as an illustration of the [polygon](#) function.

## Value

Returns a plot, but no other values.

## Author(s)

John Verzani

## References

The weather page of the *New York Times*

## See Also

[squareplot](#)

## Examples

```
record.high=c(95,95,93,96,98,96,97,96,95,97)
record.low= c(49,47,48,51,49,48,52,51,49,52)
normal.high=c(78,78,78,79,79,79,79,80,80,80)
normal.low= c(62,62,62,63,63,63,64,64,64,64)
actual.high=c(80,78,80,68,83,83,73,75,77,81)
actual.low =c(62,65,66,58,69,63,59,58,59,60)
x=rbind(record.low,record.high,normal.low,normal.high,actual.low,actual.high)
the.names=c("S","M","T","W","T","F","S")[c(3:7,1:5)]
superbarplot(x,names=the.names)
```

---

tastesgreat

*Does new goo taste great?*

---

## Description

Fictitious data on taste test for new *goo*

## Usage

```
data(tastesgreat)
```

**Format**

A data frame with 40 observations on the following 3 variables.

**gender** a factor with levels Female Male

**age** a numeric vector

**enjoyed** 1 if enjoyed, 0 otherwise

**Details**

Fictitious data on a taste test with gender and age as covariates.

**Examples**

```
data(tastesgreat)
summary(glm(enjoyed ~ gender + age, data=tastesgreat, family=binomial))
```

---

tcm1y

*One-year treasury security values*

---

**Description**

The yields at constant fixed maturity have been constructed by the Treasury Department, based on the most actively traded marketable treasury securities.

**Usage**

```
data(tcm1y)
```

**Format**

The format is: Time-Series [1:558] from 1953 to 2000: 2.36 2.48 2.45 2.38 2.28 2.2 1.79 1.67 1.66 1.41 ...

**Source**

From the tcm data set in the tseries package. Given here for convenience only. They reference <http://www.federalreserve.gov/Releases/H15/data.htm>

**Examples**

```
data(tcm1y)
ar(diff(log(tcm1y)))
```

---

`tempsalinity`*Temperature/Salinity measurements along a moving Eddy*

---

### Description

Simulated measurements of temperature and salinity in the center of 'Eddy Juggernaut', a huge anti-cyclone (clockwise rotating) Loop Current Ring in the Gulf of Mexico. The start date is October 18, 1999.

### Usage

```
data(tempsalinity)
```

### Format

The data is stored as multivariate zooreg object with variables longitude, latitude, temperature (Celsius), and salinity (psu - practical salinity units <http://www.toptotop.org/climate/psu.php>).

### Details

The temperature salinity profile of body of water can be characteristic. This data shows a change in the profile in time as the eddy accumulates new water.

### Source

Data from simulation by Andrew Poje.

### Examples

```
data(tempsalinity)
if(require(zoo)) {
  plot(tempsalinity[,3:4])
  ## override plot.zoo method
  plot.default(tempsalinity[,3:4])
  abline(lm(salinity ~ temperature, tempsalinity, subset = 1:67))
  abline(lm(salinity ~ temperature, tempsalinity, subset = -(1:67)))
}
```

---

`too.young`*What age is too young for a male to date a female?*

---

**Description**

In U.S. culture, an older man dating a younger woman is not uncommon, but when the age difference becomes too great it may seem to some to be unacceptable. This data set is a survey of 10 people with their minimum age for an acceptable partner for a range of ages for the male. A surprising rule of thumb (in the sense that someone took the time to figure this out) for the minimum is half the age plus seven. Does this rule hold for this data set?

**Usage**

```
data(too.young)
```

**Format**

A data frame with 80 observations on the following 2 variables.

**Male** a numeric vector

**Female** a numeric vector

**Examples**

```
data(too.young)
lm(Female ~ Male, data=too.young)
```

---

`twins`*Burt's IQ data for twins*

---

**Description**

IQ data of Burt on identical twins that were separated near birth.

**Usage**

```
data(twins)
```

**Format**

A data frame with 27 observations on the following 3 variables.

**Foster** IQ for twin raised with foster parents

**Biological** IQ for twin raised with biological parents

**Social** Social status of biological parents

**Source**

This data comes from the R package that accompanies Julian Faraway's notes *Practical Regression and Anova in R* (now a book).

**Examples**

```
data(twins)
plot(Foster ~ Biological, twins)
```

---

u2                                      *Song and lengths for U2 albums*

---

**Description**

Song titles and lengths of U2 albums from 1980 to 1997.

**Usage**

```
data(u2)
```

**Format**

The data is stored as a list with names. Each list entry correspond to an album stored as a vector. The values of the vector are the song lengths in seconds and the names are the track titles.

**Source**

<http://www.u2station.com/u2ography.html>

**Examples**

```
data(u2)
sapply(u2,mean) # average track length
max(sapply(u2,max)) # longest track length
sort(unlist(u2)) # lengths in sorted order
```

---

urchin.growth	<i>Data on growth of sea urchins</i>
---------------	--------------------------------------

---

**Description**

Data on growth of sea urchins.

**Usage**

```
data(urchin.growth)
```

**Format**

A data frame with 250 observations on the following 2 variables.

**age** Estimated age of sea urchin

**size** Measurement of size

**Details**

Data is sampled from a data set that accompanies the thesis of P. Grosjean.

**Source**

Thesis was found at [http://www.sciviews.org/\\_pgrosjean](http://www.sciviews.org/_pgrosjean).

**Examples**

```
data(urchin.growth)
plot(jitter(size) ~ jitter(age), data=urchin.growth)
```

---

UsingR	<i>Opens browser to UsingR webpages</i>
--------	---

---

**Description**

Opens the browser to webpages from the UsingR website.

**Usage**

```
UsingR(what = c("webpage", "errata", "changes", "exercises"))
```

**Arguments**

**what** A character string indicating what page to open. No value specified will open the main webpage. Others are "errata" for typos and erros, "changes" for changes to the text brought about by new changes to R, and "exercises" to go to page containing selected answers.

**Value**

Opens the browser to the web page

**Author(s)**

John Verzani

**References**

see <http://www.math.csi.cuny.edu>

**See Also**

See Also [getAnswer](#)

**Examples**

```
## Not run: UsingR() # main webpage  
## Not run: UsingR("err") # errata
```

---

vacation

*vacation days*

---

**Description**

vacation days

**Usage**

```
data(vacation)
```

**Format**

The format is: num [1:35] 23 12 10 34 25 16 27 18 28 13 ...

**Source**

From Kitchens' Exploring Statistics

**Examples**

```
data(vacation)  
hist(vacation)
```

---

`violinplot`*Plots violinplots instead of boxplots*

---

**Description**

This function serves the same utility as side-by-side boxplots, only it provides more detail about the different distribution. It plots violinplots instead of boxplots. That is, instead of a box, it uses the density function to plot the density. For skewed distributions, the results look like "violins". Hence the name.

**Usage**

```
violinplot(x, ...)
```

**Arguments**

<code>x</code>	Either a sequence of variable names, or a data frame, or a model formula
<code>...</code>	You can pass arguments to polygon with this. Notably, you can set the color to red with <code>col='red'</code> , and a border color with <code>border='blue'</code>

**Value**

Returns a plot.

**Author(s)**

John Verzani

**References**

This is really the boxplot function from R/base with some minor adjustments

**See Also**

`boxplot`, `densityplot`

**Examples**

```
## make a "violin"
x <- rnorm(100) ; x[101:150] <- rnorm(50,5)
violinplot(x,col="brown")
f<-factor(rep(1:5,30))
## make a quintet. Note also choice of bandwidth
violinplot(x~f,col="brown",bw="SJ")
```

---

watertemp

*Temperature measurement of water at 85m depth*

---

### Description

Water temperature measurements at 10 minute intervals at a site off the East coast of the United States in the summer of 1974.

### Usage

```
data(watertemp)
```

### Format

A zoo class object with index stored as POSIXct elements. The measurements are in Celsius.

### Source

NODC Coastal Ocean Time Series Database Search Page <http://www.nodc.noaa.gov/dsdt/tsdb/search.html>

### Examples

```
if(require(zoo)) {  
  data(watertemp)  
  plot(watertemp)  
  acf(watertemp)  
  acf(diff(watertemp))  
}
```

---

yahoo.get.hist.quote

*Download stock data from Yahoo!*

---

### Description

Downloads stock data from Yahoo!

### Usage

```
yahoo.get.hist.quote(instrument = "^gspc", destfile = paste(instrument, ".csv", sep = ""), start, end,
```

**Arguments**

instrument	Ticker symbol as character string.
destfile	Temporary file for storage
start	Date to start. Specified as "2005-12-31"
end	Date to end
quote	Any/All of "Open", "High", "Low", "Close"
adjusted	Adjust for stock splits, dividends. Defaults to TRUE
download	Download the data
origin	Dates are recorded in the number of days since the origin. A value of "1970-01-01" is the default. This was changed from "1899-12-30".
compression	Passed to yahoo

**Details**

Goes to chart.yahoo.com and downloads the stock data. By default returns a multiple time series of class mts with missing days padded by NAs.

**Value**

A multiple time series with time measuring the number of days since the value specified to origin.

**Author(s)**

Daniel Herlemont <dherlemont@yats.com>

**References**

This function was found on the mailing list for R-SIG finance

**See Also**

yahoo.get.hist.quote in the tseries package

---

yellowfin

*Yellow fin tuna catch rate in Tropical Indian Ocean*

---

**Description**

Mean catch rate of yellow fin tuna in Tropical Indian Ocean for the given years.

**Usage**

```
data(yellowfin)
```

**Format**

A data frame with 49 observations on the following 2 variables.

**year** The year

**count** Mean number of fish per 100 hooks cast

**Details**

Estimates for the mean number of fish caught per 100 hooks are given for a number of years. This can be used to give an estimate for the size, or biomass, of the species during these years assuming the more abundant the fish, the larger the mean. In practice this assumption is viewed with a wide range of attitudes.

**Source**

This data is read from a graph that accompanies Myers RA, Worm B (2003) "Rapid worldwide depletion of predatory fish communities". *Nature* 423:280-283.

**References**

See also [http://www.soest.hawaii.edu/PFRP/large\\_pelagic\\_predators.html](http://www.soest.hawaii.edu/PFRP/large_pelagic_predators.html) for rebuttals to the Myers and Worm article.

**Examples**

```
data(yellowfin)
plot(yellowfin)
```

# Index

- \*Topic **aplot**
  - simple.hist.and.boxplot, 88
- \*Topic **datagen**
  - do, 34
  - simple.sim, 93
- \*Topic **datasets**
  - age.universe, 4
  - aid, 5
  - alaska.pipeline, 6
  - alltime.movies, 7
  - aosat, 7
  - arctic.oscillations, 8
  - babies, 9
  - babyboom, 10
  - batting, 11
  - baycheck, 12
  - best.times, 13
  - blood, 14
  - breakdown, 15
  - bright.stars, 15
  - brightness, 16
  - bumpers, 17
  - BushApproval, 17
  - bycatch, 18
  - cabinet, 19
  - camp, 20
  - cancer, 21
  - carbon, 22
  - carsafety, 22
  - central.park, 23
  - central.park.cloud, 24
  - cfb, 25
  - chicken, 26
  - chips, 26
  - co2emiss, 27
  - coins, 28
  - coldvermont, 28
  - corn, 29
  - crime, 30
  - deflection, 30
  - diamond, 32
  - divorce, 33
  - dottodot, 35
  - dowdata, 36
  - dvdsales, 37
  - emissions, 37
  - ewr, 38
  - exec.pay, 39
  - fat, 39
  - father.son, 41
  - female.inc, 42
  - firstchi, 43
  - five.yr.temperature, 43
  - florida, 44
  - galileo, 45
  - galton, 46
  - gap, 47
  - gasprices, 47
  - goalspergame, 49
  - google, 49
  - grades, 50
  - grip, 50
  - hall.fame, 51
  - healthy, 52
  - heartrate, 53
  - home, 54
  - homedata, 54
  - homeprice, 55
  - homework, 56
  - HUMMER, 56
  - iq, 57
  - kid.weights, 58
  - KSI, 59
  - last.tie, 59
  - lawsuits, 60
  - malpract, 61
  - mandms, 61
  - math, 62

- maydow, 63
- midsize, 64
- MLBattend, 64
- movies, 65
- mw.ages, 66
- nba.draft, 67
- normtemp, 67
- npdb, 68
- nym.2002, 70
- OBP, 71
- oral.lesion, 71
- ozonemonthly, 72
- paradise, 73
- pi2000, 73
- primes, 74
- puerto, 74
- rat, 76
- reaction.time, 77
- reddrum, 78
- salmon.rate, 78
- salmonharvest, 79
- samhda, 80
- scrabble, 82
- skateranks, 96
- slc, 97
- smokyph, 97
- south, 98
- southernosc, 98
- sp500.excess, 99
- stud.recs, 102
- student.expenses, 102
- tastesgreat, 104
- tcm1y, 105
- tempsalinity, 106
- too.young, 107
- twins, 107
- u2, 108
- urchin.growth, 109
- vacation, 110
- watertemp, 112
- yellowfin, 113
- \*Topic **hplot**
  - DOTplot, 35
  - scatter.with.hist, 81
  - simple.eda, 85
  - simple.fancy.stripchart, 87
  - simple.freqpoly, 88
  - simple.scatterplot, 92
  - squareplot, 101
  - superbarplot, 103
- \*Topic **htest**
  - simple.median.test, 91
- \*Topic **misc**
  - QQplot, 75
  - Split.zoo, 100
  - UsingR, 109
  - yahoo.get.hist.quote, 112
- \*Topic **multivariate**
  - DensityPlot, 31
  - scatter.with.hist, 81
  - simple.densityplot, 84
  - simple.violinplot, 94
  - superbarplot, 103
  - violinplot, 111
- \*Topic **regression**
  - simple.lm, 90
- \*Topic **ts**
  - simple.lag, 89
- \*Topic **univar**
  - sim, 83
  - simple.chutes, 83
  - simple.eda, 85
  - simple.eda.ts, 86
  - simple.freqpoly, 88
  - simple.hist.and.boxplot, 88
  - simple.lag, 89
  - simple.sim, 93
  - simple.z.test, 95
  - squareplot, 101
- \*Topic **utilities**
  - getAnswer, 48
- age.universe, 4
- aid, 5
- alaska.pipeline, 6
- alltime.movies, 7
- aosat, 7
- arctic.oscillations, 8
- babies, 9
- babyboom, 10
- batting, 11
- baycheck, 12
- best.times, 13
- blood, 14
- boxplot, 32, 85
- breakdown, 15

- bright.stars, 15
- brightness, 16
- bumpers, 17
- BushApproval, 17
- bycatch, 18
  
- cabinet, 19
- camp, 20
- cancer, 21
- carbon, 22
- carsafety, 22
- central.park, 23
- central.park.cloud, 24
- cfb, 25
- chicken, 26
- chips, 26
- co2emiss, 27
- coins, 28
- coldvermont, 28
- corn, 29
- crime, 30
  
- deflection, 30
- density, 32, 85
- DensityPlot, 31
- diamond, 32
- divorce, 33
- dnstyplt (simple.densityplot), 84
- do, 34
- DOTplot, 35
- dotplot, 35
- DOTplt (DOTplot), 35
- dottodot, 35
- dowdata, 36
- dvdsales, 37
  
- emissions, 37
- errata (getAnswer), 48
- ewr, 38
- exec.pay, 39
  
- fat, 39
- father.son, 41, 46
- female.inc, 42
- firstchi, 43
- five.yr.temperature, 43
- florida, 44
  
- galileo, 45
  
- galton, 46
- gap, 47
- gasprices, 47
- getAnswer, 48, 110
- goalspergame, 49
- google, 49
- grades, 50
- grip, 50
  
- hall.fame, 51
- healthy, 52
- heartrate, 53
- home, 54
- homedata, 54
- homeprice, 55
- homework, 56
- HUMMER, 56
  
- iq, 57
  
- kid.weights, 58
- KSI, 59
  
- last.tie, 59
- lawsuits, 60
- layout, 81
- lm, 81
- lowess, 81
  
- malpract, 61
- mandms, 61
- math, 62
- maydow, 63
- midsize, 64
- MLBattend, 64
- movies, 65
- mw.ages, 66
  
- nba.draft, 67
- normtemp, 67
- npdb, 68
- nym.2002, 70
  
- OBP, 71
- oral.lesion, 71
- ozonemonthly, 72
  
- paradise, 73
- pi2000, 73
- plot, 81

- plot.window, 103
- points, 36
- polygon, 101, 104
- primes, 74
- puerto, 74
  
- qqnorm, 76
- QQplot, 75
- qqplot, 76
  
- rat, 76
- reaction.time, 77
- reddrum, 78
- replicate, 34, 93
  
- salmon.rate, 78
- salmonharvest, 79
- samhda, 80
- sapply, 93
- scatter.with.hist, 81
- scrabble, 82
- sim, 83
- simple.chutes, 83
- simple.densityplot, 84
- simple.eda, 85
- simple.eda.ts, 86
- simple.fancy.stripchart, 87
- simple.freqpoly, 88
- simple.hist.and.boxplot, 88
- simple.lag, 89
- simple.lm, 90
- simple.median.test, 91
- simple.plot.hist.and.box  
    (simple.hist.and.boxplot), 88
- simple.scatterplot, 92
- simple.sim, 93
- simple.violinplot, 85, 94
- simple.z.test, 95
- skateranks, 96
- slc, 97
- smokyph, 97
- south, 98
- southernosc, 98
- sp500.excess, 99
- split, 100
- Split.zoo, 100
- squareplot, 101, 104
- stripchart, 35
- stud.recs, 102
  
- student.expenses, 102
- superbarplot, 103
- supsmu, 81
  
- tastesgreat, 104
- tcm1y, 105
- tempsalinity, 106
- text, 36
- too.young, 107
- twins, 107
  
- u2, 108
- urchin.growth, 109
- UsingR, 109
- usingr (UsingR), 109
  
- vacation, 110
- violinplot, 32, 111
- vlnplt (simple.violinplot), 94
  
- watertemp, 112
  
- yahoo.get.hist.quote, 112
- yellowfin, 113