Hands-on Slides

DR. TAN TECK KIANG
data()
?iris
iris
str(iris)
summary(iris)
attach(iris)
table(Species)
plot(Sepal.Length,Sepal.Width)
iris1 <- subset(iris,select = -Species)
pairs(iris1)
pairs(iris1,main="Scatter Matrix - pairs")
pairs(iris1,main="Scatter Matrix - pairs, gap=0",gap=0)

?pairs
nrow(available.packages())
install.packages("car")
library(car)
scatter3d(iris$Sepal.Length, iris$Petal.Length, iris$Sepal.Width)
M <- cor(iris1)
heatmap(x = M, margins = c(12, 12))
round(cor(iris1),2)
data()

- crimtab: Student's 3000 Criminals Data
- discoveries: Yearly Numbers of Important Discoveries
- esoph: Smoking, Alcohol and (O)esophageal Cancer
- euro: Conversion Rates of Euro Currencies
- euro.cross (euro): Conversion Rates of Euro Currencies
- eurodist: Distances Between European Cities and Between US Cities
- faithful: Old Faithful Geyser Data
- fdeaths (UKLungDeaths): Monthly Deaths from Lung Diseases in the UK
- freeny: Freny's Revenue Data
- freeny.x (freny): Freny's Revenue Data
- freeny.y (freny): Freny's Revenue Data
- infert: Infertility after Spontaneous and Induced Abortion
- iris: Edgar Anderson's Iris Data
- iris3: Edgar Anderson's Iris Data
- islands: Areas of the World's Major Landmasses
- ldeaths (UKLungDeaths): Monthly Deaths from Lung Diseases in the UK
- lh: Luteinizing Hormone in Blood Samples
- longley: Longley's Economic Regression Data
- lynx: Annual Canadian Lynx trappings 1821-1934
- mdeaths (UKLungDeaths): Monthly Deaths from Lung Diseases in the UK
- morley: Michelson Speed of Light Data
- mtcars: Motor Trend Car Road Tests
- nhtemp: Average Yearly Temperatures in New Haven
- nottem: Average Monthly Temperatures at Nottingham, 1920-1939
Edgar Anderson's Iris Data

Description

This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are *Iris setosa*, *versicolor*, and *virginica*.

Usage

?iris
iris
iris3

Format

iris is a data frame with 150 cases (rows) and 5 variables (columns) named Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, and Species.
## iris

> iris

<table>
<thead>
<tr>
<th>Sepal.Length</th>
<th>Sepal.Width</th>
<th>Petal.Length</th>
<th>Petal.Width</th>
<th>Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>3.5</td>
<td>1.4</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.9</td>
<td>3.0</td>
<td>1.4</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.7</td>
<td>3.2</td>
<td>1.3</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.6</td>
<td>3.1</td>
<td>1.5</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>5.0</td>
<td>3.6</td>
<td>1.4</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>5.4</td>
<td>3.9</td>
<td>1.7</td>
<td>0.4</td>
<td>setosa</td>
</tr>
<tr>
<td>4.6</td>
<td>3.4</td>
<td>1.4</td>
<td>0.3</td>
<td>setosa</td>
</tr>
<tr>
<td>5.0</td>
<td>3.4</td>
<td>1.5</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.4</td>
<td>2.9</td>
<td>1.4</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.9</td>
<td>3.1</td>
<td>1.5</td>
<td>0.1</td>
<td>setosa</td>
</tr>
<tr>
<td>5.4</td>
<td>3.7</td>
<td>1.5</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.8</td>
<td>3.4</td>
<td>1.6</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>4.8</td>
<td>3.0</td>
<td>1.4</td>
<td>0.1</td>
<td>setosa</td>
</tr>
<tr>
<td>4.3</td>
<td>3.0</td>
<td>1.1</td>
<td>0.1</td>
<td>setosa</td>
</tr>
<tr>
<td>5.8</td>
<td>4.0</td>
<td>1.2</td>
<td>0.2</td>
<td>setosa</td>
</tr>
<tr>
<td>5.7</td>
<td>4.4</td>
<td>1.5</td>
<td>0.4</td>
<td>setosa</td>
</tr>
<tr>
<td>5.4</td>
<td>3.9</td>
<td>1.3</td>
<td>0.4</td>
<td>setosa</td>
</tr>
<tr>
<td>5.1</td>
<td>3.5</td>
<td>1.4</td>
<td>0.3</td>
<td>setosa</td>
</tr>
<tr>
<td>5.7</td>
<td>3.8</td>
<td>1.7</td>
<td>0.3</td>
<td>setosa</td>
</tr>
<tr>
<td>5.1</td>
<td>3.8</td>
<td>1.5</td>
<td>0.3</td>
<td>setosa</td>
</tr>
</tbody>
</table>
# Data Structure

```r
str(iris)
```

> `str(iris)
'data.frame': 150 obs. of 5 variables:
$ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ... 
$ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ... 
$ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ... 
$ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ... 
$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 1 ...
```

# Summary Statistics

```r
summary(iris)
```

> `summary(iris)

    Sepal.Length  Sepal.Width  Petal.Length  Petal.Width  Species
     Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa :50
     1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
    Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica:50
     Mean   :5.843   Mean   :3.057   Mean   :4.358   Mean   :1.300
    3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
attach(iris)
table(Species)

> table(Species)
Species
setosa  versicolor  virginica
  50        50        50

# Scatter Plot
plot(Sepal.Length, Sepal.Width)
# Subset
iris1 <- subset(iris, select = -Species)

# Scatter Matrix
pairs(iris1, main="Scatter Matrix - pairs")
pairs(iris1, main="Scatter Matrix - pairs, gap=0", gap=0)
Scatterplot Matrices

Description
A matrix of scatterplots is produced.

Usage
pairs(x, ...)

## S3 method for class 'formula'
pairs(formula, data = NULL, ..., subset,
       na.action = stats::na.pass)

## Default S3 method:
pairs(x, labels, panel = points, ...,
       horInd = 1:nc, verInd = 1:nc,
       lower.panel = panel, upper.panel = panel,
       diag.panel = NULL, text.panel = textPanel,
       label.pos = 0.5 + has.diag/3, line.main = 3,
       cex.labels = NULL, font.labels = 1,
       row1lattro = TRUE, gap = 1, log = "",
       horOdd = !row1lattro, verOdd = !row1lattro)
?pairs

Arguments

x

the coordinates of points given as numeric columns of a matrix or data frame. Logical and factor columns are converted to numeric in the same way that `data.matrix` does.

formula

a formula, such as ~ x + y + z. Each term will give a separate variable in the pairs plot, so terms should be numeric vectors. (A response will be interpreted as another variable, but not treated specially, so it is confusing to use one.)

data

a data frame (or list) from which the variables in formula should be taken.

subset

an optional vector specifying a subset of observations to be used for plotting.

na.action

a function which indicates what should happen when the data contain NAs. The default is to pass missing values on to the panel functions, but `na.action = na.omit` will cause cases with missing values in any of the variables to be omitted entirely.

labels

the names of the variables.

panel

function(x, y, ...) which is used to plot the contents of each panel of the display.

... arguments to be passed to or from methods.

Also, graphical parameters can be given as can arguments to plot such as `main`. `par("oma")` will be set appropriately unless specified.
Details

The $j$th scatterplot contains $x[1,j]$ plotted against $x[1,j]$. The scatterplot can be customised by setting panel functions to appear as something completely different. The off-diagonal panel functions are passed the appropriate columns of $x$ as $x$ and $y$: the diagonal panel function (if any) is passed a single column, and the text.panel function is passed a single $(x, y)$ location and the column name. Setting some of these panel functions to `NULL` is equivalent to not drawing anything there.

The `graphical parameters` `pch` and `col` can be used to specify a vector of plotting symbols and colors to be used in the plots.

The `graphical parameter` `cex` will be set by `pairs.default` unless supplied as an argument.

A panel function should not attempt to start a new plot, but just plot within a given coordinate system: thus `plot` and `boxplot` are not panel functions.

By default, missing values are passed to the panel functions and will often be ignored within a panel. However, for the formula method and `na.action = na.omit`, all cases which contain a missing values for any of the variables are omitted completely (including when the scales are selected).

Arguments `horinz` and `verinz` were introduced in R 3.2.0. If given the same value they can be used to select or re-order variables, with different ranges of consecutive values they can be used to plot rectangular windows of a full pairs plot; in the latter case ‘diagonal’ refers to the diagonal of the full plot.

Author(s)

Enhancements for R 1.0.0 contributed by Dr. Jens Oehlschlaegel-Akiyoshi and R-core members.

References

Examples

pairs(iris[1:4], main = "Anderson's Iris Data -- 3 species",
   pch = 21, bg = c("red", "green3", "blue")[unclass(iris$Species)])

## formula method, "graph" layout (row 1 at bottom):
pairs(~ Fertility + Education + Catholic, data = swiss, rowlop=FALSE,
   subset = Education < 20, main = "Swiss data, Education < 20")

pairs(USSJudgeRatings, gap=1/10) # (gap: not wasting plotting area) # show only lower triangle (and suppress labeling for whatever reason):
pairs(USSJudgeRatings, text.panel = NULL, upper.panel = NULL)

## put histograms on the diagonal
panel.hist <- function(x, ...)
  {usr <- par("usr"); on.exit(par(usr))
   par(usr = c(usr[1:2], 0, 1.5))
   h <- hist(x, plot = FALSE)
   breaks <- h$breaks; nb <- length(breaks)
   y <- h$counts; y <- y/max(y)
   rect(breaks[-nb], 0, breaks[-1], y, col = "cyan", ...)
  }
pairs(USSJudgeRatings[1:5], panel = panel.smooth,
    cex = 1.5, pch = 24, bg = "light blue", hor=-odd, TRUE,
    diag.panel = panel.hist, cex.labels = 2, font.labels = 2)

## put (absolute) correlations on the upper panels,
## with size proportional to the correlations.
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...)
  {usr <- par("usr"); on.exit(par(usr))
   par(usr = c(0, 1, 0, 1))
   r <- abs(cor(x, y))
   txt <- format(c(r, 0.123456789), digits = digits)[1]
   txt <- paste0(prefix, txt)
   if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
   text(0.5, 0.5, txt, cex = cex.cor * r)
  }
pairs(USSJudgeRatings, lower.panel = panel.smooth, upper.panel = panel.cor,
    gap=0, rowlop=FALSE)

pairs(iris[-5], log = "xy") # plot all variables on log scale
pairs(iris, log - 1:4, # log the first four
  main = "Lengths and Widths in [log]", line.main=1.5, oma=c(2,2,3,2))
Contributed Packages

Available Packages

Currently, the CRAN package repository features 14781 available packages.

Table of available packages, sorted by date of publication

Table of available packages, sorted by name
library(car)
# Function scatter3d
# Three-Dimensional Scatterplots and Point Identification
scatter3d(iris$Sepal.Length, iris$Petal.Length, iris$Sepal.Width)
# Heatmap
M <- cor(iris1)
heatmap(x = M, margins = c(12, 12))
round(cor(iris1),2)