

RjpWiki アーカイブス

【R のヘルプ機能 (06.06.22)】

R の魅力の一つは充実した (オンライン) ヘルプ機能です。このチップスではそうした機能を紹介していきます。

1 基本 `help(foo)`, `?foo`

オブジェクト `foo` の `man` 形式のオンラインヘルプ

```
help(foo) # または簡略形 ?foo
```

パッケージ `MASS` 中のオブジェクト `parcoord` のヘルプをえる (当然パッケージ `MASS` が既にインストールされていることが前提)

```
help(parcoord, package=MASS)
```

演算子等は二重引用符で囲む必要がある。

```
help("for") # もしくは ?"for"  
?"+"
```

パッケージ "stepfun" 中のオブジェクト一覧

```
help(package="stepfun") # パッケージ "stepfun" 中のオブジェクト一覧 (パッケージ未ロードでも可)
```

データセットの解説

```
data() # 現在利用可能なすべてのデータセットの一覧  
?women # データセットの解説も同様に得られる
```

パッケージにある関数やデータセットに関するヘルプを表示する場合は以下のようにする。

```
data(package = "base") # パッケージ base で使用可能なデータセットを表示  
data(USArrests, "VADeaths") # データセット 'USArrests' and 'VADeaths' をロード  
help(USArrests) # データセット 'USArrests' に関するヘルプ
```

また、R のヘルプは英語なので日本人には読みにくい場合がある。手っ取り早く使い方が見たいという場合、例えば `solve` の使い方が知りたい場合は以下のようにすれば例を出力してくれる。

```
example(solve)
```

ただし、グラフ出力がある場合で、グラフをいくつか含む場合は、単に `example` をやってしまうと数枚のグラフが一瞬で表示されきってしまう。動体視力に自信のない方は、以下を定義しておくだけでグラフの描画ごとに逐一静止してくれる。

```
par(ask = TRUE)
```

2 キーワードによるオブジェクトの検索 `help.search("foo")`

キーワードの一部で良い。現在インストール済みでパッケージ検索パス中にある全パッケージが対象。オブジェクトドキュメントのタイトル、別名、キーワード等からの検索を指定できる。キーワードは文字列なので二重括弧で囲む必要。キーワードの一部で良い。現在インストール済みでパッケージ検索パス中にある全パッケージが対象。オブジェクトドキュメントのタイトル、別名、キーワード等からの検索を指定できる。

```
help.search("linear models") # 説明に単語 'linear models' を含むドキュメント
                             # の検索 (大小文字を区別せずに検索)
help.search("print")         # トピックやタイトルが 'print' を含むドキュメント
help.search(apropos = "print") # 上と同じ
help.search(keyword = "hplot") # キーワード "hplot" を持つドキュメント
help.search("\\btry", fields = "alias") # 別名の先頭が "try" で始まる
                                       # オブジェクト (正規表現使用)
```

3 latex 形式のヘルプファイルの見方

R およびアドオンパッケージをインストールすると適当なディレクトリに膨大なヘルプ文章が登録される。一つは `help()` 関数で表示される `man` 形式のドキュメントであるが、`latex` 形式のドキュメントも一揃い用意されている。これらは `Rd` 書式という独自のフォーマットで記述されたヘルプドキュメントから適当な Perl スクリプトでインストール時に生成される。`latex` 形式のドキュメントは最終的に `ps` ファイルに変換され印刷されることを想定している。

ソースから R システムを構築するとき `make install-dvi` とすれば `dvi` ファイルが一挙にできる。ソースファイル中にあるヘルプファイルの原型の `xxx.Rd` ファイルを個別に `R CMD Rd2dvi xxx.Rd` と処理すればやはり `dvi` ファイルが得られる。`help` 関数には `tex` 形式のヘルプファイルを要求に応じてその場で処理し、`ps` ファイルを生成するオプションがある。ヘルプ関数は途中で出来た `dvi` ファイルは抹消する仕様になっている。シェル環境変数 `'R.LATEXCMD'` と `'R.DVIPSCMD'` を指定しても良い。

```
options(latexcmd="/usr/bin/latex") # 環境変数
options(dvipscmd="/usr/bin/dvips")
help(optim, offline=TRUE)
```

```
Saving help page to 'optim.ps' # optim.tex から optim.ps が作られた
```

さらに次のようにすると `xdvi` によるオンライン表示が出来る (コンパイルに結構時間がかか

るのが難)。

```
options(dvipscmd="/usr/bin/xdvi")
```

さらに上で dvips を dvipdfm に変えると pdf が作成 (/tmp 下) される。さらに dvipsemd をだましてやると、dvi ファイルも保存できるが、dvi ファイルは中間ファイルで名前が変。例えば

```
cp ./*.dvi /tmp
```

というスクリプトファイル(名前を仮に makedvi)に実行属性をつけ(*unix OS の場合)た上で

```
options(dvipscmd="./makedvi")
```

としてやる。