

RjpWiki アーカイブス

【必要な関数を探すコツ】

1 自分の使用環境で探す

1.1 キーワード検索関数 `help.search()`

R の多様な関数を使いこなすには、見えそうな関数を要領よく探すコツが要ります。例えば「ベクトル場のグラフィックスを描きたい」とします。平面上の点から矢印を描けば良いわけですが、それに使えるどんな関数があるのか分からないとします。矢印は `arrow` ですから、先ず R のキーワード検索関数 `help.search()` を使い

```
> help.search("arrow")
```

とします。キーワードは二重引用符で囲む必要があります*1。すると

```
Help files with alias or title matching 'arrow' using fuzzy matching:
```

```
arrows(base)          Add Arrows to a Plot
```

```
Type 'help(FOO, package = PKG)' to inspect entry 'FOO(PKG) TITLE'.
```

と表示されます。これはキーワードをふくむ関数が `base` パッケージに一つあったことを意味しています。(検索は現在使用中の機械にインストールされているすべてのパッケージ中から行なわれますから、こういう時のためにとりあえず使う予定のない追加パッケージも手あたり次第インストールしておくともよいかも。) 次に関数 `arrows()` はどのようなものか感触を得るために、

```
> example(arrows)
```

で例示用コードを実行します。`par(ask=TRUE)` を先に一度実行しておくこと、複数のグラフィックス出力画面が一気に流れ去ることを防ぎ、画面を順にゆっくり点検することができます。見えそうだなとなったら

```
> ?arrows          # または help(arrows)   (二重引用符は不要)
```

で関数の説明と、最後に付いている例示用コードを眺めます。すると、`arrows()` は直前の高機能作図関数の出力に矢印を加える関数であること、引数としてベクトルが使えるので一つ一つ描く必要が無いことなどが分かります。また、同様の関数として矢印の代わりに線分を描く関数 `segments()` があることも分かります。

1.2 help.seach()

help.seach() は正規表現による曖昧な検索も受け付けます

```
> help.search("ab*cd")      # 前後にそれぞれ文字列 ab, cd を含む文字列を検索
```

1.3 オブジェクトのキーワード検索関数 apropos(), find()

help.search() に似ている関数に、ほとんど同じ機能をもつ apropos(), find() があります。これはキーワードを含むオブジェクトを検索します。キーワードは二重引用符で囲んでも、囲まなくても構いません。正規表現による検索も可能です（この際は二重引用符が必要）。違いは、apropos() は現在ロードされているライブラリ中から、キーワードを含むオブジェクトだけを検索対象とすることです。

```
> apropos("lm")             # lm を含むオブジェクトを検索
> apropos(ls)               # ls を含むオブジェクトを検索
> length(apropos("."))      # すべてのオブジェクトの総数
> apropos("^pr")           # pr で始まるすべてのオブジェクトの検索
> apropos("^..?$")         # 2文字のオブジェクトを検索
> apropos("pr$")           # pr で終るオブジェクトを検索
> apropos("ub.{0,}a")       # ub と a の間に零個以上の文字列を含むパターン
> apropos("ub.{1,}a")       # ub と a の間に1個以上の文字列を含むパターン
```

1.4 組み込み関数の一覧を得る builtins()

R 2.3.0 版で 1091 種類

```
> builtins()
 [1] "zip.file.extract"      "zapsmall"
 [3] "xpdrows.data.frame"   "xor"
 [5] "writeLines"           "writeChar"
    (途中省略)
[1087] "!="                    "!"
[1089] "-.POSIXt"             "-.Date"
[1091] "-"
```

1.5 インストールされているパッケージの一覧を得る `search()`

現在使用中の計算機にインストールされているパッケージの一覧を得るためには、次の命令を使います。

```
> search()
> searchpaths() # インストール場所の絶対パスを得る
```

1.6 あるライブラリ中のオブジェクトの一覧を見る `library(help=パッケージ名)`

もちろんそのライブラリが既にインストールされている必要があります。

```
> library(gstat) # ライブラリ gstat をロード
> library(help=gstat) # gstat 中のオブジェクト (関数、データセット) の一覧を得る
```

2 検索サイトで探す

2.1 R site search でキーワード検索

R site search : <http://finzi.psych.upenn.edu/search.html>

R 関連ドキュメントからの最大の全文検索システム。というコマンドもあります。

2.2 `RSiteSearch?()`

```
> RSiteSearch("BUGS") # ブラウザが起動し BUGS の検索結果を表示
```

いろいろなオプションがあるので、ヘルプを参照。ブラウザを変更するには、環境変数 `R_BROWSER` を変更する。

2.3 S-Plus/R の関数のカテゴリー化されたリスト

S-Plus/R Function Finder : <http://biostat.mc.vanderbilt.edu/s/finder/finder.html>

2.4 インターネット (Google) で検索する

R のマニュアルや公式メイリングリストの過去記事からキーワード検索するには、

- CRAN のホームページにある検索エンジンを使う
- google で検索先サイトを指定して検索

例えば google でキーワード rpart, NA を含む文書、記事を検索するには次のようなキーワードを使います。

site:r-project.org rpart NA また用例を探すには R のファイル形式を指定します。例えば plot の場合次のように google に入力します。拡張子が r という名前のファイルをネットで検索できます。

```
filetype:r plot
```

2.5 中間さんの CRAN code search by gonzui を試す

キーワードを含む R 関係のコードを検索してくれます。

CRAN code search by gonzui : <http://rgonzui.nakama.ne.jp/CRAN>