

Base R

チートシート

ヘルプを見る

ヘルプファイルの参照

?[mean](#)
関数のヘルプを見る
help.search('weighted mean')
単語やフレーズでヘルプを検索
help(package = 'dplyr')
パッケージのヘルプを見る

オブジェクトの詳細

```
str(iris)  
オブジェクト構造の概要を表示  
class(iris)  
オブジェクトの属するクラスを表示
```

ライブラリを使う

```
install.packages('dplyr')  
CRANからパッケージをダウンロードして  
インストール
```

```
library(dplyr)  
パッケージを読み込む
```

```
dplyr::select  
パッケージ内の特定の関数を使う
```

```
data(iris)  
組み込みデータセットを読み込む
```

作業ディレクトリ

```
getwd()  
現在の作業ディレクトリを表示  
(入力と出力はここで行われる)  
  
setwd('C://file/path')  
作業ディレクトリの変更
```

RStudioでは作業ディレクトリを現在いるディレクトリにセットするにはプロジェクトを用いる。

ベクトル

ベクトルの作成

c(2, 4, 6)	2 4 6	要素を結合してベクトルを作成
2:6	2 3 4 5 6	整数の並びからベクトルを作成
seq(2, 3, by=0.5)	2.0 2.5 3.0	連続する数の間隔を指定して作成
rep(1:2, times=3)	1 2 1 2 1 2	ベクトルの繰り返しから作成
rep(1:2, each=3)	1 1 1 2 2 2	要素の繰り返しからベクトルを作成

ベクトル関数

sort(x)	rev(x)
xをソートして返す	xを逆順にして返す
table(x)	unique(x)
各要素の個数を返す	ユニークな値を返す

ベクトルの要素の指定

位置による指定

x[4]	4番目の要素
x[-4]	4番目以外の要素
x[2:4]	2番目から4番目までの要素
x[-(2:4)]	「2番目から4番目まで」以外の要素
x[c(1, 5)]	1番目および5番目の要素

値による指定

x[x == 10]	値が10に等しい要素
x[x < 0]	値が0より小さい要素
x[x %in% c(1, 2, 5)]	1, 2, 5のセットに含まれる要素

名前による指定

x['apple']	値が'apple'に等しい要素
------------	-----------------

プログラミング

For 文

```
for (変数名 in シークエンス){  
  処理  
}
```

例

```
for (i in 1:4){  
  j <- i + 10  
  print(j)  
}
```

If 文

```
if (条件){  
  処理  
} else {  
  別の処理  
}
```

例

```
if (i > 3){  
  print('Yes')  
} else {  
  print('No')  
}
```

While 文

```
while (条件){  
  処理  
}
```

例

```
while (i < 5){  
  print(i)  
  i <- i + 1  
}
```

関数

```
関数名 <- function(変数){  
  処理  
  return(新しい変数)  
}
```

例

```
square <- function(x){  
  squared <- x*x  
  return(squared)  
}
```

ファイルの入出力

入力	出力	説明
df <- read.table('file.txt')	write.table(df, 'file.txt')	タブやコンマで区切られたテキストファイルからの読み込みおよび書き出し
df <- read.csv('file.csv')	write.csv(df, 'file.csv')	値がカンマで区切られたテキストファイルからの読み込みおよび書き出し(上記の特別な場合)
load('file.RData')	save(df, file = 'file.Rdata')	Rデータファイル(R独自のファイル形式)での読み込みおよび書き出し

条件判定

a == b	aとbが等しい	a > b	aがbより大きい	a >= b	aがb以上	is.na(a)	aが欠損値である
a != b	aとbが等しくない	a < b	aがbより小さい	a <= b	aがb以下	is.null(a)	aがnull値である

データ型

Rでよく使われるデータ型の間で相互に型を変換することができる。

as.logical	TRUE, FALSE, TRUE	ブール値への変換 (TRUE または FALSE)
as.numeric	1, 0, 1	整数または浮動小数点数に変換
as.character	'1', '0', '1'	文字列への変換。一般的には因子型に対して行われる。
as.factor	'1', '0', '1', levels: '1', '0'	因子型(レベルがセットされた文字列)への変換。統計モデルなどで用いられる。

数学関連の関数

log(x)	自然対数	sum(x)	合計値
exp(x)	指数関数	mean(x)	平均値
max(x)	最大値	median(x)	中央値
min(x)	最小値	quantile(x)	クォンタイル値
round(x, n)	整数値に丸める	rank(x)	要素の順位を返す
sig.fig(x, n)	有効数字n桁に丸める	var(x)	分散
cor(x, y)	相関係数	sd(x)	標準偏差

変数の代入

```
> a <- 'apple'
> a
[1] 'apple'
```

環境変数

ls()	環境内の全ての変数を表示
rm(x)	変数xを環境から削除する
rm(list = ls())	環境内の全ての変数を削除

RStudioでは環境パネルで環境内の変数を見ることができる。

行列

```
m <- matrix(x, nrow = 3, ncol = 3)
xから行列を作成する
```



m[2,] - 行を選択



m[, 1] - 列を選択



m[2, 3] - 要素を選択

t(m)

行と列を入れ替える

m %% n

行列の掛け算

solve(m, n)
m * x = nを満たすXを探す

リスト

```
l <- list(x = 1:5, y = c('a', 'b'))
異なるデータ型の要素をまとめて一つにしたもの
```

l[[2]]

リストlの2番目の要素

l[1]

1つのみの要素からなるリスト新規作成

l\$x

名前がxの要素

l['y']

名前がyの要素からなるリストを新規作成

dplyrパッケージ
ージ利用可

データフレーム

```
df <- data.frame(x = 1:3, y = c('a', 'b', 'c'))
データフレームはリストの特別なケースで、
すべての要素の長さが同じ場合
```

x	y
1	a
2	b
3	c

リストの一部指定

df\$x



df[[2]]



データフレームを理解する

View(df)

全データを見る

head(df)

先頭6行だけを見る

行列の一部指定

df[, 2]



df[2,]



df[2, 2]

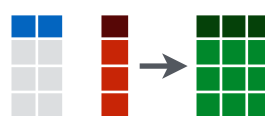


nrow(df)
行数を表示

ncol(df)
列数を表示

dim(df)
行数および列数を表示

cbind - 列と列を結合



rbind - 行と行を結合



文字列

stringrパッケージ利用可

paste(x, y, sep = ' ')	ベクトルを複数個つなげる
paste(x, collapse = ' ')	ベクトルの中の要素をつなげる
grep(pattern, x)	正規表現にマッチする文字列を探す
gsub(pattern, replace, x)	マッチする文字列を置換する
toupper(x)	大文字に変換
tolower(x)	小文字に変換
nchar(x)	文字列x中の文字数を表示

因子

factor(x)	cut(x, breaks = 4)
因子をベクトルに変換する。 因子のレベルと順序を指定できる。	数値ベクトルをいくつか に区切って因子に変換する。

統計

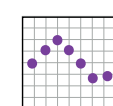
lm(x ~ y, data=df) 線形モデル	t.test(x, y) t検定	prop.test 2群の比率の 差の検定
glm(x ~ y, data=df) 一般化線形モデル	pairwise.t.test 対応のあるt検定	aov 分散分析
summary モデルの詳細を表示		

分布

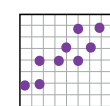
	ランダム変数	密度関数	累積分布	分位点
正規分布	rnorm	dnorm	pnorm	qnorm
ポアソン分布	rpois	dpois	ppois	qpois
二項分布	rbinom	dbinom	pbinom	qbinom
一様分布	runif	dunif	punif	qunif

プロット

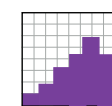
ggplot2パッケージ利用可



plot(x)
xの値を順に
プロット



plot(x, y)
xの値をyに対
してプロット



hist(x)
xのヒストグラ
ムをプロット

日付

lubridateパッケージ利用可