

ランダムウォーク

本ワークシートの内容

- ランダムウォークの生成および記述統計の解析

キーワード

- 乱数生成
- モンテカルロ法
- ランダムウォーク

対象

統計、統計力学、量子力学、数理ファイナンス、光学 など

▼ ランダムウォーク

正規分布に基づく平均値が0で標準偏差が1の乱数を20個生成します。

サンプルデータを生成するコマンドは統計 (Statistics) パッケージにあるので、はじめに統計パッケージを読み込みます。

```
with(Statistics) :
```

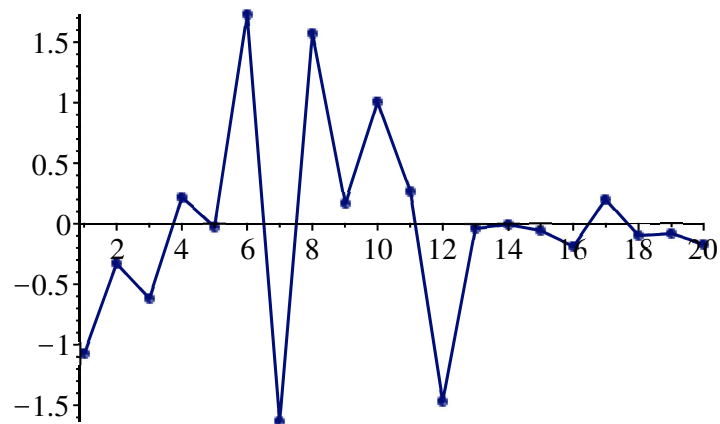
```
M := Sample(Normal(0, 1), [1, 20])
```

```
┌ 1 x 20 Matrix ────────────┐  
│ Data Type: float8 ────┐  
│ Storage: rectangular ───┐  
│ Order: Fortran_order ───┘  
└──────────────────────────┘
```

(1)

生成したデータを可視化します。

```
dataplot(M)
```



ランダムウォークの始点を 0 にするためにデータを一度リスト型に変換した後、リストの先頭に 0 を追加します。

```
M := convert(M, list) :
```

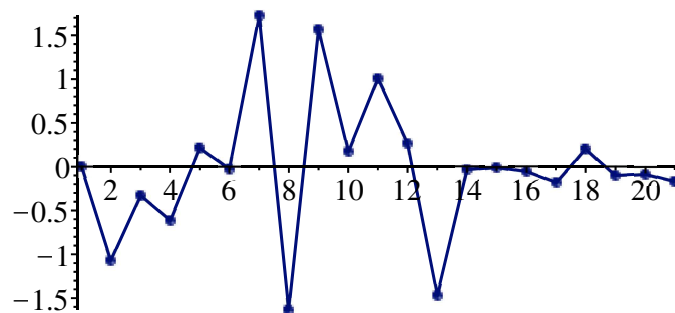
```
M := Join([[0], M])
```

```
1 .. 21 Array  
Data Type: anything  
Storage: rectangular  
Order: Fortran_order
```

(2)

データ (M) を可視化します。

```
dataplot(M)
```



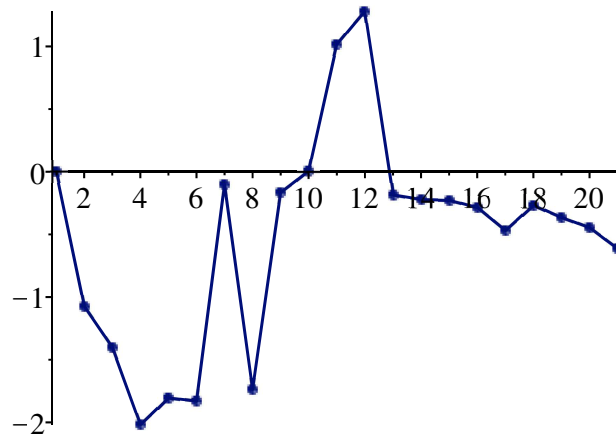
データの総和をもとめて、そのデータを可視化します。

```
S := CumulativeSum(M)
```

```
1 .. 21 Array  
Data Type: float_8  
Storage: rectangular  
Order: Fortran_order
```

(3)

`dataplot(S)`



ランダムウォークを生成するための関数を作成します。この関数によって、任意のデータ数を持つランダムウォークを簡単に生成することができます。

データ数が x のランダムウォークの関数は、以下のようになります。

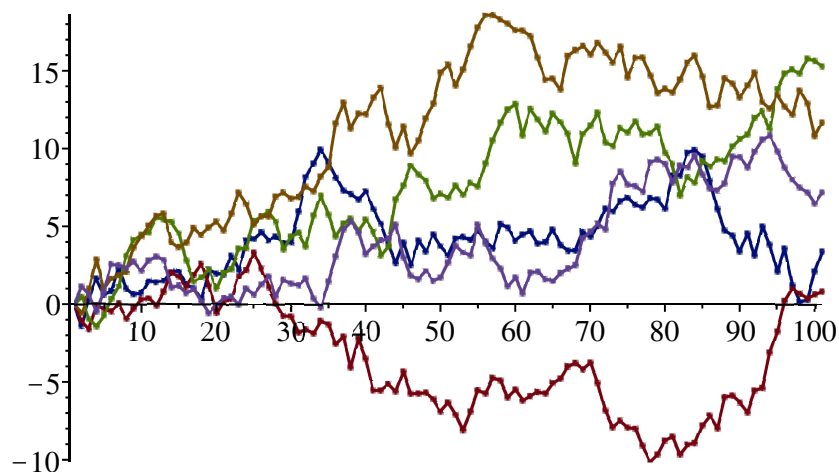
```
RandomWalk := x → CumulativeSum(Join([[0], [Sample(Normal(0, 1), [1, x])]])))  
x → Statistics:-CumulativeSum(Statistics:-Join([[0], [Statistics:-Sample(Normal(0, 1), [1, x])]]))) (4)  
RandomWalk(20)
```

<i>1 .. 21 Array</i>
<i>Data Type: float₈</i>
<i>Storage: rectangular</i>
<i>Order: Fortran_order</i>

(5)

データ数を100個もつランダムウォークのデータを5つ生成し、可視化します。

```
dataplot([seq(RandomWalk(100), i = 1..5)], symbolsize = 1)
```



以下は、ランダムウォークの最後の位置について、記述統計をおこないます。

データ数100のランダムウォークを1000生成します。

```
walks := seq(RandomWalk(100), i = 1..1000) :
```

生成したランダムウォークデータの一番最後の点を抽出します。

```
data := [seq(walks[k][-1], k = 1..1000)] :
```

記述統計データの表を作成します。

```
DataSummary(data, summarize = embed) :
```

	1
mean	-0.04145126409072342
standarddeviation	9.806242001955317
skewness	-0.07447118898564145
kurtosis	2.9598319785849805
minimum	-34.84442860042971
maximum	32.104741231656256
cumulativeweight	1000.0

▼ 主な利用コマンド

コマンド名	説明
• with (パッケージ名)	パッケージの読み込み 使用例: Statistics (統計) パッケージの読み込み with(Statistics) :
• Statistics[Sample] (確率変数または分布, 変数=範囲の下限 .. 上限)	乱数サンプルの生成
• dataplot (式, 変数=範囲の下限 .. 上限)	データセットのプロット
• convert (式, 変換名)	式を別の形式に変換
• Statistics[Join] (データセット, オプション)	データの結合
• 関数型演算子 →	関数の作成 使用例: $f(x, y) = x^2 + y^2$ の関数を作成する f := (x, y) -> x^2 + y^2

- `seq(任意の変数名を含む式、変数 = 範囲
の下限..上限)` 数列の生成
 - `Statistics[DataSummary](データセ
ット ,summarize=embed)` データセットに対する記述統計表の生成
-

無断転載禁止

Copyright © 2022 Maplesoft Japan Co., Ltd. All rights reserved.