

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])
```

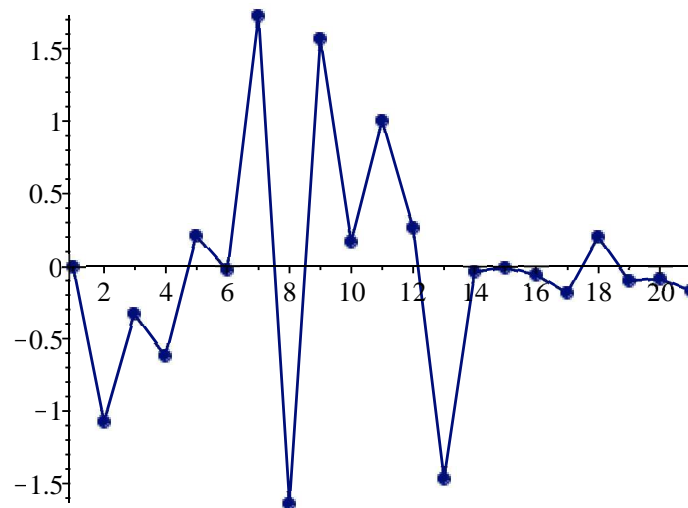
$M :=$   $\left[ \begin{array}{l} 1 \times 20 \text{ Matrix} \\ \text{Data Type: float}_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array} \right]$

(1)

出力をダブルクリックすると、M に格納されたデータを見ることができます。

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。

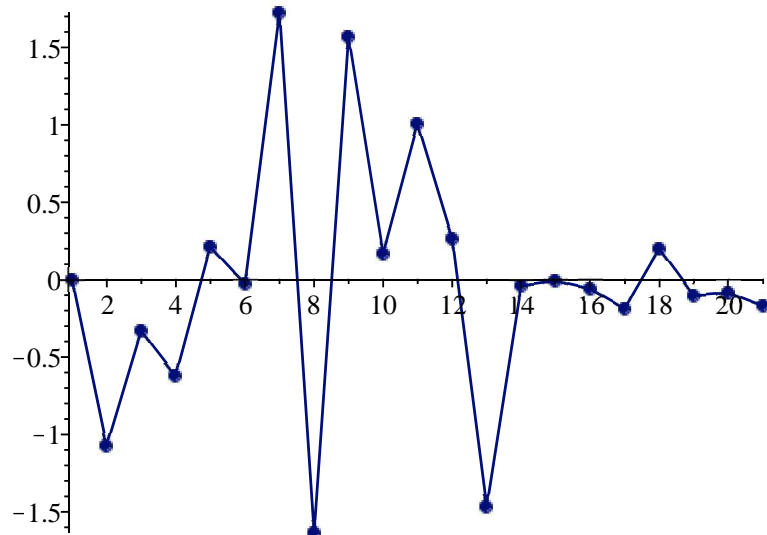
- ```
> M := convert(M, list):
> M := Join([[0], M])
```

$M :=$   $\left[ \begin{array}{l} 1 \dots 21 \text{ Array} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array} \right]$

(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。

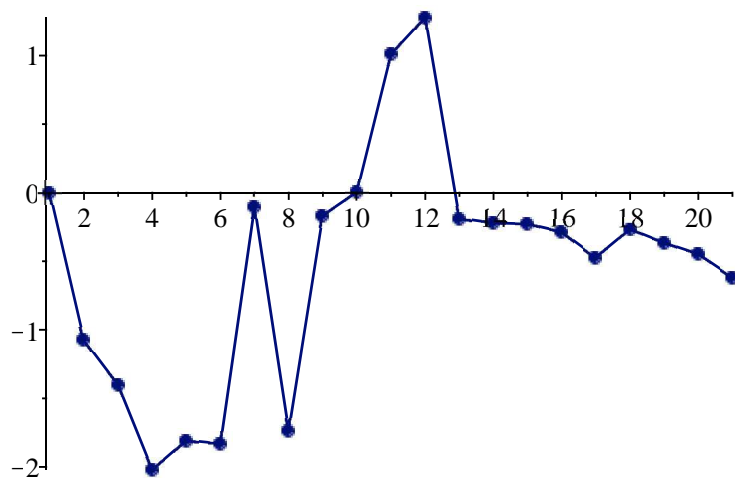
> `S := CumulativeSum(M)`

$S :=$ 

|                               |
|-------------------------------|
| 1 .. 21 Array                 |
| Data Type: float <sub>8</sub> |
| 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])]]))`  
`RandomWalk := x → Statistics:-CumulativeSum(Statistics:-Join([[0], [Statistics:-Sample(Normal(0, 1), [1, x])]]))`

(4)

> `RandomWalk(20)`

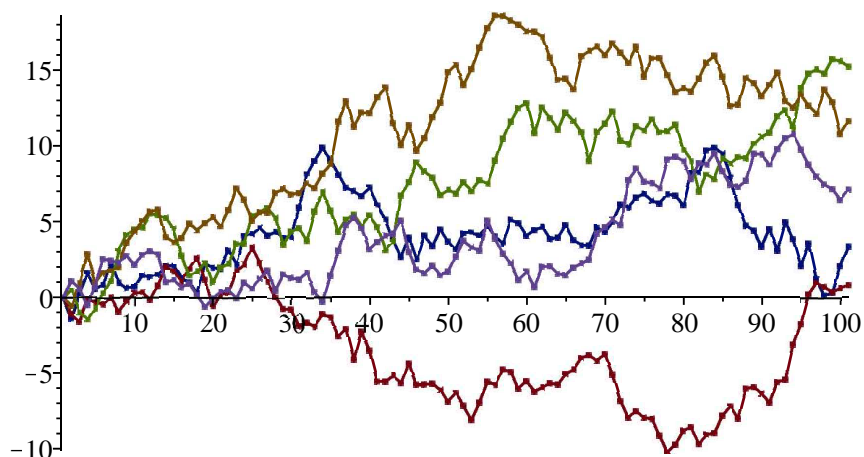
```

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

```

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                                                                                 |
|--------------------------------------------------------------------|------------------------------------------------------------------------------------|
| • <a href="#">with(パッケージ名)</a>                                     | パッケージの読み込み<br>使用例: Statistics (統計) パッケージの読み込み<br><b>with(Statistics) :</b>         |
| • <a href="#">Statistics[Sample]</a>                               | 乱数サンプルの生成                                                                          |
| • <a href="#">dataplot</a>                                         | データセットのプロット                                                                        |
| • <a href="#">convert</a>                                          | 式を別の形式に変換                                                                          |
| • <a href="#">Statistics[Join]</a>                                 | データの結合                                                                             |
| • <a href="#">関数型演算子</a>                                           | 関数の作成<br>使用例: $f(x, y) = x^2 + y^2$ の関数を作成する<br><b>f := (x, y) -&gt; x^2 + y^2</b> |
| • <a href="#">seq</a>                                              | 数列の生成                                                                              |
| • <a href="#">Statistics[DataSummary](データセット, summarize=embed)</a> | 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 © 2016 CYBERNET SYSTEMS CO., LTD. All rights reserved.