

个人资料



某小皮

访问：2554462次

积分：25044

等级：

BLOG

7

排名：第250名

原创：531篇

转载：30篇

译文：5篇

评论：242条

Welcome to 皮皮blog~

站长统计

+

-

1.01M visits  
REVOLVERMAPS

文章搜索

文章分类

-----Love Python----- (0)

Python (42)

Python类、函数和模块 (32)

Python拓展包 (6)

Numpy小记 (15)

Scipy小记 (8)

异步赠书：9月重磅新书升级，本本经典

程序员9月书讯

每周荐书：ES6、虚拟现实、物联网（评论送书）

numpy教程：随机数模块numpy.random

标签：

random

库

numpy

2014-09-24 00:14

32156人阅读

评论(0)

收藏

举报

分类：

Numpy小记 (14)

版权声明：本文为博主皮皮http://blog.csdn.net/pipisorry原创文章，未经博主允许不得转载。

目录(?)

[+]

http://blog.csdn.net/pipisorry/article/details/39508417

随机数种子

RandomState

RandomState exposes a number of methods for generating random numbers drawn from a variety of probability distributions.

使用示例

prng = np.random.RandomState(123456789) # 定义局部种子

prng.rand(2, 4)

prng.chisquare(1, size=(2, 2)) # 卡方分布

prng.standard\_t(1, size=(2, 3)) # t 分布

prng.poisson(5, size=10) # 泊松分布

[概率与统计分析]

[class numpy.random.RandomState]

random.seed()

random.seed(123456789) # 种子不同，产生的随机数序列也不同，随机数种子都是全局种子

要每次产生随机数相同就要设置种子，相同种子数的Random对象，相同次数生成的随机数字是完全相同的；

random.seed(1)

这样random.randint(0,6, (4,5))每次都产生一样的4\*5的随机矩阵

This method is called when RandomState is initialized. It can be called again to re-seed the generator.

[numpy.random.seed]

关于种子的介绍可参见[Java - 常用函数Random函数]

皮皮blog

numpy.random模块

linspace(start, end, num): 如linspace(0,1,11)结果为[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1];

arange(n): 产生一个从0到n-1的向量，如arange(4)结果为[0,1,2,3]

简单随机生成数据相关函数

rand(d0, d1, ..., dn)	Random values in a given shape.
randn(d0, d1, ..., dn)	Return a sample (or samples) from the "standard normal" distribution.
randint(low[, high, size, dtype])	Return random integers from low (inclusive) to high (exclusive).

Matplotlib小记 (8)
Pandas小记 (10)
Networkx小记 (2)
Sympy小记 (2)
Django小记 (14)
Crawler：Python爬虫小记 (8)
PythonQT小记 (5)
Scikit-Learn (16)
-----小小技术流----- (0)
主题模型Topic Model (19)
概率图模型PMG (21)
机器学习MachineLearning (38)
机器学习公开课 (19)
深度学习DeepLearning (2)
tensorflow (0)
分类 Classification (10)
聚类 Clustering (2)
海量数据挖掘MMDS (23)
Git小记 (10)
Spark (9)
Hadoop (6)
Docker (9)
Resys (9)
NLP (11)
BigData (4)
Dataming (1)
架构 (1)
Math (32)
-----Language World----- (0)
C++ (36)
Scala (11)
R (1)
Java (15)
Android (25)
Html (3)
Matlab (3)
assembly (3)
Perl (0)
Database (3)
Linux (41)
windows (14)
-----算法结构----- (0)
Algorithm&Datastructure (14)
大数据算法 (3)
编程之美 (9)
OJ (15)
逻辑之美 (2)
Coding (2)
-----OTH----- (10)
Office (11)
Internet (5)
inforSec (3)
paper (7)
mine (5)
others (4)

博客专栏

<code>random_integers(low[, high, size])</code>	Random integers of type np.int between <i>low</i> and <i>high</i> , inclusive.
<code>random_sample([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>random([size])</code>	Return random floats in the half-open interval [0.0, 1.0).产生随机矩阵，如 <code>random.random([2,3])</code> 产生一个2x3维的随机数
<code>ranf([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>sample([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>choice(a[, size, replace, p])</code>	Generates a random sample from a given 1-D array
<code>bytes(length)</code>	Return random bytes.

[Simple random data<sup>¶</sup>]

## np.random模块使用示例

### np.random.rand(a, b)

```
from numpy import random

x = random.rand(2, 3)
print(x)
[[ 0.1169922   0.08614147  0.17997144]
 [ 0.5694889   0.43067372  0.62135592]]

x, y = random.rand(2, 3)
print(x)
print(y)
[ 0.60527337  0.78765269  0.71884661]
[ 0.67420571  0.946359   0.7632273 ]
```

[numpy - 基本数据类型、多维数组ndarray及函数操作]

### np.random.randint(a, b, size=(c, d))

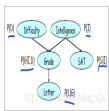
size : int or tuple of ints, optional

```
raw_user_item_mat = random.randint(0, 10, size=(3,4)) #指定生成随机数范围
和生成的多维数组大小
print(raw_user_item_mat)
[[3 6 2 8]
 [3 1 2 4]
 [9 4 5 0]]
```

[numpy.random.randint]  
[Random sampling (numpy.random)]

### 二项分布函数

np.random.binomial(n,p,size=N),函数的返回值表示n中成功的次数，且以 $C_n^x \cdot p^x \cdot (1-p)^{(n-x)}$ 的概率选择成功x次  
每一轮抛9枚硬币：  
outcome = np.random.binomial(9, 0.5, size=len(cash))  
[二项分布]  
超几何分布  
超几何分布是统计学上一种离散概率分布。它描述了由有限个物件中抽出n个物件，成功抽出指定种类的物件的次数（不归还）。  
在产品质量的不放回抽检中，若N件产品中有M件次品，抽检n件时所得次品数X=k，则 $P(X=k)=C(M, k) \cdot C(N-M, n-k)$



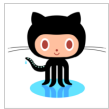
概率图模型  
文章：21篇  
阅读：117473



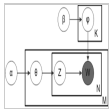
PythonQt  
文章：5篇  
阅读：37962



Django  
文章：13篇  
阅读：76548



Git  
文章：8篇  
阅读：45235



主题模型  
文章：16篇  
阅读：112674



机器学习  
文章：18篇  
阅读：56219

#### 最新评论

主题模型TopicModel: LDA编程3  
峰临子: 佩服!!!

pyinstaller使用-python项目转换为  
某小皮: @qq\_33901438:那就是  
input执行出错了? 可以试试将  
python代码转换成exe目录, 而...

ssh用法及命令  
刘雅雯\_Viola: SSH是一种网络协  
议

Docker教程: Docker镜像导出及  
weixin\_37744198: 博主有没有  
windows容器上运行的, 导出来  
的tar包? 我镜像下载不了, 想  
导入tar来作为基础镜...

深度学习: 前馈神经网络neural r  
暗夜J使者: 嘻嘻, 挺漂亮啊!

Pycharm中进行Python远程开发  
weixin\_39794492:  
@weixin\_39753205:我也遇到  
了, 你是怎么解决的?

机器学习算法及其损失函数  
某小皮: @wangfenghui132:可以  
的, 只要声明出处, 欢迎转载

机器学习算法及其损失函数  
wangfenghui132: 博主, 可以转  
载你的文章吗

EM算法原理详解  
qq\_38109275: 您您好, 我想请  
问Wij和Phi之间的区别, 思考了  
几天还是有点疑惑, 希望您能给  
我解答一下, 不胜感激。

python3爬虫 - cookie登录实战  
RunhwGuo: helped me!

#### 友情链接

Contact me

跳跳的博客

$k/C(N,n)$ ,  $C(a,b)$  为古典概型的组合形式,  $a$  为下限,  $b$  为上限, 此时我们称随机变量  $X$  服从超几何分布 (hypergeometric distribution)。

(1) 超几何分布的模型是不放回抽样。

(2) 超几何分布中的参数是  $M, N, n$  上述超几何分布记作  $X \sim H(N, n, M)$ 。

NumPy random模块中的hypergeometric函数可以模拟这种分布。

```
outcomes = np.random.hypergeometric(25, 1, 3, size=len(points))
```

#使用hypergeometric函数初始化游戏的结果矩阵。该函数的第一个参数为罐中普通球的数量, 第二个参数为“倒毒球”的数量, 第三个参数为每次采样 (摸球) 的数量。共进行超几何分布  $size$  次。返回  $size$  个抽样结果, 也就是普通球 (正品) 的数目。

[超几何分布]

## 连续分布

连续分布可以用PDF (Probability Density Function, 概率密度函数) 来描述。随机变量落在某一区间的概率密度函数在该区间的曲线下方的面积。

NumPy的random模块中有一系列连续分布的函数——beta、chisquare、exponential、f、gamma、t、laplace、lognormal、logistic、multivariate\_normal、noncentral\_chisquare、noncentral\_f、normal等绘制正态分布

随机数可以从正态分布中产生, 它们的直方图能够直观地刻画正态分布。

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

#使用NumPy random模块中的normal函数产生指定数量的随机数。

```
N=10000
```

```
normal_values = np.random.normal(size=N) #lz一般使用stats.norm.rvs(loc=0, scale=0.1, size=10)来生成高斯分布
```

随机数[Scipy教程 - 统计函数库scipy.stats]

#绘制分布直方图和理论上的概率密度函数 (均值为0、方差为1的正态分布) 曲线。

```
dummy, bins, dummy = plt.hist(normal_values, np.sqrt(N), normed=True, lw=1)
```

```
sigma = 1
```

```
mu = 0
```

```
plt.plot(bins, 1/(sigma * np.sqrt(2 * np.pi)) * np.exp(-(bins-mu)**2 / (2 * sigma**2)), lw=2) #lz提示, 也可以使用
```

scipy.stat.norm.pdf来生成非随机的高斯分布图[Scipy教程 - 统计函数库scipy.stats]

```
plt.show()
```

对数正态分布

```
np.random.lognormal(size=N)
```

皮皮blog

## random应用实例

从大小为  $n$  的原始样本集  $D$  中不放回地随机选取  $n1$  个样本点, 得到样本集  $D1$  和剩下的样本集  $D1\_left$ :

```
random_index = np.ones_like(class_labels, dtype=bool)
```

```
random_index[np.random.choice(range(len(data_arr)), n1, replace=False)] = False
```

```
D1 = data_arr[random_index]
```

```
D1_left = data_arr[~random_index]
```

from: <http://blog.csdn.net/pipisorry/article/details/39508417>

ref: Python模块: 生成随机数模块random

顶 踩

王老板儿的博客

少东的博客

志力的博客

pipi pages

4

1

上一篇 推荐系统：非个性化推荐Non-personalized recommendation

下一篇 面试逻辑题

相关文章推荐

- python-numpy一些方法总结
  - Python全栈工程师特训班-韦玮
  - numpy.random.seed()的使用
  - Blink在阿里集团的应用实践-陈守元
  - Hadoop：Hadoop单机伪分布式的安装和配置
  - Vue2.x知识点面面通
  - python数据分析(1)-numpy产生随机数
  - 大型Web构架设计案例解析
- numpy.random.uniform介绍
  - 机器学习案例实战-欺诈检测
  - np.random.RandomState、np.random.ra
  - Android开发实战30分钟集成第三方SDK
  - TensorFlow学习（五）：数学与随机值
  - numpy 随机数种类np.random.RandomState、...
  - numpy教程：随机数模块numpy.random
  - numpy的random模块

查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved

