引言：在机器学习还有深度学习中，经常会用到这几个函数，为了便于以后熟练使用，现在对这几个函数进行总结。

（一）np.random.rand()

该函数括号内的参数指定的是返回结果的形状，如果不指定，那么生成的是一个浮点型的数；如果指定一个数，那么生成的是一个numpy.ndarray类型的数组；如果指定两个数字，那么生成的是一个二维的numpy.ndarray类型的数组。如果是两个以上的数组，那么返回的维度就和指定的参数的数量个数一样。其返回结果中的每一个元素是服从0~1均匀分布的随机样本值，也就是返回的结果中的每一个元素值在0-1之间。

举例说明：

import numpy as np

mat = np.random.rand()

print(mat)

print(type(mat))

mat = np.random.rand(2)

print(mat)

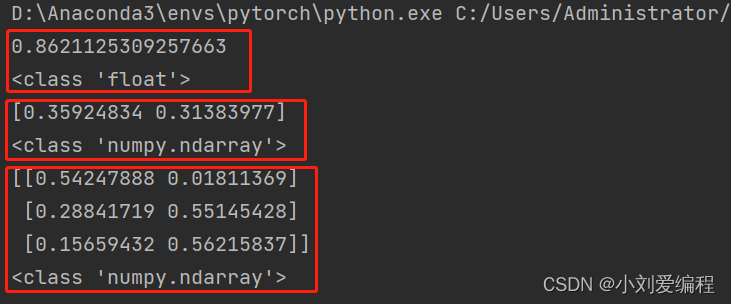
print(type(mat))

mat = np.random.rand(3, 2)

print(mat)

print(type(mat))

结果为：注意我用红框框起来的一组对应两个print输出，可对应程序看结果。



（二）np.random.randn()

该函数和rand()函数比较类似，只不过运用该函数之后返回的结果是服从均值为0，方差为1的标准正态分布，而不是局限在0-1之间，也可以为负值，因为标准正态分布的曲线是关于x轴对阵的。其括号内的参数如果不指定，那么生成的是一个浮点型的数；如果指定一个数，那么生成的是一个numpy.ndarray类型的数组；如果指定两个数字，那么生成的是一个二维的numpy.ndarray类型的数组。和rand()相比，除了元素值不一样，其他的性质是一样的。

举例说明：

import numpy as np

mat = np.random.randn()

print(mat)

print(type(mat))

mat = np.random.randn(2)

print(mat)

print(type(mat))

mat = np.random.randn(3, 2)

print(mat)

print(type(mat))

结果为：

图示

中度可信度描述已自动生成

（三）np.random.randint(low，high，size，dtype)

该函数中包含了几个参数，其具体含义为：

low：生成的元素值的最小值，即下限，如果没有指定high这个参数，则low为生成的元素值的最大值。

high：生成的元素值的最大值，即上限。

size：指定生成元素值的形状，也就是数组维度的大小。

dtype：指定生成的元素值的类型，如果不指定，默认为整数型

返回结果：返回值是一个大小为size的数组，如果指定了low和high这两个参数，那么生成的元素值的范围为[low,high)，不包括high；如果不指定high这个参数，则生成的元素值的范围为[0,low)。如果不指定size这个参数，那么生成的元素值的个数只有一个。

举例说明：

import numpy as np

# 指定一个参数low

mat = np.random.randint(low=1)

print(mat)

print(type(mat))

# 指定low和high，生成一个[low,high)的元素值

mat = np.random.randint(low=1, high=5)

print(mat)

print(type(mat))

# 指定size大小，生成一个三行三列的二维数组，元素个数为3x3=9个

mat = np.random.randint(low=2, high=10, size=(3, 3))

print(mat)

# 查看默认元素值的类型

print(type(mat[0][0]))

mat = np.random.randint(low=2, high=10, size=(3, 3), dtype=np.uint8)

print(mat)

print(type(mat[0][0]))

结果为：

文本

低可信度描述已自动生成

（四）np.random.uniform(low，high，size)

参数说明：

low：生成元素值的下界，float类型，默认值为0

high：生成元素值的上界，float类型，默认值为1

size：输出样本的数目，可以指定一个值，也可指指定大于等于两个值

返回对象：ndarray类型，形状为size中的数值指定，其元素个数为size指定的参数的乘积

我们前面已经说过了rand()这个函数，它返回的元素值是服从0-1的均匀分布，那如果不想要生成的是0-1范围内的均匀分布，想要其它范围内的均匀分布怎么办呢。

uniform()实现了这个功能，它可以生成服从指定范围内的均匀分布的元素。其返回值的元素类型为浮点型。需注意的是元素值的范围包含low，不包含high。

举例说明：

import numpy as np

# 指定一个参数low

mat = np.random.uniform()

print(mat)

print(type(mat))

# 指定low和high，生成一个[low,high)的元素值

mat = np.random.uniform(low=5, high=10)

print(mat)

print(type(mat))

# 指定size大小，生成一个三行三列的二维数组，元素个数为3x3=9个

mat = np.random.uniform(low=2, high=10, size=(3, 3))

print(mat)

# 查看默认元素值的类型

print(type(mat[0][0]))

mat = np.random.uniform(low=2, high=10, size=(3, 3, 2))

print(mat)

print(type(mat[0][0][0]))

结果为：

文本

描述已自动生成

总结：以上就是常用的随机数生成函数，具体用哪一个，可根据自己需求，想要生成什么随机数，那就使用什么样的函数。

编写不易，转载请注明出处！

————————————————

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

原文链接：https://blog.csdn.net/BaoITcore/article/details/125273828