欢迎继续我们的学习。

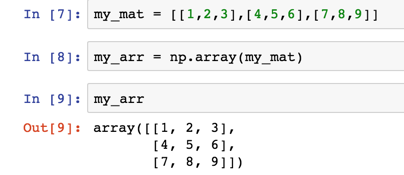
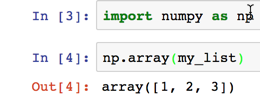
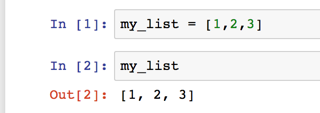
在上一课的内容中，我们简单了解了下什么是Numpy，以及如何安装配置Numpy。

在这一课的内容中，我们将正式介绍Numpy的数组。

首先打开Anaconda，并打开Jupyter Notebook，创建一个新的python3文件，将其命名为Numpy Arrays。

在代码区创建一个列表并输出：

在以上代码中，我们只是简单创建了一个列表，然后将其输出，相信大家都看得懂。



好了，接下来就可以把这个列表添加到numpy arrays中了。

在以上代码中，我们导入了numpy这个package,然后调用其array函数，将刚刚创建的列表当做参数，从而得到一个一维数组。

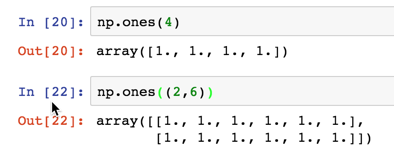
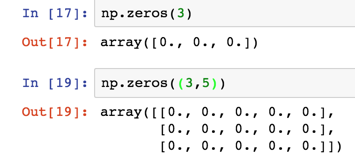
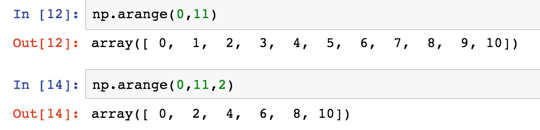
通过这种方式，我们将python中的普通列表转换成了数组。

接下来让我们尝试创建一个二维数组：

在以上代码中，我们首先创建了一个矩阵，然后使用numpy的array函数将其转换成二维数组。

接下来了解一下numpy的另外一个函数arange，使用arange可以快速生成所需的特定数组，比如：

在以上代码中，首先使用arange函数生成了一个从0到10的一维数组。



而接着用arange函数生成了另外一个从0到10的一维数组，但是每个数字之间的间隔是2.

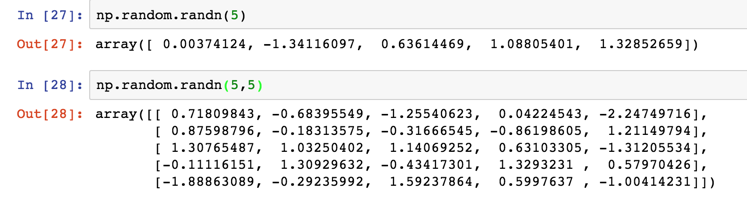
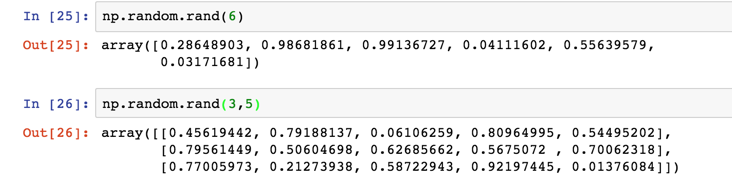
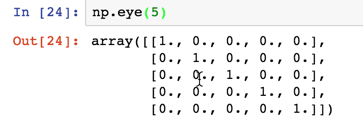
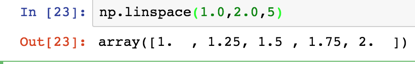
接着了解几个常用的函数。

使用zeros函数可以快速生成元素都是0的一维或多维数组（矩阵）。

使用ones函数可以快速生成元素都是1的一维或多维数组（矩阵）。

linspace函数跟arange类似，只不过其3个参数代表的是样本数量，而不是arange中的样本间距。

eye函数的作用是生成n阶的单位矩阵。也就是说除了对角线上的元素是1，其它元素都是0。

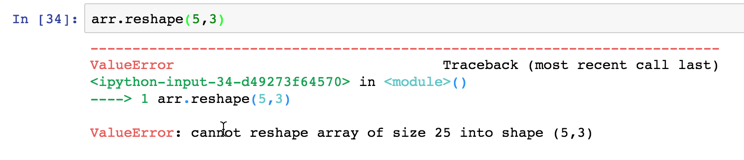
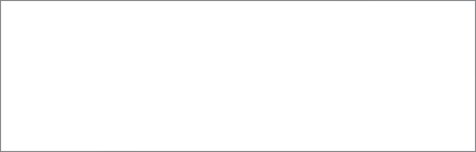
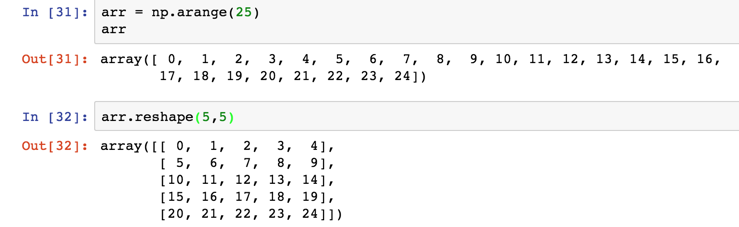
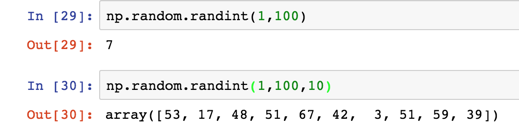


rand函数的作用是按照所输入的参数生成对应的随机数组或矩阵，其元素的数值在0到1之间。

以上的randn和rand函数的区别在于，randn的元素是满足正态分布的，均值为0，方差为1。

randint函数可以用来生成一个随机整数，前两个参数用来指定随机数所在的范围，最后一个参数用来指定随机数的数量，从而生成一维数组。

接下来再看另外一个很有用的函数reshape：



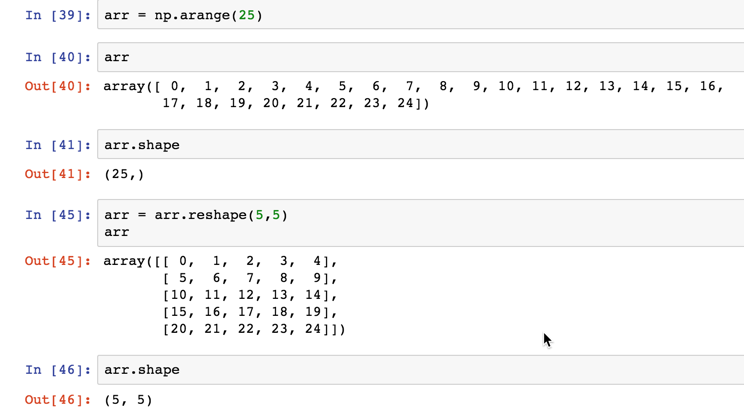
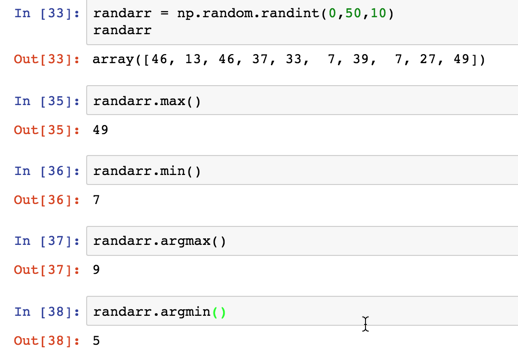
在以上代码中，我们首先使用arange函数生成了一个一维数组，然后使用reshape函数将其转换成一个5x5的矩阵。

需要注意的是，如果reshape中的参数之积不等于一维数组中的元素数量，那么就会报错：

再来看几个常用的函数：

max函数和min函数可以用来获取一维数组中的元素最大值和最小值。

而argmax函数和argmin函数则可以用来获取一维数组中最大和最小元素的位置（index序号）。



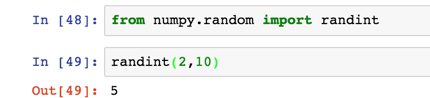
继续往下看：

在以上代码中，我们使用shape函数就可以了解数组的形态（比如一维数组的维度，或是矩阵的行数和列数）。

再来看看dtype函数：

使用该函数可以获取数组中元素的数据类型。

最后，在结束本课的学习之前，让我们了解一个小小的技巧。



可以看到，通过这种方式，我们就不需要每次都要输入np.random了。

另外还有一个小小的技巧，在jupyter notebook中，可以使用tab键进行函数名及变量名的自动补全。

好了，这一课的内容就到这里了，我们下一课再见~