

Part 3. Numpy 安装与介绍

在本节中，我将说明 Numpy 的安装与介绍。

课程视频

Numpy 安装

在虚拟环境中，可以运行下述命令安装Numpy。

```
pip install numpy
```

在 Conda环境中，可以运行下述命令安装Numpy。

```
conda install numpy
```

安装之后，运行命令检验是否安装成功。

```
python -c "import numpy as np;print(np.__version__)"
```

如果输出的版本是你期望的 Numpy 版本，说明安装成功。

Numpy 介绍

Numpy官网: <https://numpy.org/>

Numpy文档: <https://numpy.org/doc/stable/>

Numpy的github: <https://github.com/numpy/numpy>

NumPy (Numerical Python) 是 Python 中一个用于科学计算的库。

它提供了一个高效的多维数组对象（`ndarray`），并且支持大量的数学运算和数据操作。

NumPy 是许多数据分析、机器学习和科学计算库的基础。

功能展示

1. `import numpy as np` 说明

这使用 NumPy 时的标准导入方式。

`np` 是对 NumPy 库的常用别名，几乎所有使用 NumPy 的 Python 代码都会采用这种方式。

通过这种方式，你可以轻松调用 NumPy 中的函数和方法。

```
import numpy as np
```

2. `ndarray` (多维数组)

`ndarray` 是 NumPy 的核心数据结构，它是一个高效、固定大小的多维数组，支持批量的数学运算。

```
arr = np.array([1, 2, 3]) # 创建一个一维数组
print(arr)
```

```
[1 2 3]
```

3. 广播 (Broadcasting)

NumPy 支持数组形状不同但可以进行算术运算的功能，称为广播。

```
a = np.array([1, 2, 3])
b = np.array([10])
print(a + b) # 输出 [11, 12, 13]
```

```
[11 12 13]
```

4. 数学运算

NumPy 提供了大量的数学函数，可以对数组进行快速运算，包括加、减、乘、除等操作。

```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
print(np.add(a, b)) # 输出 [5 7 9]
```

```
[5 7 9]
```

5. 创建数组

NumPy 提供多种方式来创建数组，比如创建全 0 数组、全 1 数组或指定范围的数组。

```
np.zeros((2, 3)) # 创建一个 2x3 的零数组
```

```
array([[0., 0., 0.],
       [0., 0., 0.]])
```

```
np.ones((2, 3)) # 创建一个 2x3 的一数组
```

```
array([[1., 1., 1.],
       [1., 1., 1.]])
```

```
np.arange(0, 10, 2) # 创建一个从 0 到 10 步长为 2 的数组
```

```
array([0, 2, 4, 6, 8])
```

应用场景

NumPy 广泛应用于以下领域：

1. 数据分析与处理：

NumPy 提供的高效数组操作可以加速大规模数据的处理，特别适用于科学计算和数据分析。

2. 机器学习：

在机器学习中，NumPy 用于数据预处理、特征工程、计算模型的损失函数、梯度下降优化等任务。

3. 数值计算：

NumPy 可以用于解决线性代数问题、最优化问题、数值积分等。

4. 信号与图像处理：

NumPy 提供的数学函数和数组操作可用于信号处理、图像变换、滤波等应用。

总结

NumPy 是一个强大的工具，提供了高效的数组操作和大量数学函数，使得 Python 在处理科学计算、数据分析、机器学习等领域更加高效和便捷。