



Chapter.02 데이터 분석 라이브러리-01. Numpy를 사용하는 이유



Numpy : Numerical computing with Python. 수치연산 및 벡터 연산에 최적화된 라이브러리.

Numpy!

NumPy

""" To try the examples in the browser: 1. Type code in the input cell and press Shift + Enter to execute 2.

 <https://numpy.org/>

- 2005년에 만들어졌으며, 100% 오픈소스입니다.
- 최적화된 C code로 구현되어 있어 엄청나게 좋은 성능을 보입니다.
- 파이썬과 다르게 수치 연산의 안정성이 보장되어 있습니다. (**numerical stable**)
- N차원 실수값 연산에 최적화되어 있습니다. == N개의 실수로 이루어진 벡터 연산에 최적화되어 있습니다.

```
# numpy example
import numpy as np

arr = np.array([1, 2, 3])
print(np.linalg.norm(arr)) # print L2 norm of vector (1, 2, 3)
```

Numpy를 사용해야 하는 이유

1. 데이터는 벡터로 표현됩니다. 데이터 분석이란 벡터 연산입니다. 그러므로 벡터 연산을 잘 해야 데이터 분석을 잘할 수 있습니다.
2. (native) 파이썬은 수치 연산에 매우 약합니다. 실수값 연산에 오류가 생기면 (numerical error) 원하는 결과를 얻지 못할 수 있습니다. 많은 실수 연산이 요구되는 머신러닝에서 성능 저하로 이어질 수 있습니다.
3. numpy는 벡터 연산을 빠르게 처리하는 것에 최적화되어 있습니다. 파이썬 리스트로 구현했을 때보다 훨씬 더 높은 속도를 보여줍니다.

Hands-on

1. C언어로 코드를 구현하면, 파이썬으로 구현했을 때와 어떤 차이가 있는지 서술해보세요. (CPython)
2. 수치적 안정성(numerical stable)이 없으면 왜 성능 저하를 가져오는지 서술해보세요.
3. Numpy는 어떻게 벡터 연산을 빠르게 처리하는지 서술해보세요.