

[숙제10] NumPy 및 SciPy 연습

언어와 컴퓨터

2018년 11월 16일 금요일 13시까지

- 보고서 파일 hw10_00000.pdf를 제출하라.
- 보고서에는 코드 실행 결과와 간단한 설명이 있으면 된다.

13강(2018-11-07) 실습 코드 `vectors.py`(111행까지) 및 `statistics.py`(153행까지)에서는 NumPy와 SciPy를 사용한 코드가 주석으로 처리되어 있다. 이 코드의 주석 처리를 해제한 뒤, 실습 코드에서 직접 정의한 함수와 NumPy, SciPy에서 가져 온 함수의 실행 결과를 비교하고, 특이사항이 있으면 서술하라.

예 변수 `v`의 값을 리스트 `[1, 7, 8, 3]`, `w`를 `[3, 4, 3, 9]`라고 하자.

두 벡터의 합을 구하기 위해 실습 코드에서 `vector_add()` 함수를 정의하였다. `vector_add(v, w)`의 반환값은 리스트 `[4, 11, 11, 12]`이다. 벡터의 합은 NumPy에서 제공하는 함수 `add()`로도 구할 수 있다. `add(v, w)`의 반환값은 `array([4, 11, 11, 12])`로, 직접 정의한 함수로 구한 `vector_add(v, w)`와 네 성분의 값이 모두 같다.

```
>>> import numpy as np
>>>
>>> v = [1, 7, 8, 3]
>>> w = [3, 4, 3, 9]
>>> u = [0, 9, 2, 5]
>>>
>>> def vector_add(v, w):
...     """adds two vectors componentwise"""
...     return [v_i + w_i for v_i, w_i in zip(v,w)]
...
>>> vector_add(v, w)
[4, 11, 11, 12]
>>> np.add(v, w)
array([ 4, 11, 11, 12])
```

리스트 `v`와 `w`를 `numpy.ndarray` 자료형으로 변환하면, 아래와 같이 `add()` 함수 없이 덧셈 연산자로 벡터의 합을 구하는 것도 가능하다.

```
>>> type(vector_add(v, w)), type(np.add(v, w))
(<class 'list'>, <class 'numpy.ndarray'>)
>>> np.array(v) + np.array(w)
array([ 4, 11, 11, 12])
```