

STEP21. 연산자 오버로드 (2)

✎ (DeZero를 더 쉽게 사용하도록 개선하는 작업 필요)
 목표 : Variable 인스턴스와 ndarray 인스턴스, int, float 등도 함께 사용할 수 있게 함

ndarray와 함께 사용하기

Variable을 ndarray 인스턴스와 함께 사용

1. as_variable 함수 구현

```
def as_variable(obj):
    if isinstance(obj, Variable):
        return obj
    return Variable(obj)
```

```
import weakref

class Function:
    # *□□□ : 일의 개수의 인수 ( 가변길이 ) 를 건네 함수를 호출할 수 있음
    def __call__(self, *inputs):
        inputs = [as_variable(x) for x in inputs]
```

- 인수로 주어진 객체를 Variable 인스턴스로 변환해주는 함수
- 함수 인수 obj 가 Variable 인스턴스 또는 ndarray 인스턴스라고 가정

float, int와 함께 사용하기

파이썬의 자료형과 사용 가능하도록 개선

```
def add(x0, x1):
    x1 = as_array(x1)
    return Add()(x0, x1)
```

- 파이썬의 float와 int, np.float64와 np.int64 같은 타입과도 함께 사용 하도록 x가 Variable 인스턴스일 때 x + 3.0 같은 코드를 실행

as_array 함수 사용

```
y = x + 3.0
print(y)
```

- as_array 를 사용하면 x1이 float나 int 인 경우 ndarray 인스턴스로 변환될
- ndarray 인스턴스는 Function 클래스에서 Variable 인스턴스로 변환될

문제점 1: 첫 번째 인수가 float 나 int인 경우

```
[E] 0.3s Python
...
-----
TypeError                                Traceback (most recent call last)
Cell In[25], line 1
----> 1 y = 2.0 * x

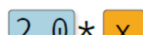
TypeError: unsupported operand type(s) for *: 'float' and 'Variable'
```

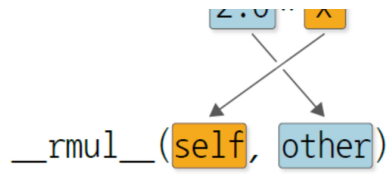
오류가 발생하는 과정

- 연산자 왼쪽에 있는 2.0의 `__mul__` 메서드를 호출 시도
- 2.0은 float 타입이므로 `__mul__` 메서드는 구현되어 있지 않음
- 다음은 * 연산자 오른쪽에 있는 x의 특수 메서드를 호출 시도
- x가 오른쪽에 있기 때문에 (`__mul__` 대신) `__rmul__` 메서드를 호출 시도
- Variable 인스턴스에는 `__rmul__` 메서드가 구현되어 있지 않음

해결 방법

그림 21-1 `__rmul__` 메서드로 인수가 전달되는 방식





- 이항 연산자의 경우 피연산자(항)의 위치에 따라 호출되는 특수 메서드가 다름

```
Variable.__add__ = add
Variable.__radd__ = add
Variable.__mul__ = mul
Variable.__rmul__ = mul
```

문제점 2: 좌항이 ndarray 인스턴스인 경우

```
x = Variable(np.array(2.0))
y = np.array([2.0]) + x
```

ndarray 인스턴스가 좌항이고, Variable 인스턴스가 우항인 경우

- 좌항인 ndarray 인스턴스의 `__add__` 메서드 호출
- 우항이 Variable 인스턴스의 `__radd__` 메서드가 호출되어야 함
- 연산자 우선순위를 지정해야 함
- Variable 인스턴스의 속성에 `__array_priority__` 추가하고 그 값을 큰 정수로 설정

```
class Variable:
    __array_priority__ = 200
    ...
```

- Variable 인스턴스의 연산자 우선순위를 ndarray 인스턴스의 연산자 우선순위보다 높일 수 있음
- 좌항이 ndarray 인스턴스라 해도 우항인 Variable 인스턴스의 연산자 메서드가 우선적으로 호출됨