

20.2 运算符重载

下面先重载乘法运算符*。乘法的特殊方法是`__mul__(self, other)`(参数`self`和`other`的相关内容将在后面解释)。如果定义(实现)了`__mul__`方法,那么在使用*进行计算时,`__mul__`方法就会被调用。下面试着实现`Variable`类的`__mul__`方法,代码如下所示。

```
Variable:
...
def __mul__(self, other):
    return mul(self, other)
```

上面的代码向已经实现的`Variable`类中添加了`__mul__`方法。这样在使用*进行计算时,被调用的就是`__mul__`方法,这个方法内部又会调用`mul`函数。下面我们用*运算符做一些计算。

```
a = Variable(np.array(3.0))
b = Variable(np.array(2.0))
y = a * b
print(y)
```

运行结果

variable(6.0)

上面的代码成功实现了 $y = a * b$ 的计算。当执行 $a * b$ 时,实例`a`的`__mul__(self, other)`方法被调用。这时,运算符*左侧的`a`作为`self`参数、右侧的`b`作为`other`参数传给了`__mul__`方法,具体如图20-2所示。