

## レポート用紙

講義名 : 数値解析 2	年月日 : 2025 年 9 月 26 日 ( 金 )
学籍番号 : 2318082	氏名 : 鈴木 祐亮

&lt;課題&gt;

```

# 2318082 鈴木 祐亮
# ex_diff.py: 順伝播と逆伝播
import numpy as np

# f1(x1, x2) = x1 + x2 class
class f1:
    def __init__(self):
        self.params = []
        self.grad = [1.0, 1.0]
        self.out = None

    # x1 + x2
    def forward(self, x):
        out = x[0] + x[1]    # x1 + x2
        self.out = out
        return out

    # df1/dx1, df1/dx2 = 1, 1
    def backward(self, dout):
        return self.grad

# f2(x1, x2) = x1 * x2 class
class f2:
    def __init__(self):
        self.params = []
        self.grad = [0.0, 0.0]
        self.out = None

    # x1 * x2
    def forward(self, x):
        out = x[0] * x[1]    # x1 * x2
        self.out = out
        return out

    # df1/dx1, df1/dx2 = x2, x1
    def backward(self, dout):
        self.grad = [dout[1], dout[0]]
        return self.grad

# メイン処理
# x = [2, 3]

# 初期化
func1 = f1()
func2 = f2()

# ① x = [-3, -2, -1]
# 順伝播
# F(x1, x2, x3) = f2(f1(x1, x2), x3)
x = np.array([-3, -2, -1])
ret1 = func1.forward(x)
x_new = [ret1, x[2]]
ret2 = func2.forward(x_new)
print(f'① F({x[0]}, {x[1]}, {x[2]}) = {ret2}')

# 逆伝播
# dF(x1, x2, x3)
x = np.array([-3, -2, -1])

```

## レポート用紙

```
diff2 = func2.backward([func1.forward(x), x[2]])
print('df2 = ', diff2)
diff1 = func1.backward(diff2)
print('df1 = ', diff1)
```

```
ret = [diff2[0] * diff1[0]] #  $\partial F / \partial x_1$ 
ret.append(diff2[0] * diff1[1]) #  $\partial F / \partial x_2$ 
ret.append(diff2[1]) #  $\partial F / \partial x_3$ 
print('dF(x1, x2, x3) = ', ret)
```

```
print()
# ② x = [-5, 7, 4]
# F(x1, x2, x3) = f2(f1(x1, x2), x3)
x = np.array([-5, 7, 4])
ret1 = func1.forward(x)
x_new = [ret1, x[2]]
ret2 = func2.forward(x_new)
print(f'② F({x[0]}, {x[1]}, {x[2]}) = ', ret2)
```

```
# 逆伝播
# dF(x1, x2, x3)
x = np.array([-5, 7, 4])
diff2 = func2.backward([func1.forward(x), x[2]])
print('df2 = ', diff2)
diff1 = func1.backward(diff2)
print('df1 = ', diff1)
```

```
ret = [diff2[0] * diff1[0]] #  $\partial F / \partial x_1$ 
ret.append(diff2[0] * diff1[1]) #  $\partial F / \partial x_2$ 
ret.append(diff2[1]) #  $\partial F / \partial x_3$ 
print('dF(x1, x2, x3) = ', ret)
```

```
● (myenv) suzukiyuusuke@suzukiyuuryounoMacBook-Air 数値解析2 % python3 02_ex_diff.py
① F(-3, -2, -1) = 5
df2 = [np.int64(-1), np.int64(-5)]
df1 = [1.0, 1.0]
dF(x1, x2, x3) = [np.float64(-1.0), np.float64(-1.0), np.int64(-5)]

② F(-5, 7, 4) = 8
df2 = [np.int64(4), np.int64(2)]
df1 = [1.0, 1.0]
dF(x1, x2, x3) = [np.float64(4.0), np.float64(4.0), np.int64(2)]
```