|  |  |
| --- | --- |
| 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 |

<課題>

# 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])

diff2 = func2.backward([func1.forward(x), x[2]])

print('df2 = ', diff2)

diff1 = func1.backward(diff2)

print('df1 = ', diff1)

ret = [diff2[0] \* diff1[0]] # ∂F/∂x1

ret.append(diff2[0] \* diff1[1]) # ∂F/∂x2

ret.append(diff2[1]) # ∂F/∂x3

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]] # ∂F/∂x1

ret.append(diff2[0] \* diff1[1]) # ∂F/∂x2

ret.append(diff2[1]) # ∂F/∂x3

print('dF(x1, x2, x3) = ', ret)

テキスト

自動的に生成された説明