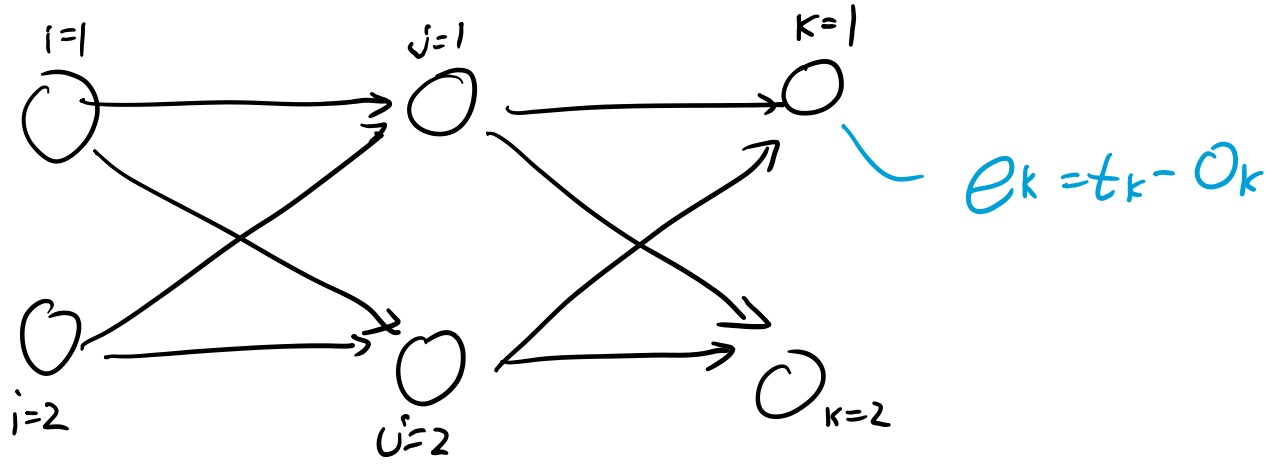
3/24(화) 신경망 backpropagation

2020년 3월 24일 화요일 오전 9:09



$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \cdot \sum_n (t_n - O_n)^2$$

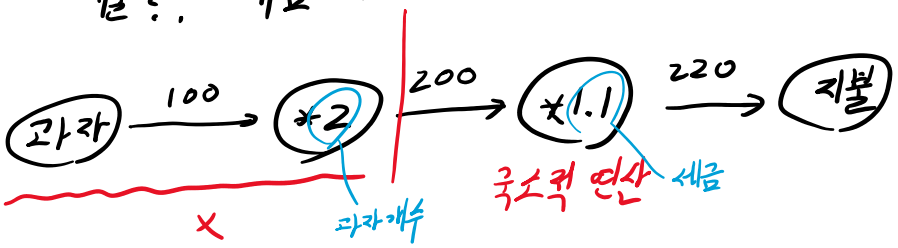
$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} (t_k - O_k)^2$$

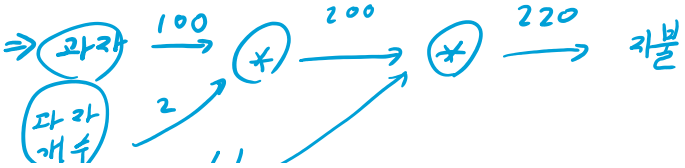
$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial O_k} \cdot \frac{\partial O_k}{\partial w_{jk}}$$

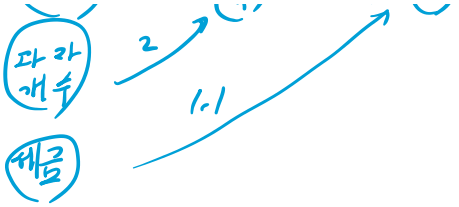
* 계산 그래프 : 계산을 그래프로 표현

ex) 길동은 1개 100원 과자를 2개 구매

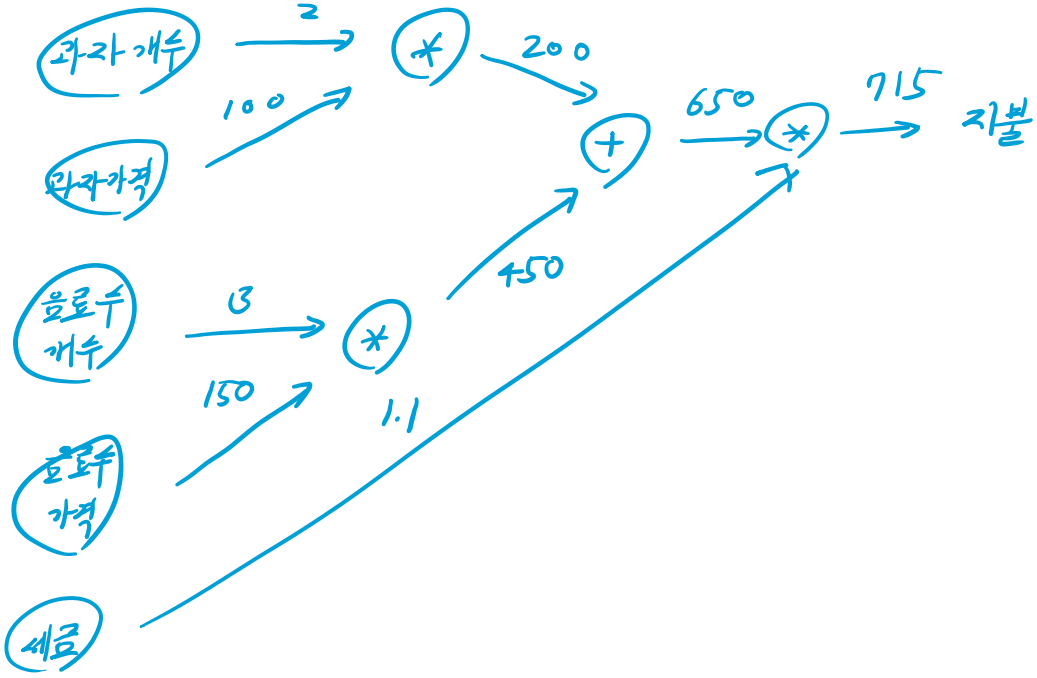
지불? 세금 10%







ex) 길동이가 과자를 2개, 음료수 3개, 과자는 1개 100원, 음료수 150원 세금 10%



과자가격 : x , 지불금액 : L $\Leftrightarrow \frac{\partial L}{\partial x}$

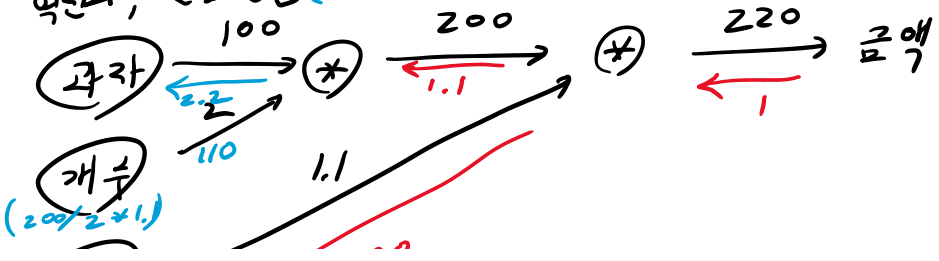
$f : wx+b$, $g = wx$

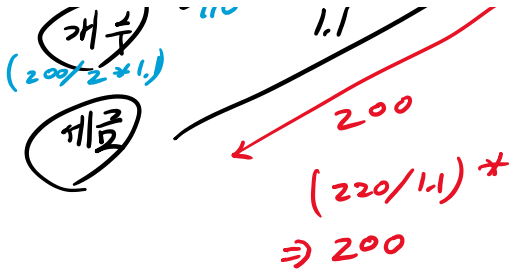
$f = g+b$

$\frac{\partial f}{\partial g} = 1$, $\frac{\partial f}{\partial b} = 1$

$\frac{\partial g}{\partial w} = x$, $\frac{\partial g}{\partial x} = w$

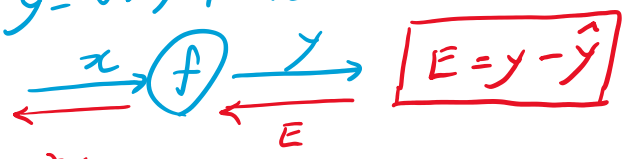
역전파, 미분값 전달 (200/100)*1.1 (220/200)*1





과자 가격에 대한 최종 계산 금액의 미분 값 ? 2.2

$y = f(x) = x^2$ 의 계산 그래프



$E \cdot \frac{\partial y}{\partial x}$

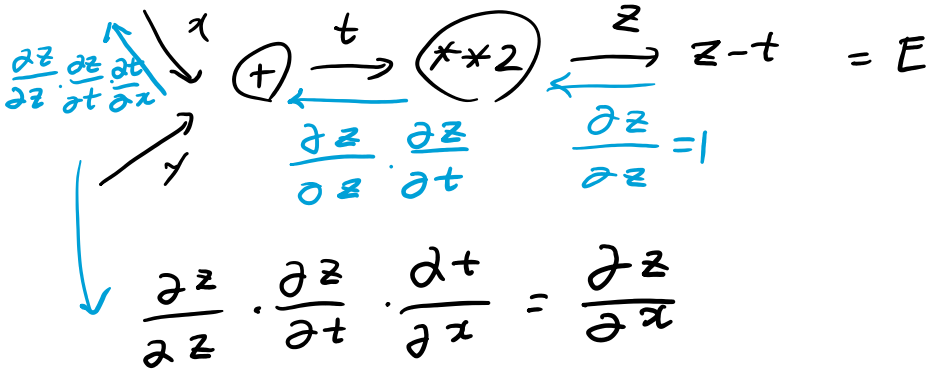
• 연쇄법칙 ? 합성함수 미분과 관련된 성질

$z = (x+y)^2$, $t = x+y$, $z = t^2$

※ 합성함수의 미분은 합성 함수를 구성하는 각 함수의 미분의 곱으로 표현된다.

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial t} \Big/ \frac{\partial t}{\partial x}$$

$2t = 2t \cdot 1 \Rightarrow 2(x+y)$

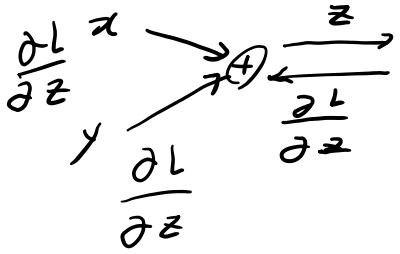


$$\frac{\partial z}{\partial z} \cdot \frac{\partial z}{\partial t} \cdot \frac{\partial t}{\partial x} = \frac{\partial z}{\partial x}$$

* 덧셈 노드의 역전파

$z = x+y$, $\frac{\partial z}{\partial x} = 1$, $\frac{\partial z}{\partial y} = 1$

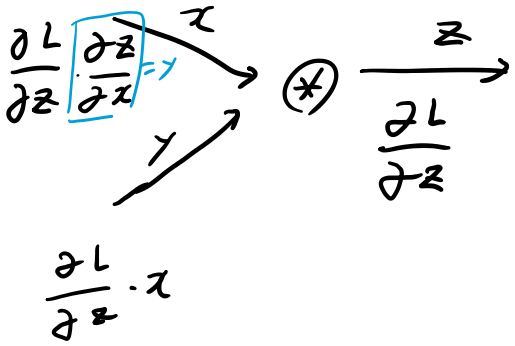
$z = x+y$, $\frac{\partial z}{\partial x} = 1$, $\frac{\partial z}{\partial y} = 1$



,

* 곱셈 노드의 역전파

ex) $z = xy \Rightarrow \frac{\partial z}{\partial x} = y$, $\frac{\partial z}{\partial y} = x$



$\frac{\partial L}{\partial z} \cdot x$

```
snack=100
snack_num=2
tax=1.1
```

```
class BackTest:
    def __init__(self):
        self.x=None
        self.y=None
        print("초기화 함수")
    def forward(self,x,y):
        self.x=x
        self.y=y
        out=x*y
        return out
    def backward(self,dout):
        dx=dout*self.y#구문
        dy=dout*self.x#구문
        return dx,dy#
```

```
mul_snack_layer=BackTest()#객체 생성->생성자(초기화) 함수 호출
```

초기화 함수

```
snack_price=mul_snack_layer.forward(snack,snack_num)
print(snack_price)
```

```
200
```

```
mul_tax_layer=BackTest()
price=mul_tax_layer.forward(snack_price,tax)
print(price)
```

```
초기화 함수
220.00000000000003
```

```
dprice=1
dsnack_price,dtax=mul_tax_layer.backward(dprice)
print(dsnack_price,dtax)#1.1, 200v
dsnack,dsnack_num=mul_snack_layer.backward(dsnack_price)
print(dsnack,dsnack_num)
```

```
1.1 200
2.2 110.00000000000001
```