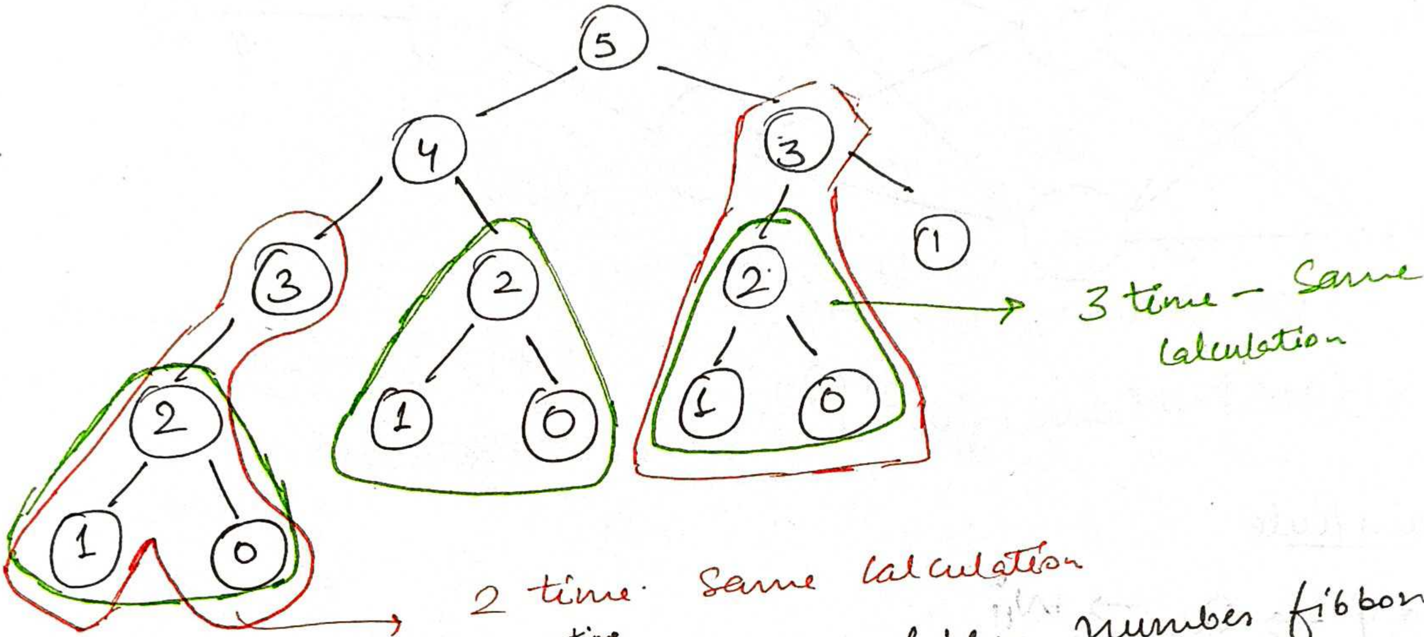
Memoization

"Memoization is an optimization technique used primarily to speedup computer programs by storing the result of expensive function calls and returning the cached result when the same inputs occur again."

ex: fibbonaci series



So, this take more time to solve higher number fibbonaci series. Bcz these method solve same calculation multiple time and this method take ↑↑ time to solve.

for eg:- fibbonaci series code

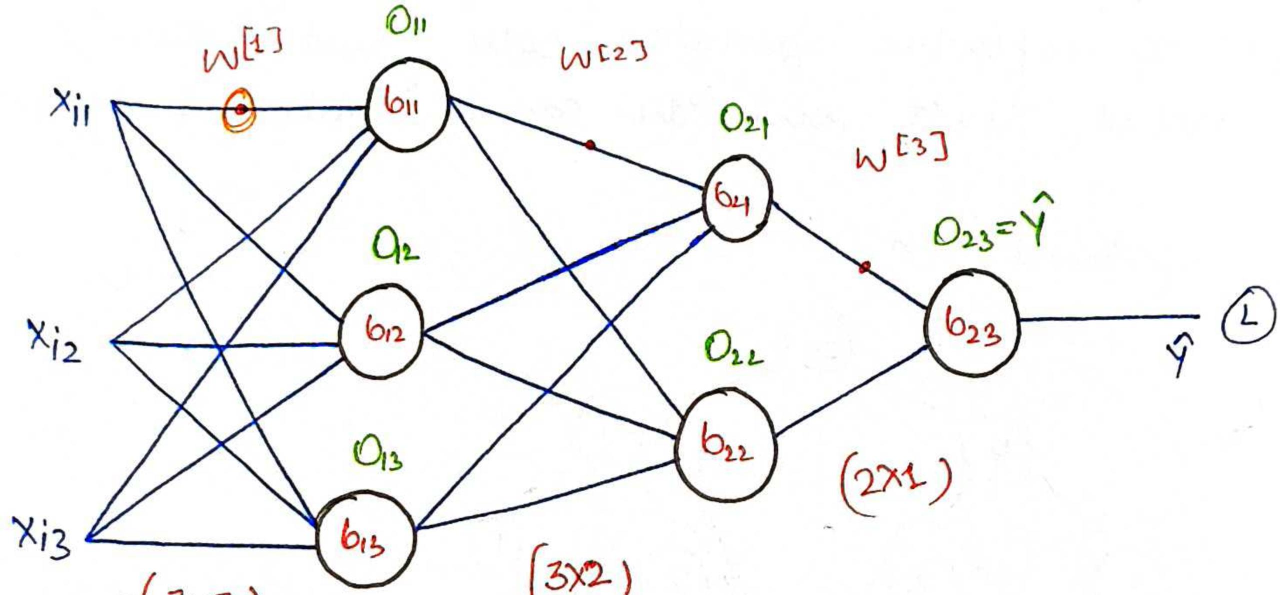
```
def fib(num, dict):
    if n in dict:
        return d[n]
    else:
        dict[n] = fib(n-1, d) + fib(n-2, d)
        return d[n]

d = {0: 1, 1: 1}
fib(38, d)
```

→ If data is already present in dict, so, we can use same value.

If not then calculate and store in dict. so, other number can use.

→ Sending dictionary

MLP Memoization



(3x3) (3x2) (2x1)

$(9+3)=12$ $(6+2)=8$ $(2+1)=3$ $= 23$ → trainable Parameter

Chain Rule

$L \rightarrow \hat{y} \rightarrow O_{21} \rightarrow w_{11}^{2}$

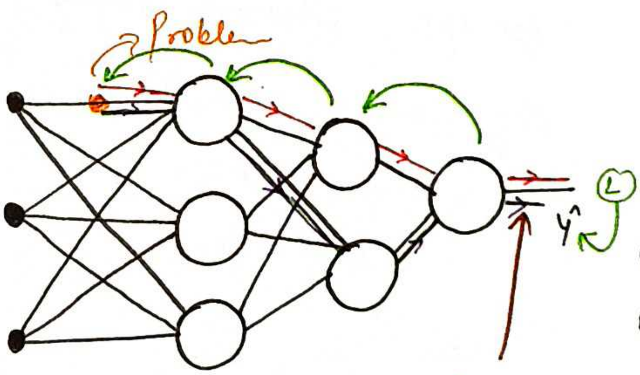
$$\frac{\partial L}{\partial w_{11}^{3}} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial w_{11}^{3}}$$

$$\frac{\partial L}{\partial w_{11}^{2}} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial O_{21}} \times \frac{\partial O_{21}}{\partial w_{11}^{2}}$$

Problem

Chain Rule

$L \rightarrow \hat{y} \rightarrow O_{21} \rightarrow O_{11} - w_{11}^{1}$

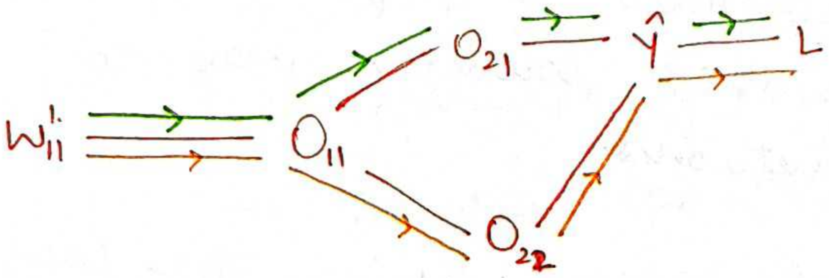


In Math, differentiate this situation (both path) ← So we have to consider both path.

eg:-

$$
\begin{tikzcd}
 & f(x) \arrow[dr] & \\
x \arrow[ur] \arrow[dr] & & h(f(x), g(x)) \\
 & g(x) \arrow[ur] &
\end{tikzcd}
$$

$$\frac{\partial h}{\partial x} = \left[\frac{\partial h}{\partial f(x)} \times \frac{\partial f(x)}{\partial x}\right] + \left[\frac{\partial h}{\partial g(x)} \times \frac{\partial g(x)}{\partial x}\right]$$



$$\frac{\partial L}{\partial w'_{11}} = ?$$

$$\frac{\partial L}{\partial w'_{11}} = \frac{\partial L}{\partial \hat{y}}\left[\underline{\frac{\partial \hat{y}}{\partial O_{21}} \times \frac{\partial O_{21}}{\partial O_{11}} \times \frac{\partial O_{11}}{\partial w_{11}}} + \underline{\frac{\partial \hat{y}}{\partial O_{22}} \times \frac{\partial O_{22}}{\partial O_{11}} \times \frac{\partial O_{11}}{\partial w'_{11}}}\right]$$

→ Used in $\frac{\partial L}{\partial w_{11}^2}$ → We don't need to calculate again.

Backpropagation → chain diff (rule) + Memoization

Summary

1. Layer increasing ↔ complexity increase to calculate derivative
2. When complexity increase ⟷ same value calculate multiple time and time also increase.
3. Used Memoization to store value and use same value when need to calculate.