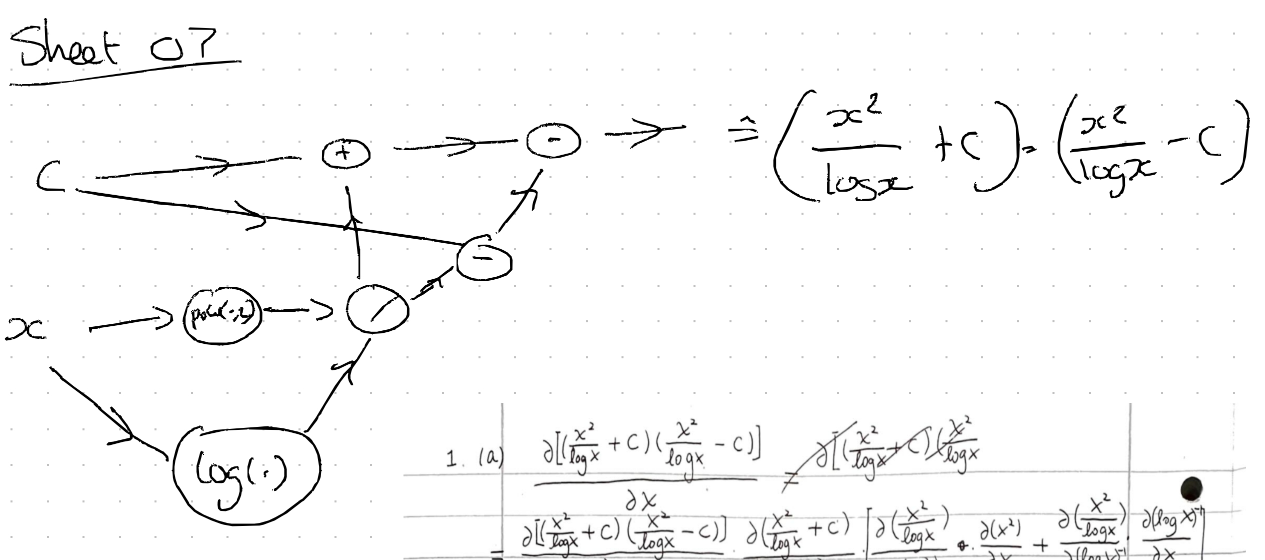
Sheet 07




1. (a) $\frac{\partial\left[\left(\frac{x^2}{\log x}+c\right)\left(\frac{x^2}{\log x}-c\right)\right]}{\partial x}$ $\neq$ $\cancel{\partial\left[\left(\frac{x^2}{\log x}+c\right)\left(\frac{x^2}{\log x}\right.\right.}$

$= \frac{\partial\left[\left(\frac{x^2}{\log x}+c\right)\left(\frac{x^2}{\log x}-c\right)\right]}{\partial\left(\frac{x^2}{\log x}+c\right)} \cdot \frac{\partial\left(\frac{x^2}{\log x}+c\right)}{\partial\left(\frac{x^2}{\log x}\right)} \left[\frac{\partial\left(\frac{x^2}{\log x}\right)}{\partial(x^2)} \cdot \frac{\partial(x^2)}{\partial x} + \frac{\partial\left(\frac{x^2}{\log x}\right)}{\partial(\log x)^{-1}} \cdot \frac{\partial(\log x)^{-1}}{\partial x}\right]$

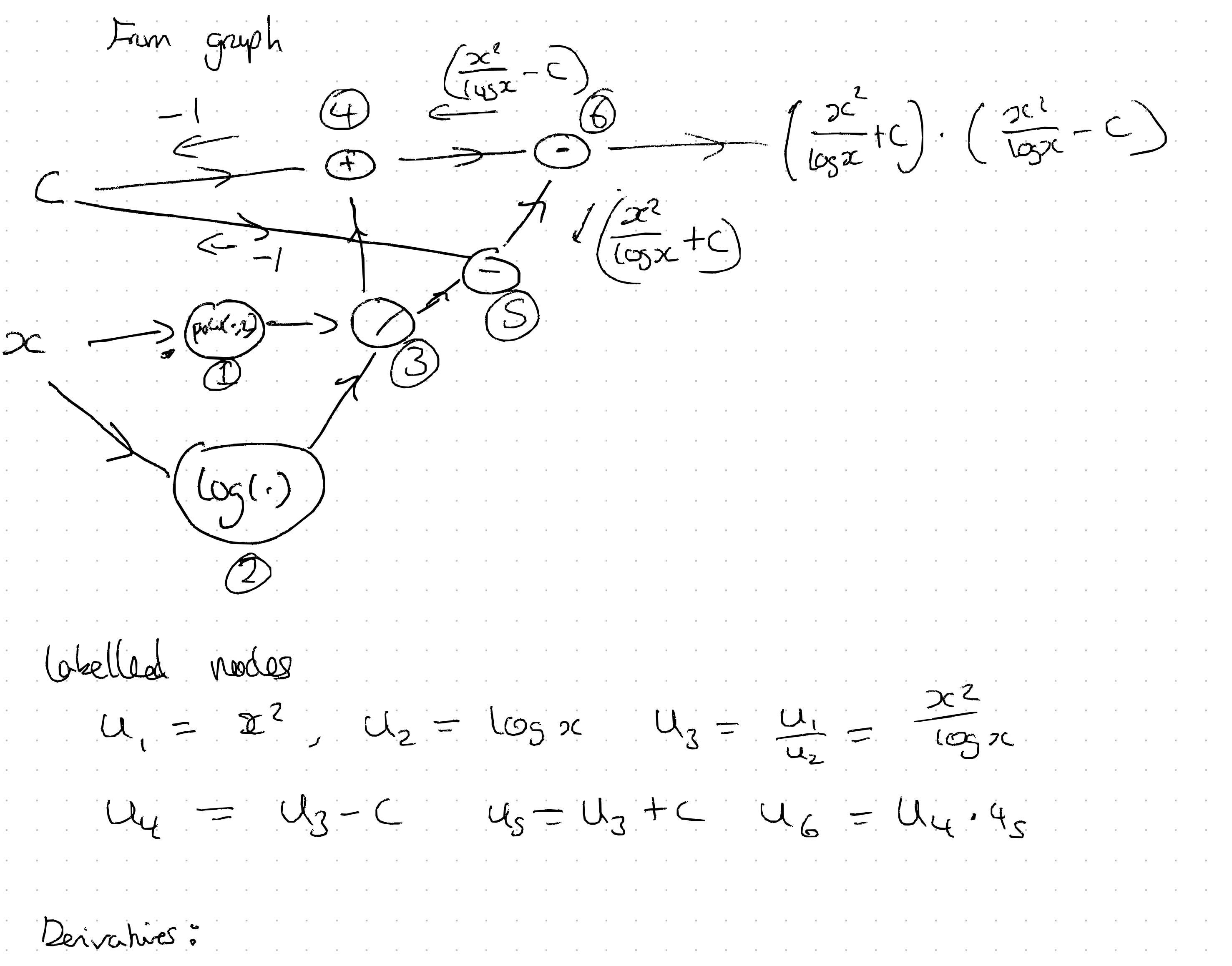
$+ \frac{\partial\left[\left(\frac{x^2}{\log x}+c\right)\left(\frac{x^2}{\log x}-c\right)\right]}{\partial\left(\frac{x^2}{\log x}-c\right)} \cdot \frac{\partial\left(\frac{x^2}{\log x}-c\right)}{\partial\left(\frac{x^2}{\log x}\right)} \left[\frac{\partial\left(\frac{x^2}{\log x}\right)}{\partial(x^2)} \cdot \frac{\partial(x^2)}{\partial x} + \frac{\partial\left(\frac{x^2}{\log x}\right)}{\partial(\log x)^{-1}} \cdot \frac{\partial(\log x)^{-1}}{\partial x}\right]$

a) $\frac{x^4}{\log^2 x} - c^2$

derivative: $\frac{\partial}{\partial x}\left(\frac{x^4}{\log^2 x}\right) = \frac{4x^3}{\log^2 x} - \frac{2x^4}{\log^3 x}\frac{1}{x}$

$= 2x^3\left(\frac{2}{\log^2 x} - \frac{1}{\log^3 x}\right)$

From graph




labelled nodes

$u_1 = x^2, \quad u_2 = \log x \quad u_3 = \frac{u_1}{u_2} = \frac{x^2}{\log x}$

$u_4 = u_3 - c \quad u_5 = u_3 + c \quad u_6 = u_4 \cdot u_5$

Derivatives:

1) $\frac{\partial u_1}{\partial x} = 2x$

2) $\frac{\partial u_2}{\partial x} = 1/x$

3) $\frac{\partial u_3}{\partial x} = \frac{1}{u_2}\frac{\partial u_1}{\partial x} - \frac{u_1}{u_2^2}\frac{\partial u_2}{\partial x} = \frac{2x\log x - x}{\log^2 x}$

4) $\frac{\partial u_4}{\partial x} = \frac{\partial u_3}{\partial x}$

5) $\frac{\partial u_5}{\partial x} = \frac{\partial u_3}{\partial x}$

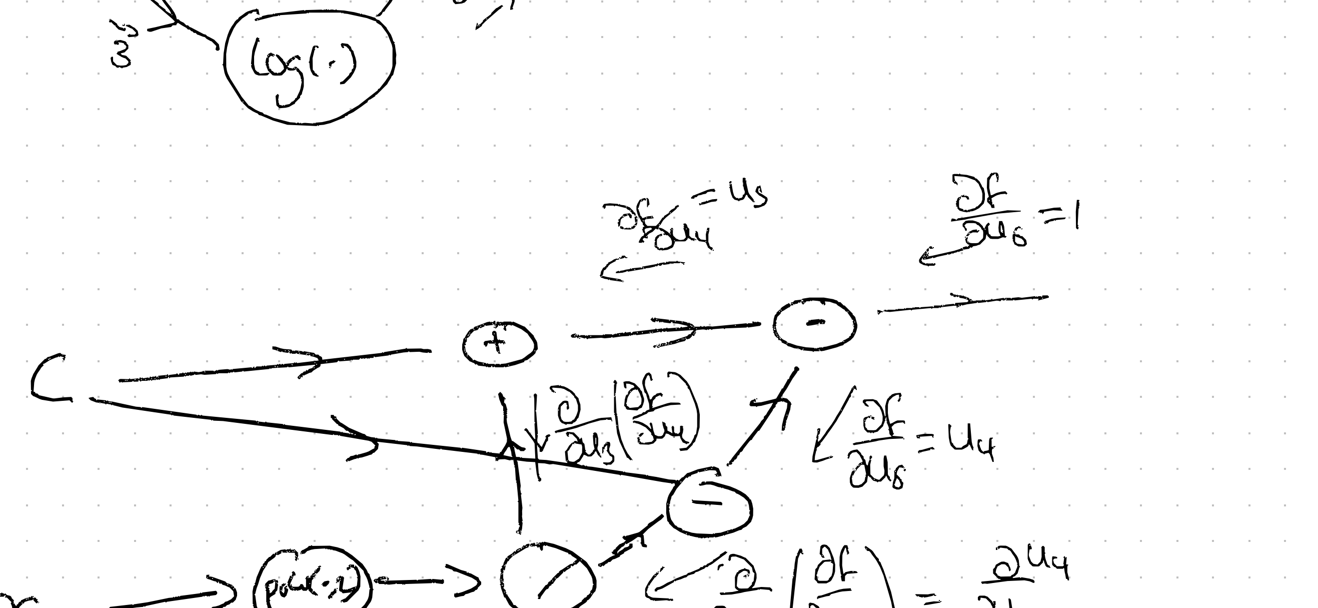
6) $\frac{\partial u_6}{\partial x} = u_4\frac{\partial u_5}{\partial x} + u_5\frac{\partial u_4}{\partial x}$

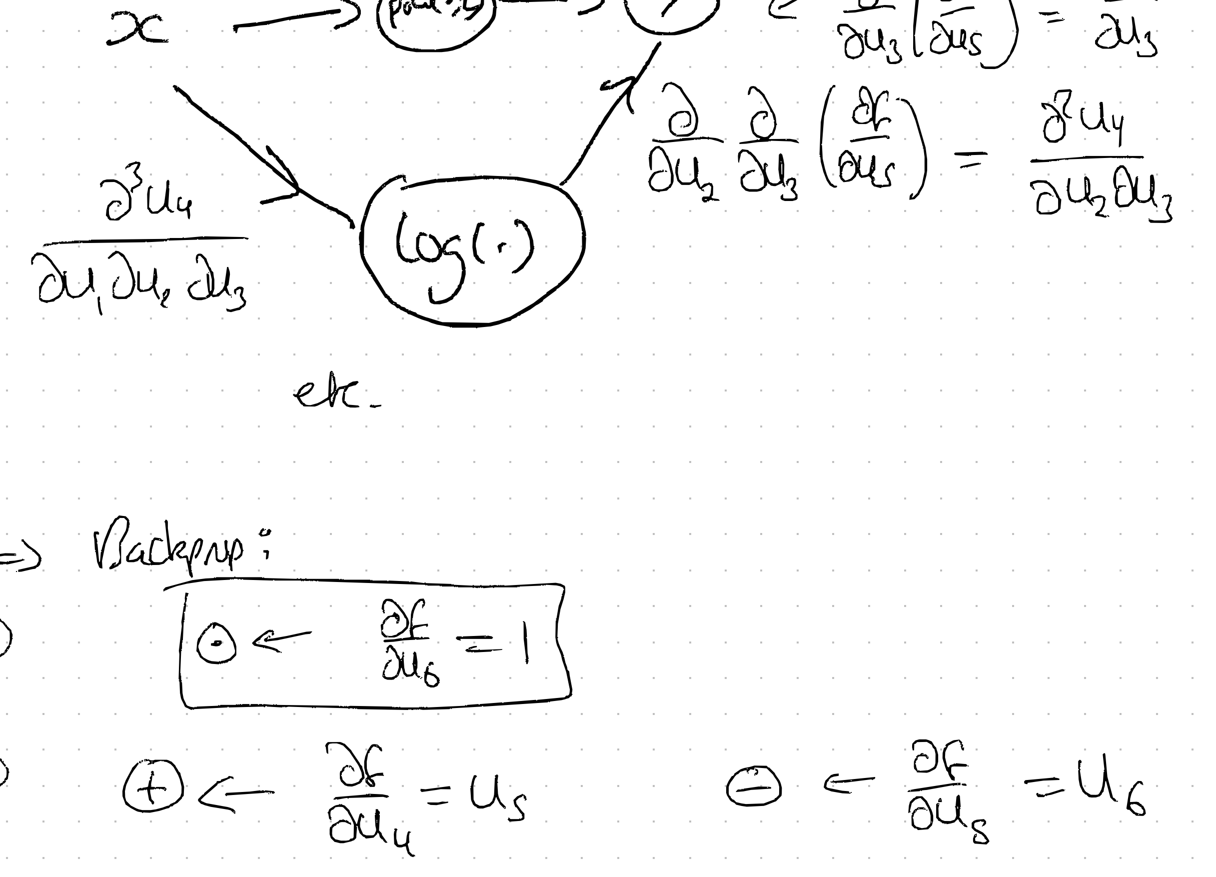
$\Rightarrow f = u_6 \Rightarrow \frac{\partial f}{\partial x} = u_4\frac{\partial u_5}{\partial x} + u_5\frac{\partial u_4}{\partial x}$

$\frac{\partial u_5}{\partial x} = \frac{\partial u_4}{\partial x} = \frac{\partial u_3}{\partial x}$

$\Rightarrow \frac{2x\log x - x}{\log^2 x}\left[\left(\frac{x^2}{\log x} - c\right) + \left(\frac{x^2}{\log x} + c\right)\right] = \left[\frac{2x\log x - x}{\log^2 x}\right]\frac{x^2}{\log x}$

b)




etc.

$\Rightarrow$ Backprop:

1. $\boxed{\odot \leftarrow \frac{\partial f}{\partial u_6} = 1}$

2. $\oplus \leftarrow \frac{\partial f}{\partial u_4} = u_5 \qquad \ominus \leftarrow \frac{\partial f}{\partial u_5} = u_6$

3. $c \leftarrow \frac{\partial u_5}{\partial c} \quad -1$

   $\leftarrow \frac{\partial u_4}{\partial c} \quad +1 \qquad = 0$

4. $\frac{\partial f}{\partial u_3} = \frac{\partial f}{\partial u_4}\frac{\partial u_4}{\partial u_3} = u_5\frac{\partial u_4}{\partial u_3}$

   $\oslash \leftarrow \frac{\partial f_1}{\partial u_3} \Rightarrow u_4\frac{\partial u_5}{\partial u_3}$ $\Big\}$ $\left(\frac{x^2}{\log x} - c\right) \cdot \frac{\partial}{\partial\frac{x^2}{\log x}}\left(\frac{x^2}{\log x} + c\right)$

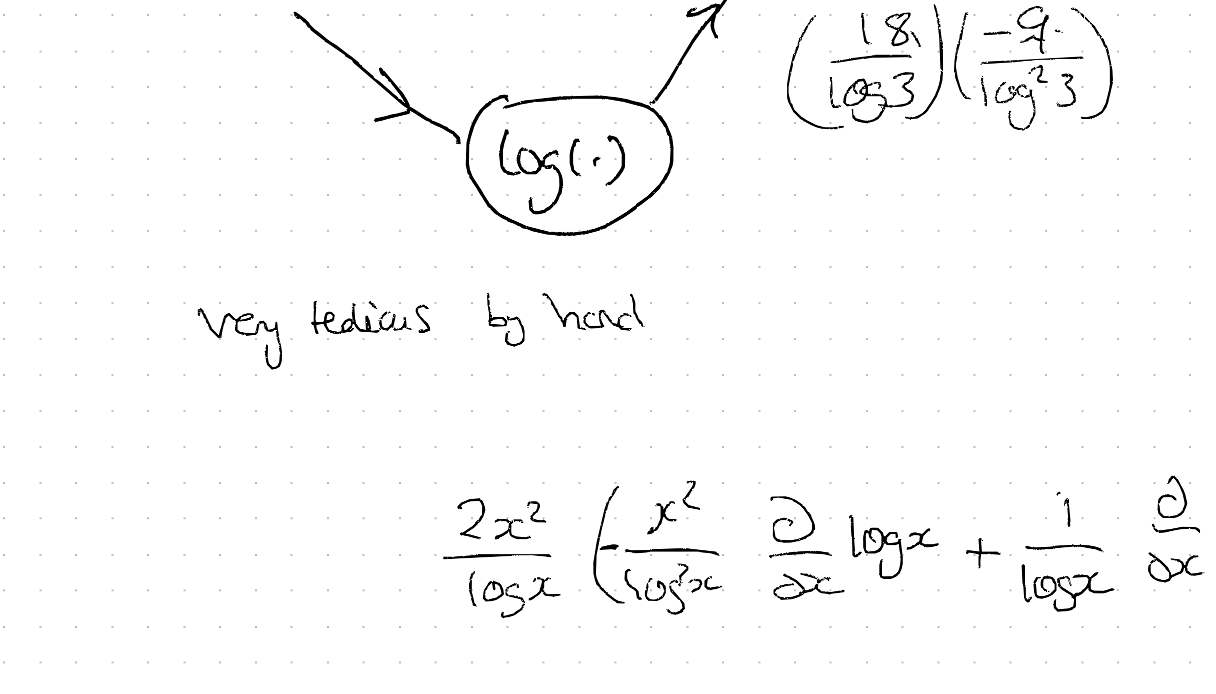
   $+ \ldots$

   $= \frac{2x^2}{\log x}$

5. log(·) $\leftarrow$ $\frac{\partial f}{\partial u_2} = \frac{\partial f}{\partial u_3}\frac{\partial u_3}{\partial u_2} = \left(\frac{2x^2}{\log x}\right)\frac{\partial}{\partial \log x}\left(\frac{x^2}{\log x}\right)$

6. pow(·,2) $\leftarrow$ $\frac{\partial f}{\partial u_1} = \frac{\partial f}{\partial u_3}\frac{\partial u_3}{\partial u_1} = \left(\frac{2x^2}{\log x}\right)\frac{\partial}{\partial x^2}\left(\frac{x^2}{\log x}\right)$

Now calculating values




very tedious by hand

$\frac{2x^2}{\log x}\left(-\frac{x^2}{\log^2 x}\frac{\partial}{\partial x}\log x + \frac{1}{\log x}\frac{\partial}{\partial x}x^2\right)$

$= \frac{2x^2}{\log x}\left(-\frac{x}{\log^2 x} + \frac{2x}{\log x}\right) = \boxed{\frac{4x^3}{\log^2 x} - \frac{2x^3}{\log^3 x}}$

$\frac{\partial f}{\partial c} = -2c \qquad \frac{\partial f}{\partial x} =$

c) Using pytorch

$\frac{\partial f}{\partial x} = 48.757$ ✓ $\quad \frac{4 \times 27}{\log^2 3} - \frac{2 \times 27}{\log^3 3} = 48.757$ ✓

$\frac{\partial f}{\partial c} = -10.0 = -2c$ ✓

$\rightarrow$ as predicted

```python
import torch
```
✓ 8.4s Python

```python
# Define x and c as PyTorch tensors with requires_grad=True to enable gradient computation
x = torch.tensor(3.0, requires_grad=True)  # x = 3
c = torch.tensor(5.0, requires_grad=True)  # c = 5

# Define the function f(x, c)
u1 = x**2
u2 = torch.log(x)
u3 = u1 / u2
u4 = u3 + c
u5 = u3 - c
f = u4 * u5  # f = (u3 + c) * (u3 - c)
```
✓ 0.1s Python

```python
# Compute the gradients using backward()
f.backward()

# Output the computed derivatives
df_dx = x.grad  # ∂f/∂x
df_dc = c.grad  # ∂f/∂c

print(f"Derivative with respect to x: {df_dx.item()}")
print(f"Derivative with respect to c: {df_dc.item()}")
```
✓ 0.2s Python

```
Derivative with respect to x: 48.756893157958984
Derivative with respect to c: -10.0
```

2) ADAM optimisers

① $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$

decay rate, often $\beta = 0.9$

Momentum (mean)

gradient

↳ moving average of gradients

② $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$

Second moment (variance)

grad $^2$

↳ moving average of squared gradients.

③ Bias correction:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \qquad \hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

↳ Adjust for initialisation bias

④ Update parameters

learning rate

$$w_{t+1} = w_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t}^2 + \varepsilon}$$

Prevent poles

updates weights with scaling $\sim \frac{m_t}{\sqrt{v_t}}$

---

b) Initialise to zero

$\Rightarrow m_t = (1-\beta_1) g_t \quad \Rightarrow \hat{m}_t = g_t$

$v_t = (1-\beta_2) g_t^2 \quad \Rightarrow \hat{v}_t = g_t^2$

$$w_{t+1} = w_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}$$

$$\frac{\hat{m}_t}{\sqrt{\hat{v}_t}} = \frac{g}{\sqrt{g^2}} = \boxed{\text{sign}(g)}$$

c) Second iteration:

$$m_{t+1} = \beta_1 (1-\beta_1) g_t + (1-\beta_1) g_{t+1}$$

$$v_{t+1} = \beta_2 (1-\beta_2) g_t^2 + (1-\beta_2) g_{t+1}^2$$

$$\hat{m}_{t+1} = \frac{\beta_1 g_t + g_{t+1}}{1+\beta_1}$$

$$\hat{v}_{t+1} = \frac{\beta_2 g_t^2 + g_{t+1}^2}{1+\beta_2}$$

$$\Rightarrow \frac{\hat{m}_{t+1}}{\sqrt{\hat{v}_{t+1}}} = \frac{g_{t+1}(1 + g_t/g_{t+1})}{\sqrt{g_{t+1}^2}\sqrt{(1+(g_t/g_{t+1})^2}} \frac{\sqrt{(1+\beta_2)}}{1+\beta_1} = \text{Sign}(g) \times \frac{1+\varepsilon}{\sqrt{1+\varepsilon^2}} \frac{\sqrt{1+\beta_2}}{1+\beta_1}$$

Assuming: $g_t/g_{t+1}$ small

$$\approx \text{Sign}(g)(1+\varepsilon)(1-\tfrac{1}{2}\varepsilon^2)\sqrt{\frac{1+\beta_2}{(1+\beta_1)^2}}$$

$$= \text{Sign}(g)(1+\varepsilon)\sqrt{\frac{1+\beta_2}{(1+\beta_1)^2}} + O(\varepsilon^2)$$

Small perturbation to gradient

extra weighting

d) ~~Gradient rescaling is possible solution~~

~~use $\hat{g}_t = \frac{g_t}{1-\beta_1^t}$~~

Alternatively: change learning rate → Start small, and then grow.

↳ Adaptive learning rate

e) MLP: weights $w$ trained with Adam and L2 Reg.

Is there difference with L2 penalty $\|w\|_2^2$ in loss or weight decay directly onto weights

L2 reg:

$$L' = L + \lambda \|w\|_2^2$$

For Adam: $g_t = \nabla L$

$$\Rightarrow g_t \to \nabla L' = \boxed{\nabla L + 2\lambda w}$$

↳ $w_{t+1}$ affected implicitly

↳ Additional term in grad.

$\boxed{\text{↳ Effective strength of Reg varies}}$

weight decay:

$$w_{t+1} = w_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} - \alpha \lambda w_t$$

$\boxed{\text{AdamW}}$

↳ additional decoupled weight decay

↳ does not interfere with adaptive gradient descent

⇒ often favoured

→ const. effective strength of grad.

3) Work backwards from final
$7 \times 7 \times 512$

$\Rightarrow$ Field of view is $7 \times 7$

Max pooling $2 \times 2$ $\rightarrow$ Receptive field doubles (going backward)
Conv : $3 \times 3$ $\Rightarrow$ New field = old + 2 × Dilation
Here $\boxed{\text{Dilation} = 1}$
$\Rightarrow$ New = old + 2

Working backwards:

Block 5 : Max pool, 3 × conv
Backwards 3 conv $\Rightarrow$ Max pool

1 pixel in 7x7 view: $1 \rightarrow 2$ Max pool
$3 \rightarrow 5$
$5 \rightarrow 7$
$7 \rightarrow 14$ conv

Block 4 :
Rev : 3 conv, max pool.

$8 \rightarrow 16$
$16 \rightarrow 18$
$18 \rightarrow 20$
$20 \rightarrow 22$

Block 3 : Same again

$22 \rightarrow 44$
$44 \rightarrow 46$
$46 \rightarrow 48$
$48 \rightarrow 50$

Block 2 : Max pool, 2 conv

$50 \rightarrow 100$
$100 \rightarrow 102$
$102 \rightarrow 104$

Block 1 :

$104 \rightarrow 208$
$208 \rightarrow 210$
$210 \rightarrow 212$

Each pixel in $7 \times 7$ has info from
$212 \times 212$ subset of
initial image

(b) # parameters (with bias) in convolutional layers $= \cancel{3\times3\times64}$

$$(3\times3\times3+1)\times64 + (3\times3\times64+1)\times64$$
$$+ (3\times3\times64+1)\times128 + (3\times3\times128+1)\times128$$
$$+ (3\times3\times128+1)\times256 + (3\times3\times256+1)\times256\times2$$
$$+ (3\times3\times256+1)\times512 + (3\times3\times512+1)\times512\times2$$
$$\cancel{+(3\times3\times256+1)\times2} + (3\times3\times512+1)\times512\times3$$
$$= 14714688 \approx 1.47\times10^{7}$$

# parameters in fully connected layers

$$= (7\times7\times512\times4096+4096)$$
$$+ 4096\times4096+4096$$
$$+ 4096\times1000+1000$$
$$= 123642856 \approx 1.24\times10^{8}$$

# parameters in total $= 138357544 \approx 1.38\times10^{8}$

~~#p of c~~ $\dfrac{\#p \text{ of convolutional}}{\#p \text{ of fully connected}} = 11.90\%$