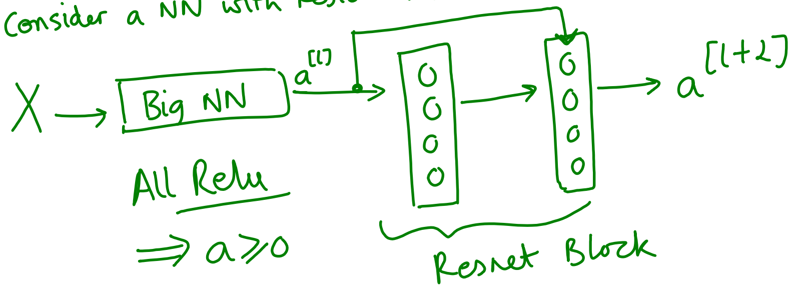
Why Resnets Work?

- consider a normal N/w:

$X \rightarrow \boxed{\text{Big NN}} \rightarrow a^{[l]}$

- Consider a NN with Residual Block



$$a^{[l+2]} = g\left(z^{[l+2]} + a^{[l]}\right)$$

$$= g\left(W^{[l+2]} \cdot a^{[l+1]} + b^{[l+2]} + a^{[l]}\right)$$

If $W^{[l+2]} = b^{[l+2]} = 0$

then $a^{[l+2]} = g(a^{[l]}) = \text{Relu}(a^{[l]})$

$= a^{[l]}$ (since $a \geq 0$ & $\text{Relu}(a \geq 0) = a$)

$\Rightarrow$ Identity func. is easy for Residual block to learn

ie, $a^{[l+2]} = a^{[l]}$ is possible

Because of Residual block

$\Rightarrow$ Residual NN performs atleast as good as normal NN

$\Rightarrow$ If we add a few Residual blocks in between/end of NN, we will achieve performance $\geq$ performance to normal NN. The other issue with normal NN is that, if the layers are large, then it becomes harder to even learn params for the Identity func in later layers, in Resnet, you just pass the prev state of the N/w & w/o learning new params you can achieve the Identity matrix

— Because in normal NN, they can't even predict the Identity Activation, performance worsens (see graph on prev page), while in Resnet, they predict Identity + other learning using Residual Block $\Rightarrow$ predictive performance $\uparrow$ as layers $\uparrow$

$$a^{[l+2]} = g(z^{[l+2]} + a^{[l]})$$

$\implies \dim(z^{[l+2]}) = \dim(a^{[l]})$

How to ensure this?

- Make the convolutions the same in size
- If they were diff in dim, say $a^{[l]} = (128 \times 1)$ and $z^{[l+2]} = (256 \times 1)$

then Add W_s in between

$$z^{[l+2]} + W_s \cdot a^{[l]}$$

$\hookrightarrow \dim = (256 \times 128)$

$\implies (256 \times 128) \times (128 \times 1)$

$= (256 \times 1)$

$\implies$ Now you can Add the 2.

Resnet example given In Video $\rightarrow$ W2 L4