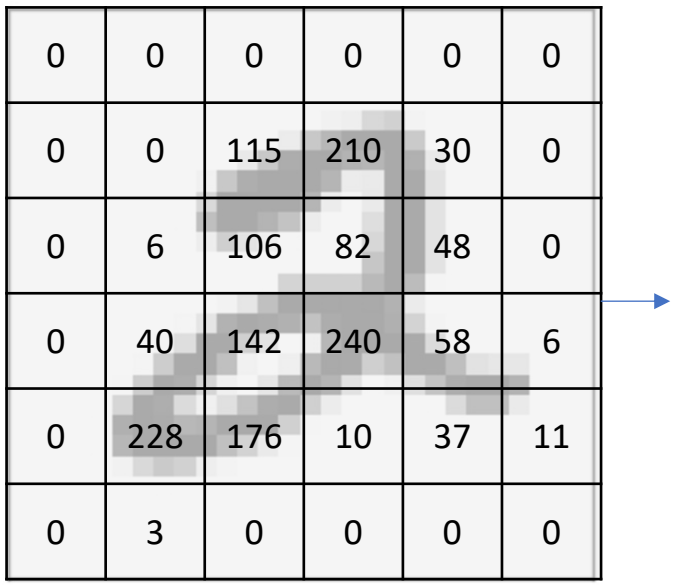
-1	-1	-1	-1	-1	-1
-1	-1	1	1	1	-1
-1	-1	-1	1	1	-1
-1	1	1	1	1	-1
-1	1	1	-1	-1	1
-1	-1	-1	-1	-1	-1

-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1

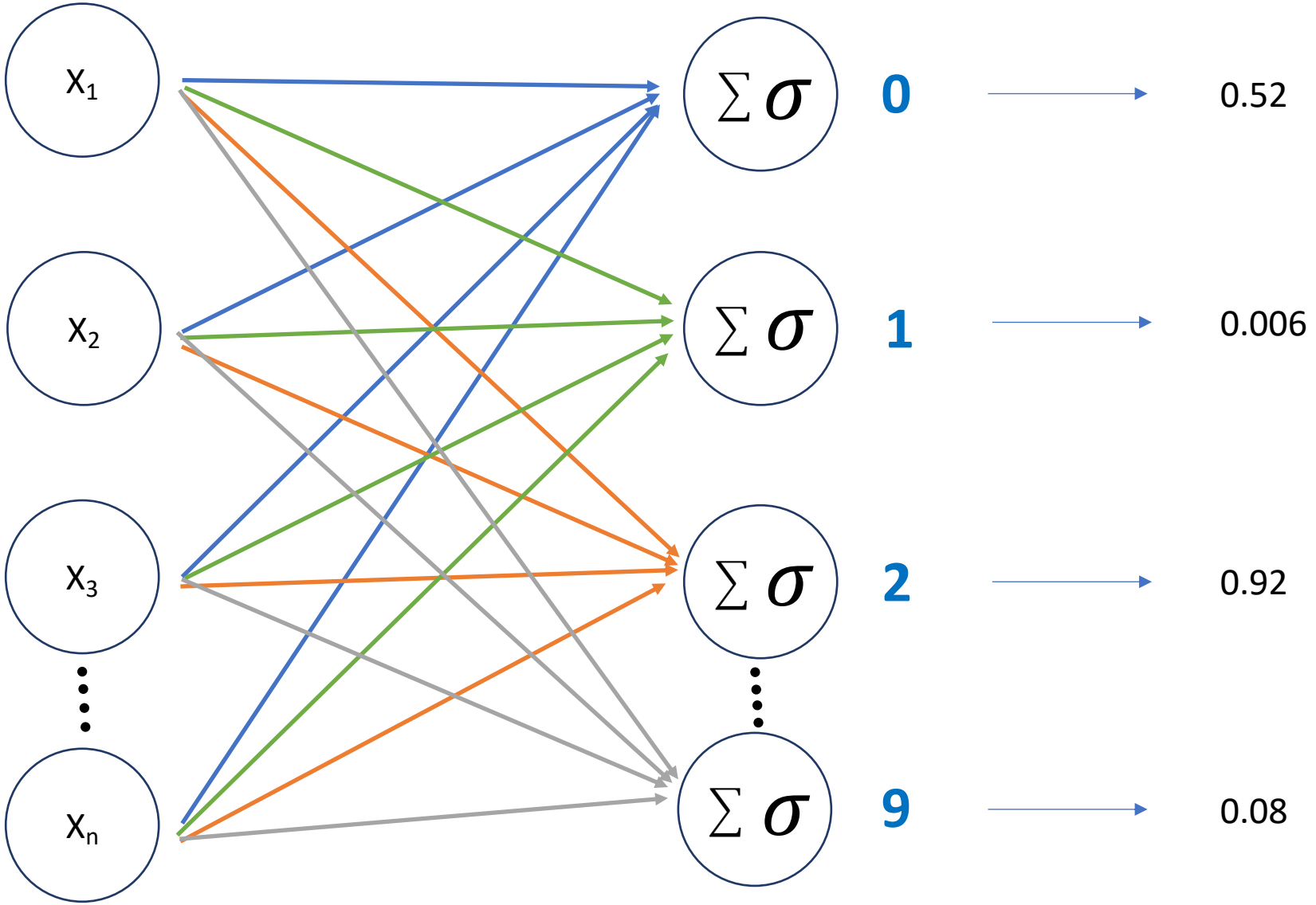
-1	-1	-1	-1	-1	-1
1	1	1	1	-1	-1
-1	-1	1	-1	-1	-1
-1	1	1	-1	-1	-1
-1	1	-1	-1	-1	-1
-1	1	-1	-1	-1	-1

-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1



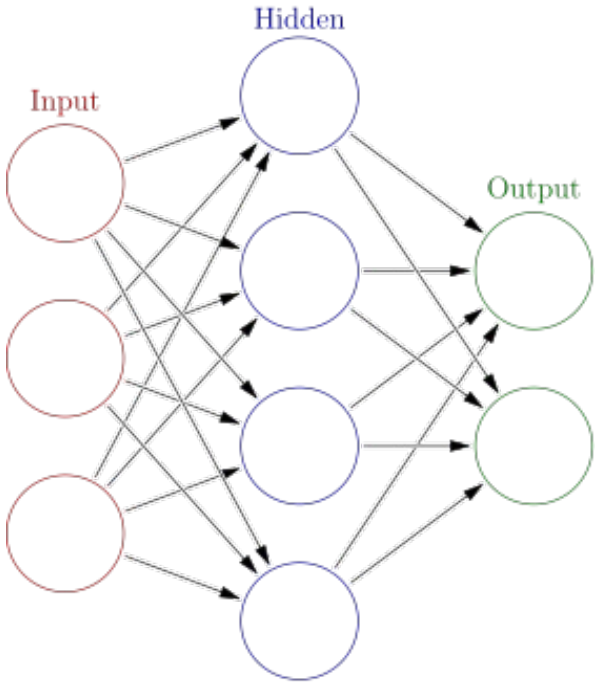
0	0	0	0	0	0
0	0	115	210	30	0
0	6	106	82	48	0
0	40	142	240	58	6
0	228	176	10	37	11
0	3	0	0	0	0

0
0
0
0
0
0
0
115
210
30
0
0
6
106
...
0



$$y = 0.05 * X_1 + 0.25 * X_2 + 0.15 * X_3 + 0.012 * X_4 - 1.22$$

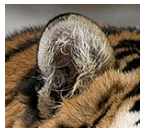
Neural Networks



- image size = 1920 * 1080 * 3
- First layer neurons = 1920 * 1080 * 3 = 6,220,800
- += Hidden layer neurons
- Weights between input and hidden layer = First layer neurons * Hidden layer neurons = around 30 million neurons

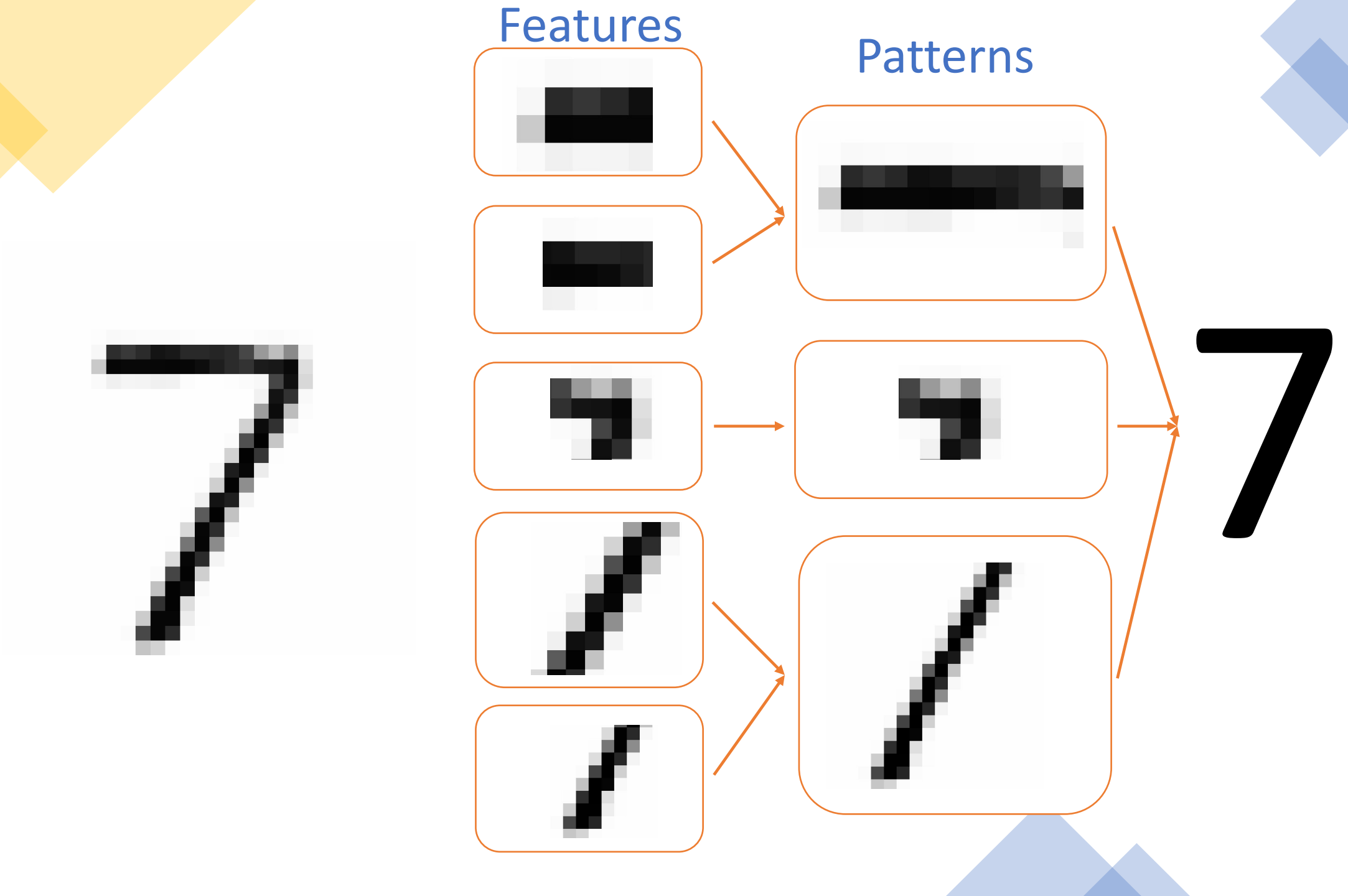
Tiger’s eye?

Tiger’s nose?



Tiger’s ear?

Tiger’s face?


Features
Patterns
7

-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1

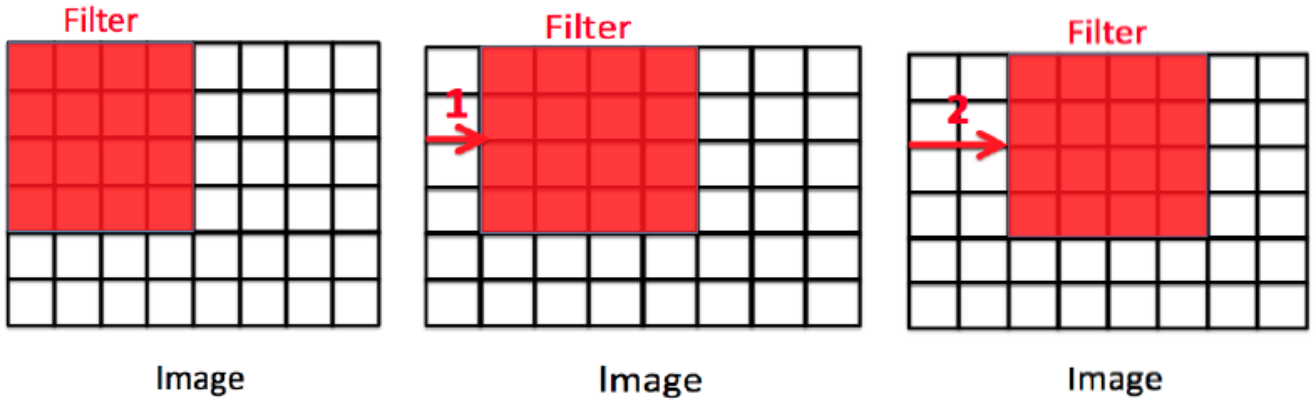
-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1

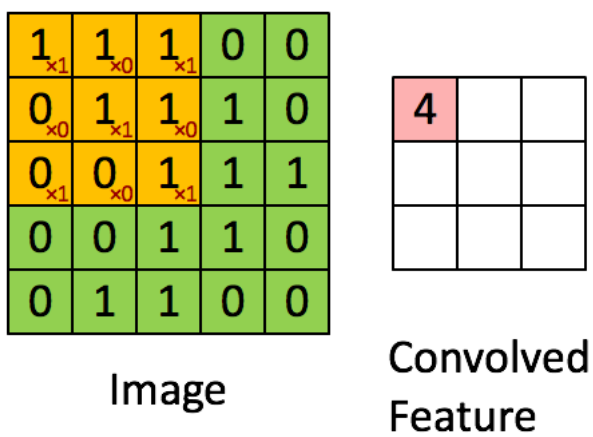
(-1)+(-1)+(-1)+(-1)+1+1+(-1)+(-1)+(-1)=-7

-7 / 9 = -0.78

-0.78	-0.11	-0.11	
	0.11	0.11	

Stride





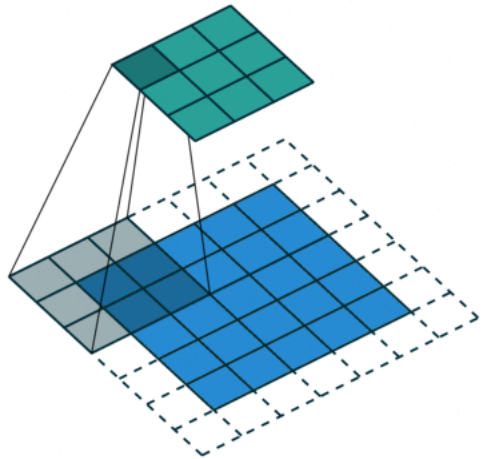


Image credit: https://medium.com/analytics-vidhya/convolution-padding-stride-and-pooling-in-cnn-13dc1f3ada26
https://commons.wikimedia.org/wiki/File:Convolution_arithmetic_-_Padding_strides.gif

Tiger’s Eye Detector

Feature Map

	1	1			

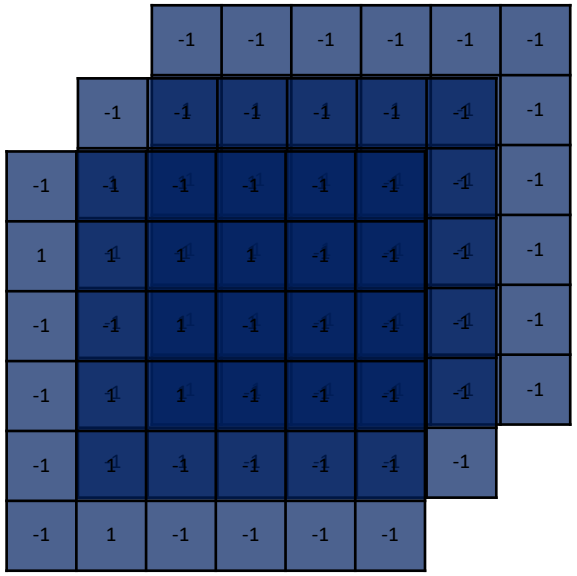
Feature Map

		1		1	

Tiger’s Eye Detector

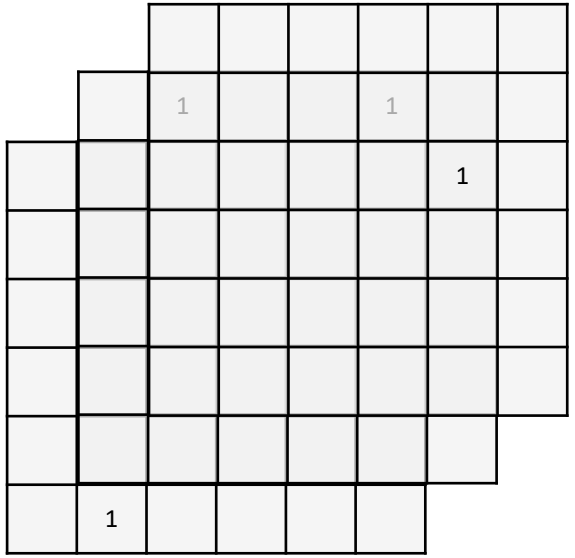
*

Filters



=

Feature Maps



*

Eye

-1	-1	1
-1	1	1
-1	1	-1

Nose

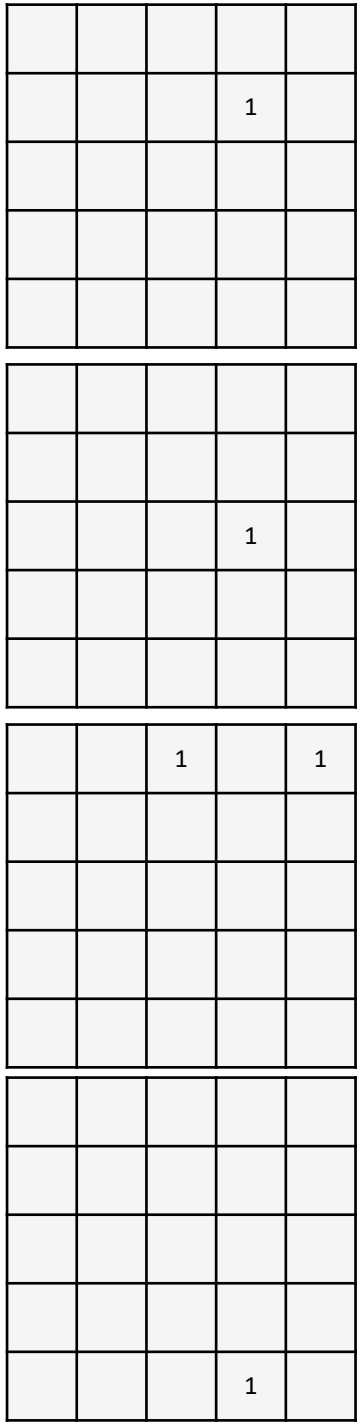
-1	-1	1
-1	1	1
-1	1	-1

Ear

-1	-1	1
-1	1	1
-1	1	-1

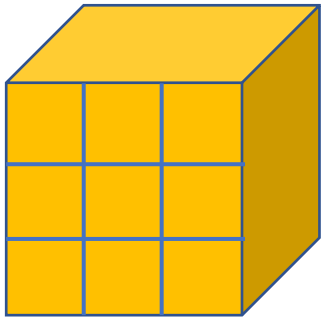
Mouth

-1	-1	1
-1	1	1
-1	1	-1

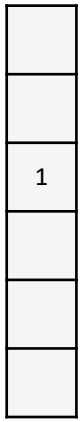


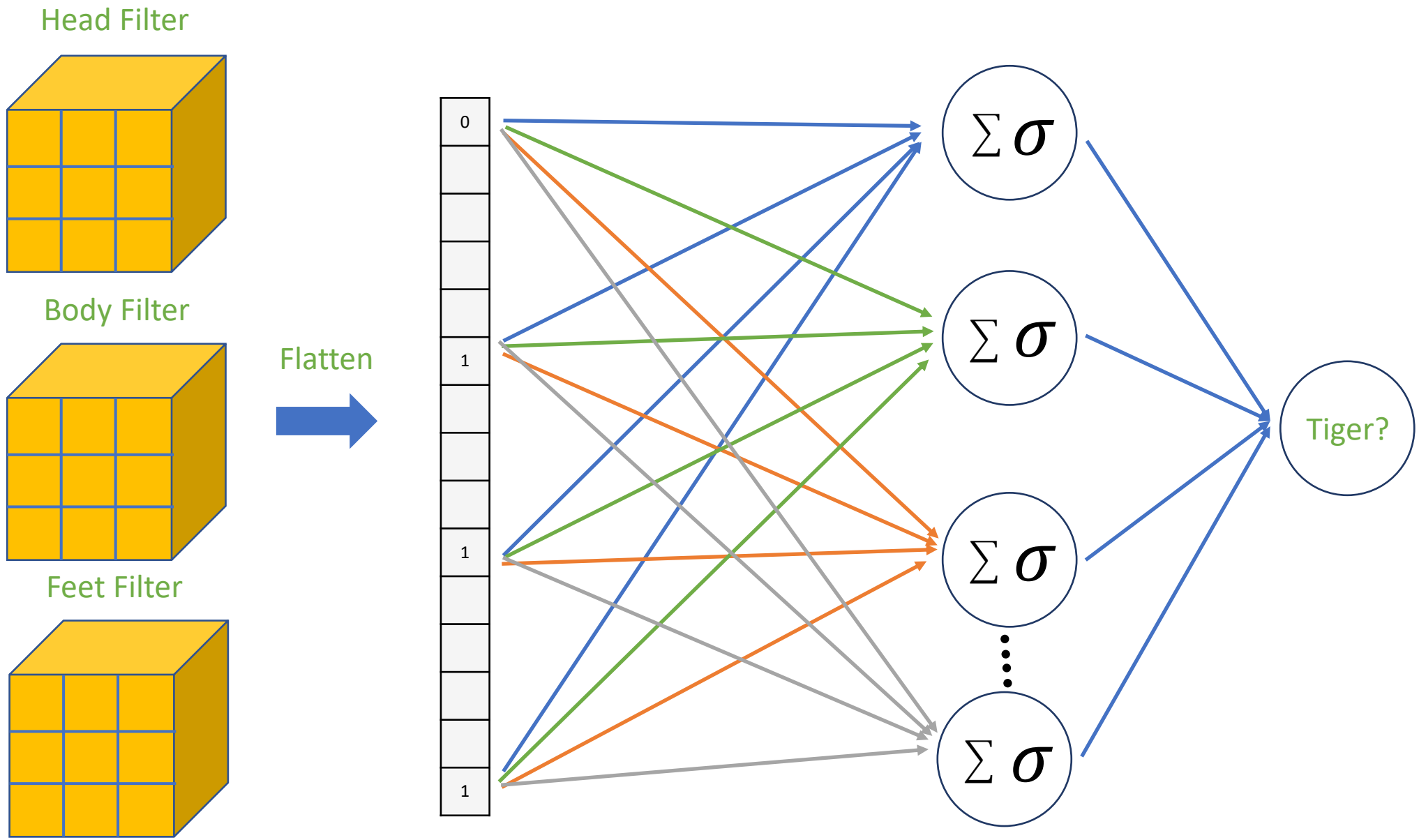
*

Head Filter



Flatten

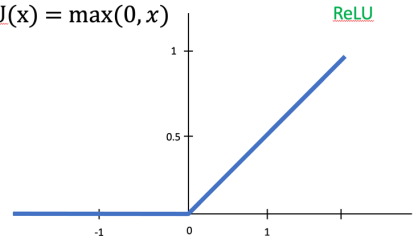



Head Filter
Body Filter
Feet Filter
Flatten
0
1
1
1
∑ σ
∑ σ
∑ σ
∑ σ
Tiger?

-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1

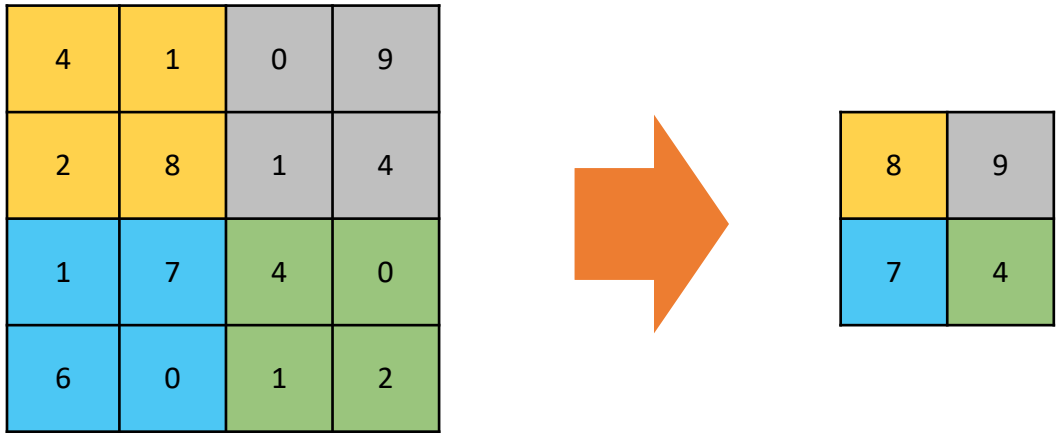
-0.78	-0.11	-0.11	1
	0.11	0.11	

$\mathrm{ReLU}(x) = \max(0, x)$



0	0	0	1
	0.11	0.11	

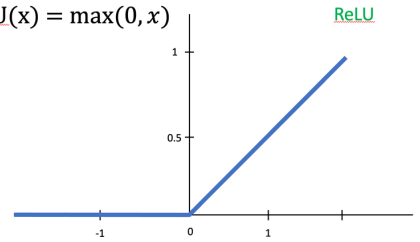
(Max) Pooling



2 by 2 filter with stride = 2

-1	-1	-1	-1	-1	-1
-1	1	1	1	1	-1
-1	-1	-1	1	-1	-1
-1	-1	-1	1	-1	-1
-1	1	1	-1	-1	-1
-1	-1	1	-1	-1	-1

$\mathrm{ReLU}(x) = \max(0, x)$



-0.78	-0.11	-0.11	1
	0.11	0.11	

0	0	0	1
	0.11	0.11	

Max Pooling

0.11	0.11
0.33	0

Benefits of Pooling

- Reduces dimensions and computation
- Reduces overfitting with less parameters

Eye

-1	-1	1
-1	1	1
-1	1	-1

Nose

-1	-1	1
-1	1	1
-1	1	-1

Ear

-1	-1	1
-1	1	1
-1	1	-1

Mouth

-1	-1	1
-1	1	1
-1	1	-1

*

			1	

			1	

		1		1

			1	

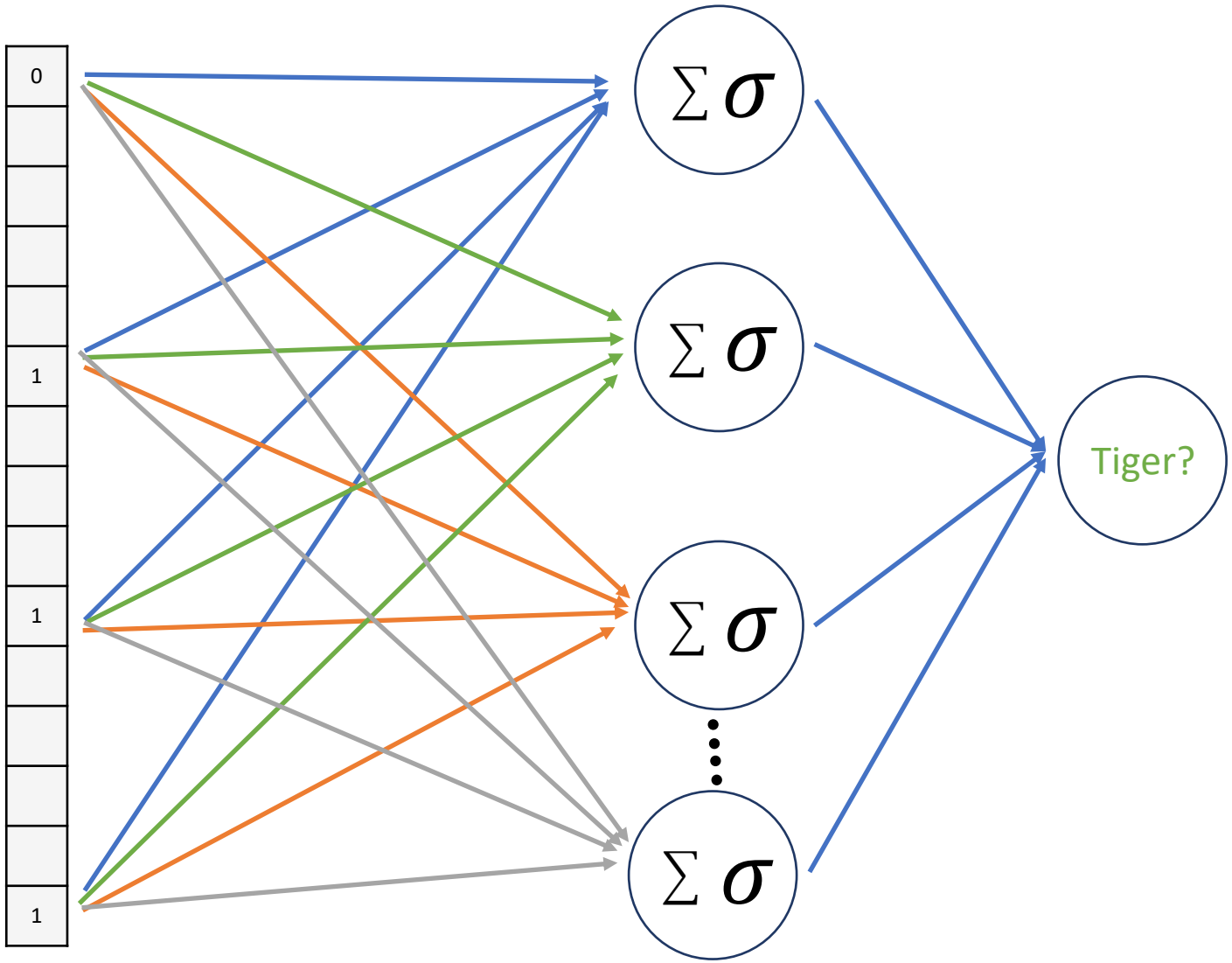
Pooling

Head Filter

Pooling

Flatten

0
1
1
1


0
1
1
1
∑ σ
∑ σ
∑ σ
∑ σ
Tiger?

Any Questions?