



Ayudantía 13

# Redes Neuronales Convolutionales

Por Blanca Romero y Felipe Villagrán

29 de noviembre 2024

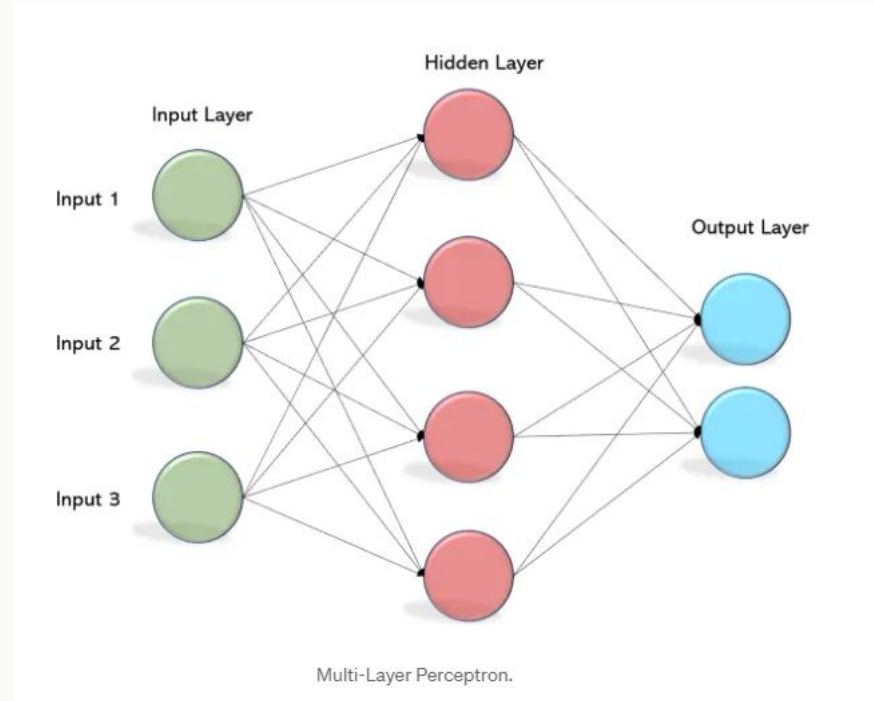


# Recordando MLPs

Se compone de múltiples perceptrones

Sirven para abordar problemas más complejos, por ejemplo, clasificar datos que no son linealmente separables

En la capa de input los datos son **aplanados**





# Limitaciones de los MLPs

Pérdida de información espacial



→  
(Son **muy malos** para trabajar  
con conjuntos de datos visuales)

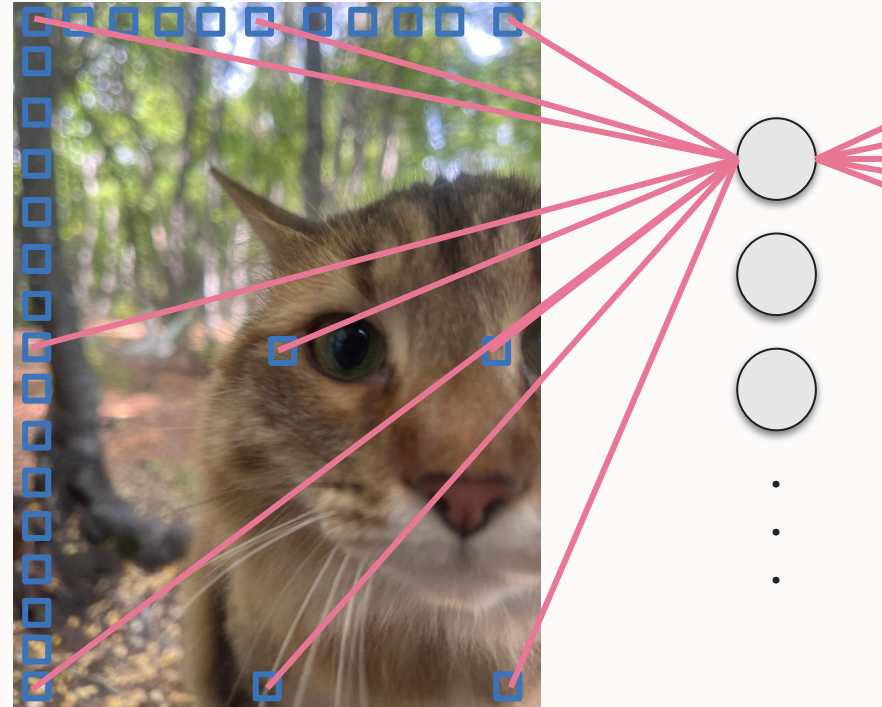




# Limitaciones de los MLPs

Si analizáramos esta foto con un MLP...

**Cada** neurona de **cada** capa analiza tooooooda la imagen, generando una cantidad **enorme** de parámetros

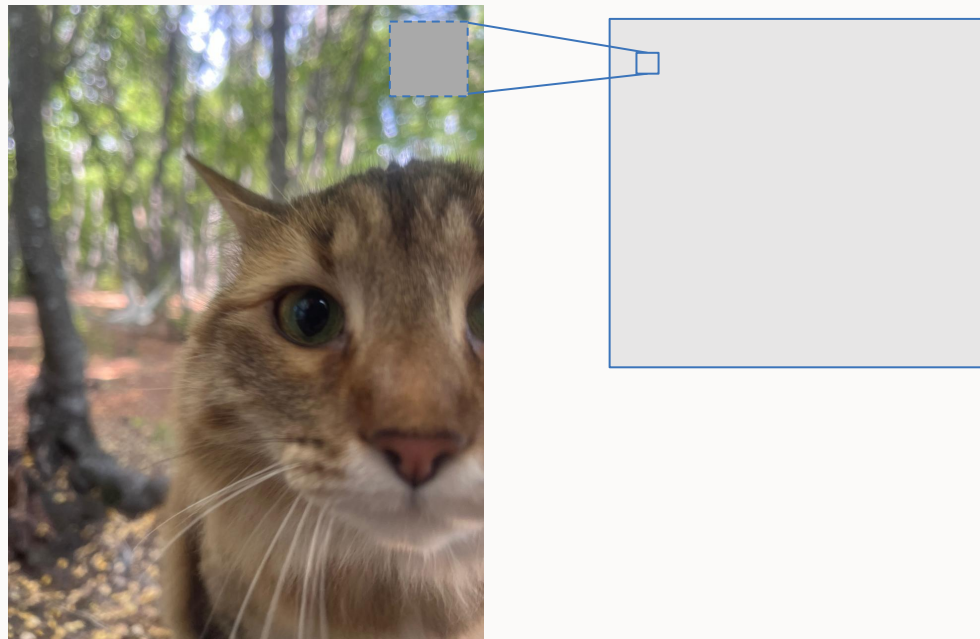




# Redes Neuronales Convolucionales - CNN

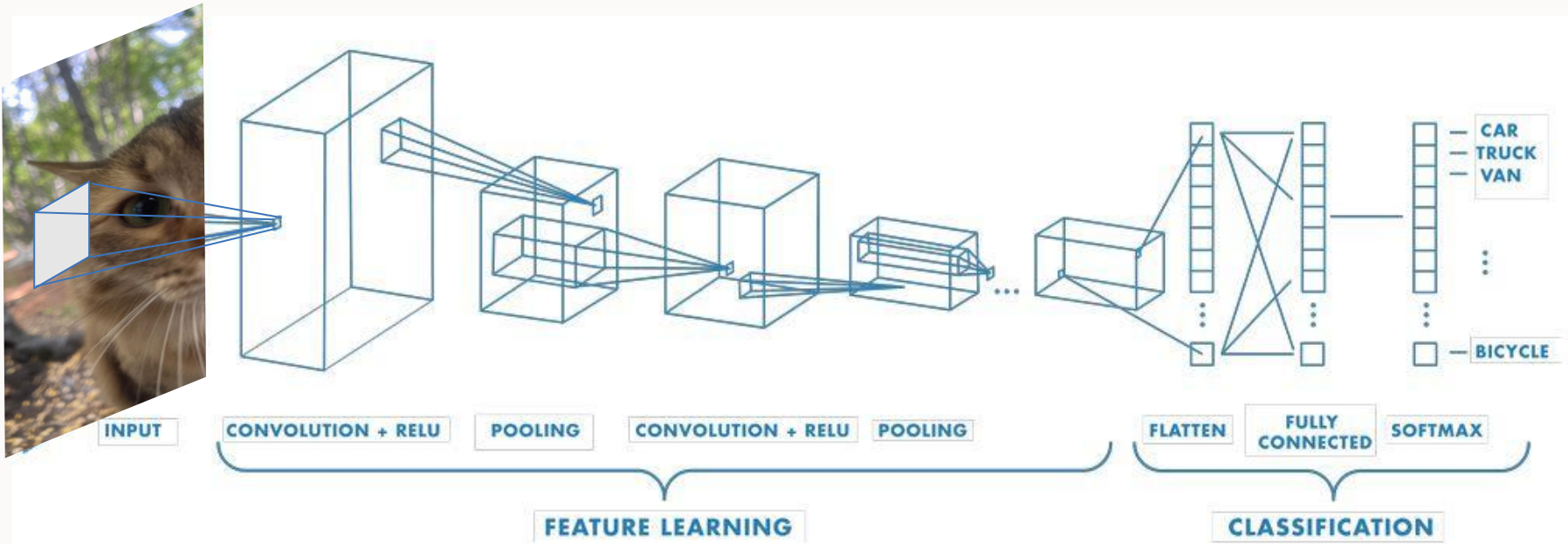
Las CNN nos permiten utilizar la distribución espacial de píxeles.

Usan filtros (kernels) que se barren sobre la foto. Se toma el producto punto en cada posición y así se obtienen *feature maps* (mapas de activación) que destacan ciertas características o patrones de la imagen de input.





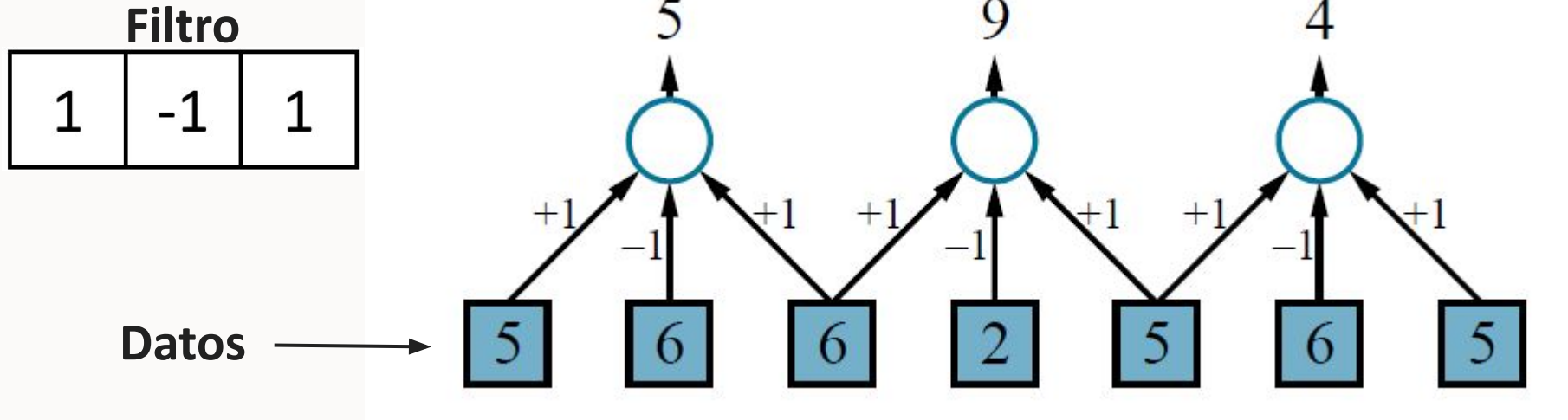
# Estructura



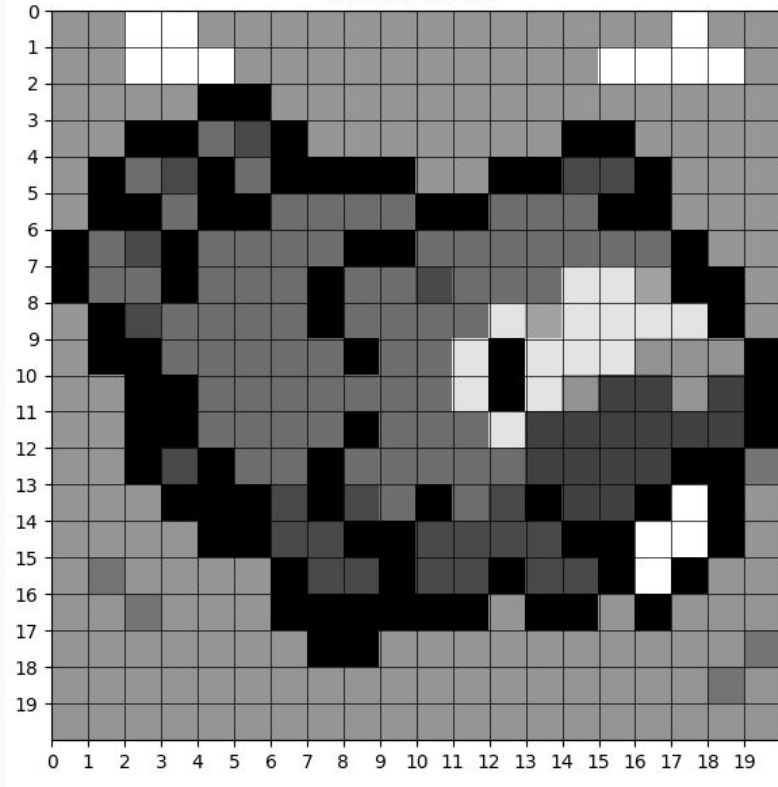
# Convolución



Es desplazar una función (**filtro**) por sobre otra (**datos**) y sumar la multiplicación de ambas



# Convolución 2D



0	150	150	255	255	150	150	150	150	150	150	150	150	150	150	150	150	150	255	150	150
1	150	150	255	255	255	150	150	150	150	150	150	150	150	150	150	150	255	255	255	150
2	150	150	150	150	0	0	150	150	150	150	150	150	150	150	150	150	150	150	150	150
3	150	150	0	0	110	73	0	150	150	150	150	150	150	150	0	0	150	150	150	150
4	150	0	110	73	0	110	0	0	0	0	150	150	0	0	73	73	0	150	150	150
5	150	0	0	110	0	0	110	110	110	110	0	0	110	110	110	0	0	150	150	150
6	0	110	73	0	110	110	110	110	0	0	110	110	110	110	110	110	0	150	150	150
7	0	110	110	0	110	110	110	0	110	110	73	110	110	110	228	228	162	0	0	150
8	150	0	73	110	110	110	110	0	110	110	110	110	228	162	228	228	228	228	0	150
9	150	0	0	110	110	110	110	110	0	110	110	228	0	228	228	228	147	147	147	0
10	150	150	0	0	110	110	110	110	110	110	110	228	0	228	147	65	65	149	65	0
11	150	150	0	0	110	110	110	110	0	110	110	110	228	65	65	65	65	65	65	0
12	149	149	0	73	0	110	110	0	110	110	110	110	110	65	65	65	65	0	0	117
13	149	149	149	0	0	0	73	0	73	110	0	110	73	0	65	65	0	255	0	149
14	149	149	149	149	0	0	73	73	0	0	73	73	73	73	0	0	255	255	0	149
15	149	116	149	149	149	149	0	73	73	0	73	73	0	73	73	0	255	0	149	149
16	149	149	116	149	149	149	0	0	0	0	0	0	0	149	0	0	149	0	149	149
17	149	149	149	149	149	149	149	0	0	149	149	149	149	149	149	149	149	149	149	116
18	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	116	149
19	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149



# Convolución 2D



20x20 Bidoof

0	150	150	255	255	150	150	150	150	150	150	150	150	150	150	150	150	255	150	150
1	150	150	255	255	255	150	150	150	150	150	150	150	150	150	255	255	255	255	150
2	150	150	150	150	0	0	150	150	150	150	150	150	150	150	150	150	150	150	150
3	150	150	0	0	110	73	0	150	150	150	150	150	150	0	0	150	150	150	150
4	150	0	110	73	0	110	0	0	0	0	150	150	0	0	73	73	0	150	150
5	150	0	0	110	0	0	110	110	110	110	0	0	110	110	110	0	0	150	150
6	0	110	73	0	110	110	110	110	0	0	110	110	110	110	110	110	0	150	150
7	0	110	110	0	110	110	110	0	110	110	73	110	110	110	228	228	162	0	150
8	150	0	73	110	110	110	110	0	110	110	110	110	228	162	228	228	228	228	0
9	150	0	0	110	110	110	110	110	0	110	110	228	0	228	228	228	147	147	147
10	150	150	0	0	110	110	110	110	110	110	110	228	0	228	147	65	65	149	65
11	150	150	0	0	110	110	110	110	0	110	110	110	228	65	65	65	65	65	65
12	149	149	0	73	0	110	110	0	110	110	110	110	110	65	65	65	65	0	117
13	149	149	149	0	0	0	73	0	73	110	0	110	73	0	65	65	0	255	0
14	149	149	149	149	0	0	73	73	0	0	73	73	73	73	0	0	255	255	0
15	149	116	149	149	149	149	0	73	73	0	73	73	0	73	73	0	255	0	149
16	149	149	116	149	149	149	0	0	0	0	0	0	149	0	0	149	0	149	149
17	149	149	149	149	149	149	149	0	0	149	149	149	149	149	149	149	149	149	116
18	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	116
19	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149	149
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

3x3 Kernel

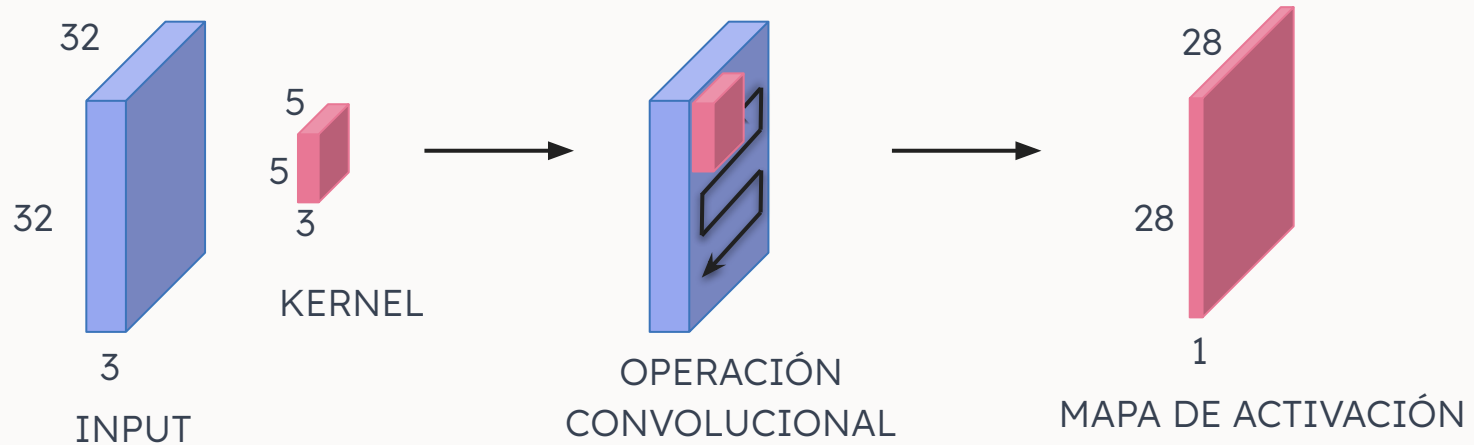
0	1	0	-1
1	1	0	1
2	1	1	1
0	1	2	3

Output

0	750	750	915	660	555	600	750	750	750	750	750	750	855	855	855	960	960		
1	495	345	260	438	438	373	600												
2																			
3																			
4																			
5																			
6																			
7																			
8																			
9																			
10																			
11																			
12																			
13																			
14																			
15																			
16																			
17																			
18																			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	



# Capa convolucional





# Capa convolucional



# Kernel



En una CNN, **kernel** es la matriz de pesos que compone un filtro

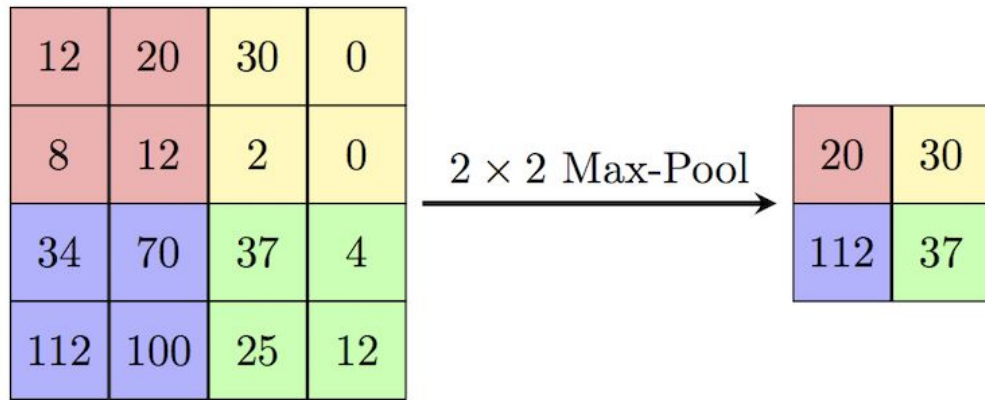
$w_1$	$w_2$	$w_3$
$w_4$	$w_5$	$w_6$
$w_7$	$w_8$	$w_9$

# Pooling



Es una operación que **reduce** la dimensionalidad

En consecuencia, **reduce** el **costo computacional** y el *overfitting*



# Clasificación



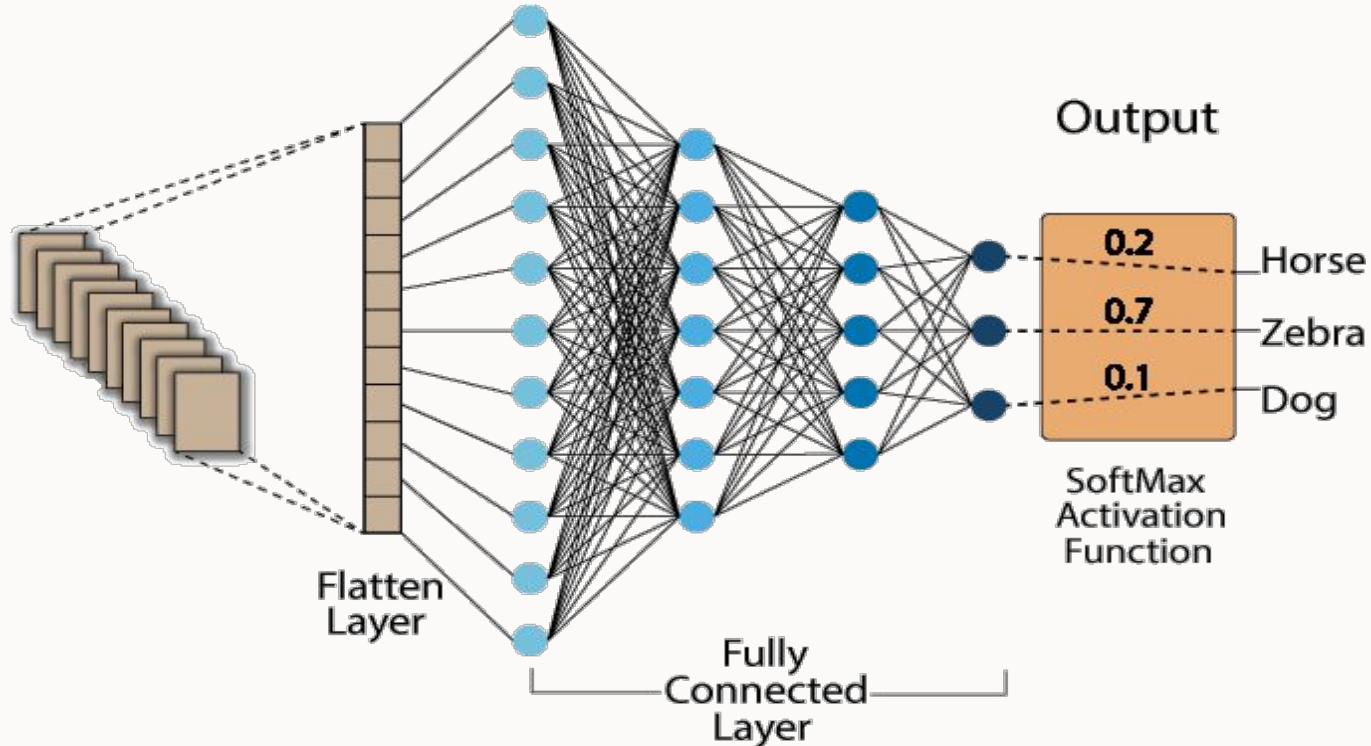
Para tareas de clasificación, necesitamos que la salida sea una capa de **K** neuronas.

**K** es la cantidad de clases.

# Clasificación



Volvemos a los MLP → Aplanamos la matriz



# Aplicaciones



- **Clasificación**
- **Reconocimiento de objetos/caras**
- **Generación de imágenes**
- **Incluso procesamiento de lenguaje natural (NLP)**





# Conclusión

## CNN's

- Más utilizadas con datos de imágenes.
- Se utilizan para grandes bases de datos con mucha información, por sus reducidos parámetros.
- Es intrínsecamente más complejo debido a sus variadas capas.
- Son las redes profundas más eficientes en el análisis de imágenes.

## MLP's

- Más utilizadas con datos tabulados.
- Su número de parámetros crece con los datos, por lo que no siempre son convenientes.
- "Solo" es una red de neuronas.
- Son ineficientes en el análisis de imágenes, y no son invariantes bajo traslaciones.