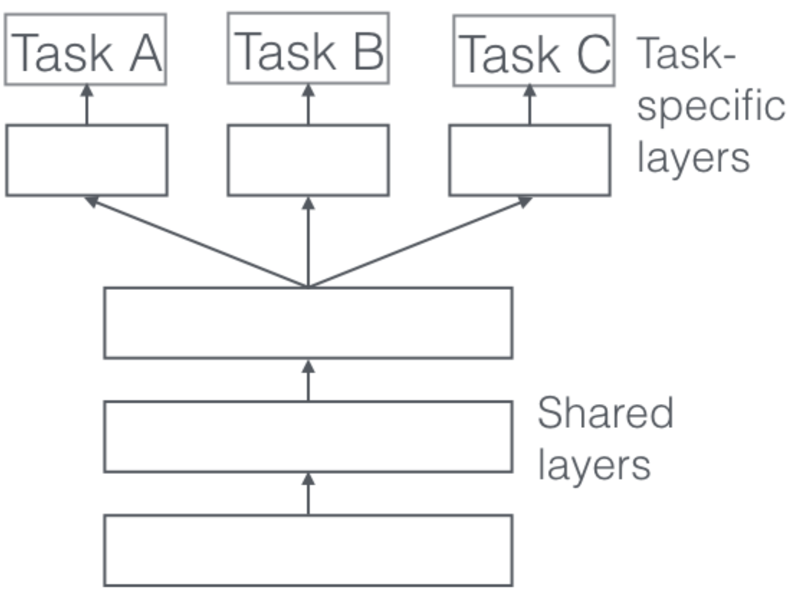
Redes Convolucionales

Compartición de Pesos

Compartición de Pesos

- Es interesante verificar que esta idea se ha manifestado de variadas formas en la literatura, encontrando su expresión más “formal” en el problema denominado *multi-task learning*.



Rich Caruana (1997). Multitask learning. *Machine learning*, *28*(1), 41-75.

Principios Básicos de una CNN

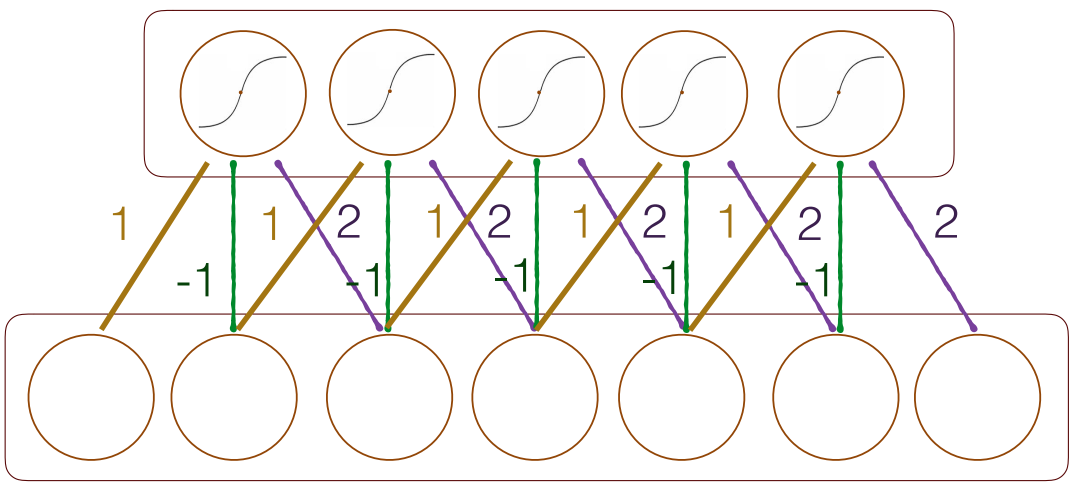
- Las capas de una red convolucional serán capaces de **patrones locales en grandes arreglos multi-dimensionales de datos.** Esto se logrará mediante 3 principios básicos de diseño:

- **Conectividad local.**
- Compartición de pesos.
- Pooling.

- A esto sumaremos un principio transversal de diseño: mantener la topología de los datos a medida que se propaga por la red.

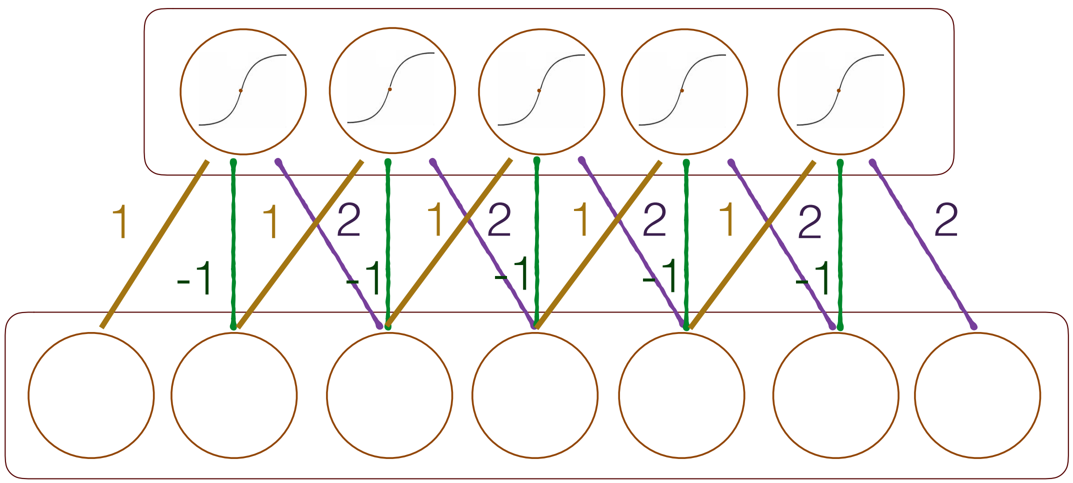
Compartición de Pesos

- Una forma de limitar el número de parámetros de a red, sin podar excesivamente la conectividad, es forzar a un grupo de unidades a **compartir los pesos** es decir agregar al problema de optimización la restricción de que éstos deben ser idénticos.



Canales o Filtros

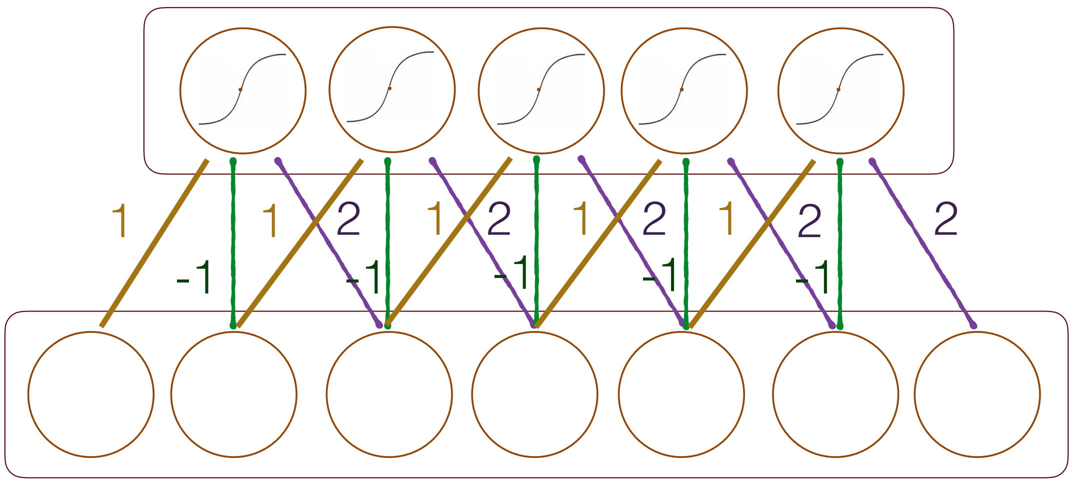
- Compartir pesos nos permitirá definir **grupos de neuronas con campos visuales muy similares entre sí** (stride pequeño, por ejemplo 1). Estas neuronas trabajarán en conjunto para detectar un determinado patrón en diferentes posiciones del volumen entrante.



- Nos referiremos a un grupo como ***un canal*** o ***un filtro***.

Mapas de Características (Feature Maps)

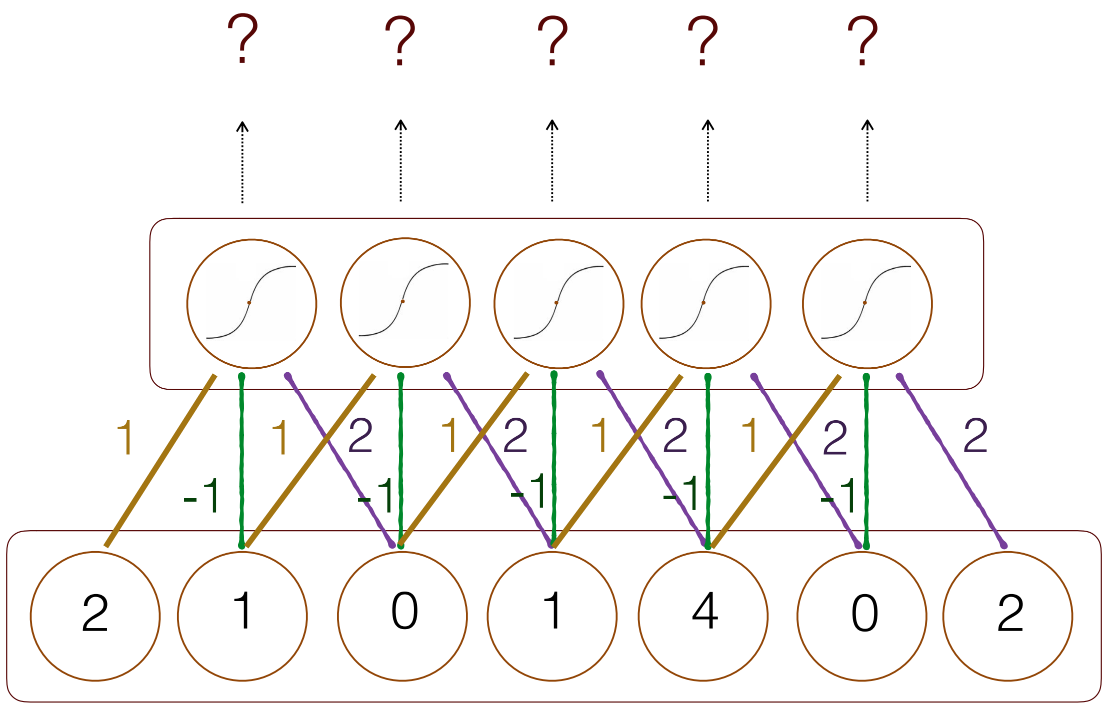
- El número de neuronas del FM se determinará para poder cubrir completamente el patrón de entrada considerando el stride y el tamaño del campo receptivo de canal.



- La activación de este grupo de neuronas se denominará **mapa de características (feature map, FM).** A veces se llama también FM al grupo de unidades en sí mismo.

Mapas de Características (Feature Maps)

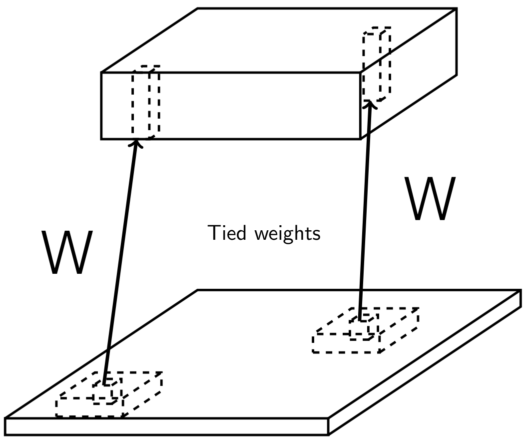
- Claramente, aún si el grupo de unidades comparte pesos, su activación sobre el mismo patrón de entrada no es igual.

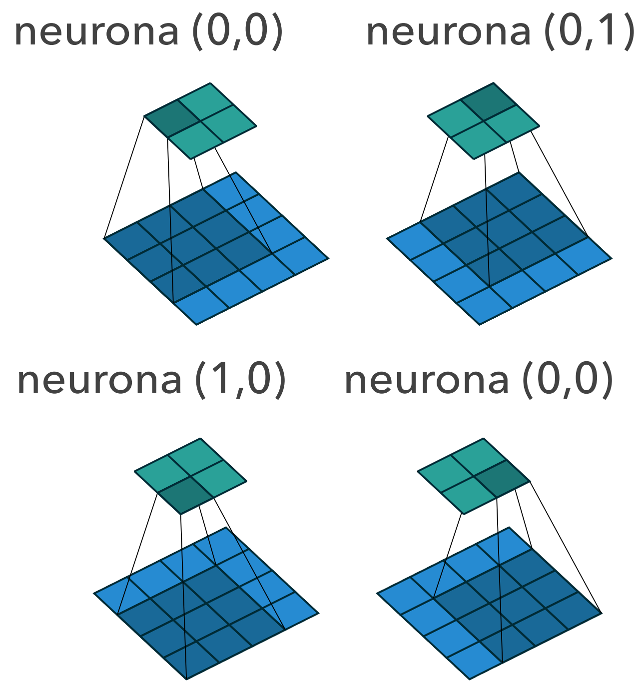


(*) Asuma umbrales de activación 0 y funciones de activación lineales.

Mapas 2D

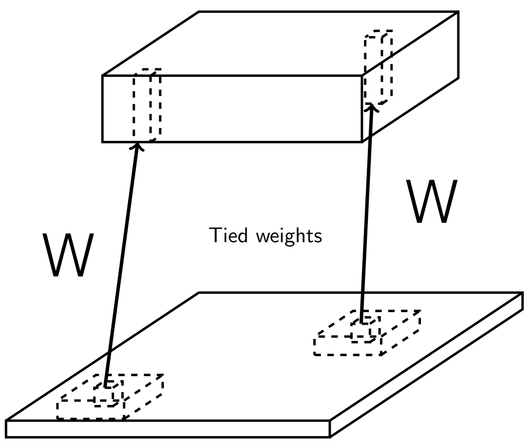
- En el caso bidimensional vale lo mismo: el stride será típicamente unitario en cada dimensión $K_i = K_j = 1$ y el número de neuronas de cada canal se determinará para poder "barrer" completamente el patrón de entrada.

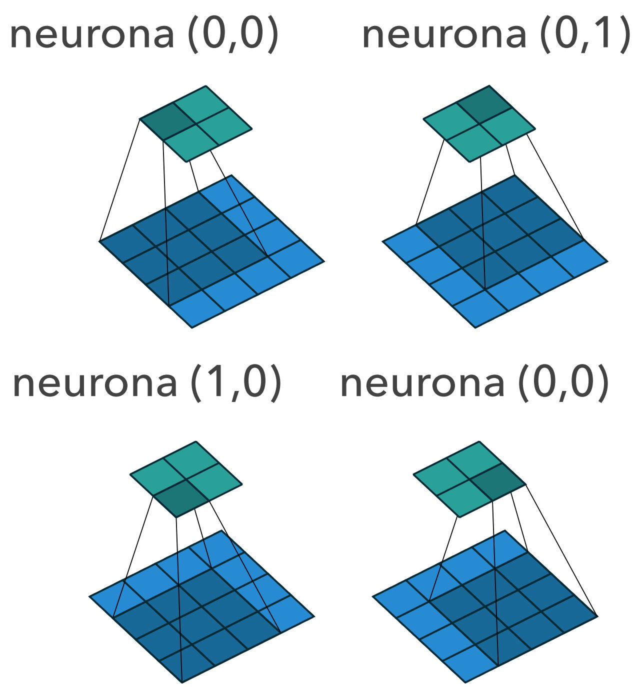




Mapas 2D

- Por consistencia con la forma del campo visual, los “pesos” de cada neurona se dispondrán en una **matriz W de tamaño $S_I x S_j$ idéntico al tamaño de campo receptivo.** Por definición, esta matriz será compartida por todo el grupo de neuronas de un filtro o canal.





Ejemplo

- Consideremos por ejemplo un filtro 2D con campos receptivos de $3x3$ y strides de $1x1$, donde la neurona (i, j) se "mira" los valores entre las posiciones (i, j) y $(i + 2, j + 2)$ del patrón de entrada. Procesaremos un input de $5x5$ y por lo tanto el canal estará formado por **9 unidades dispuestas en un arreglo de $3x3$**.

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

Input

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

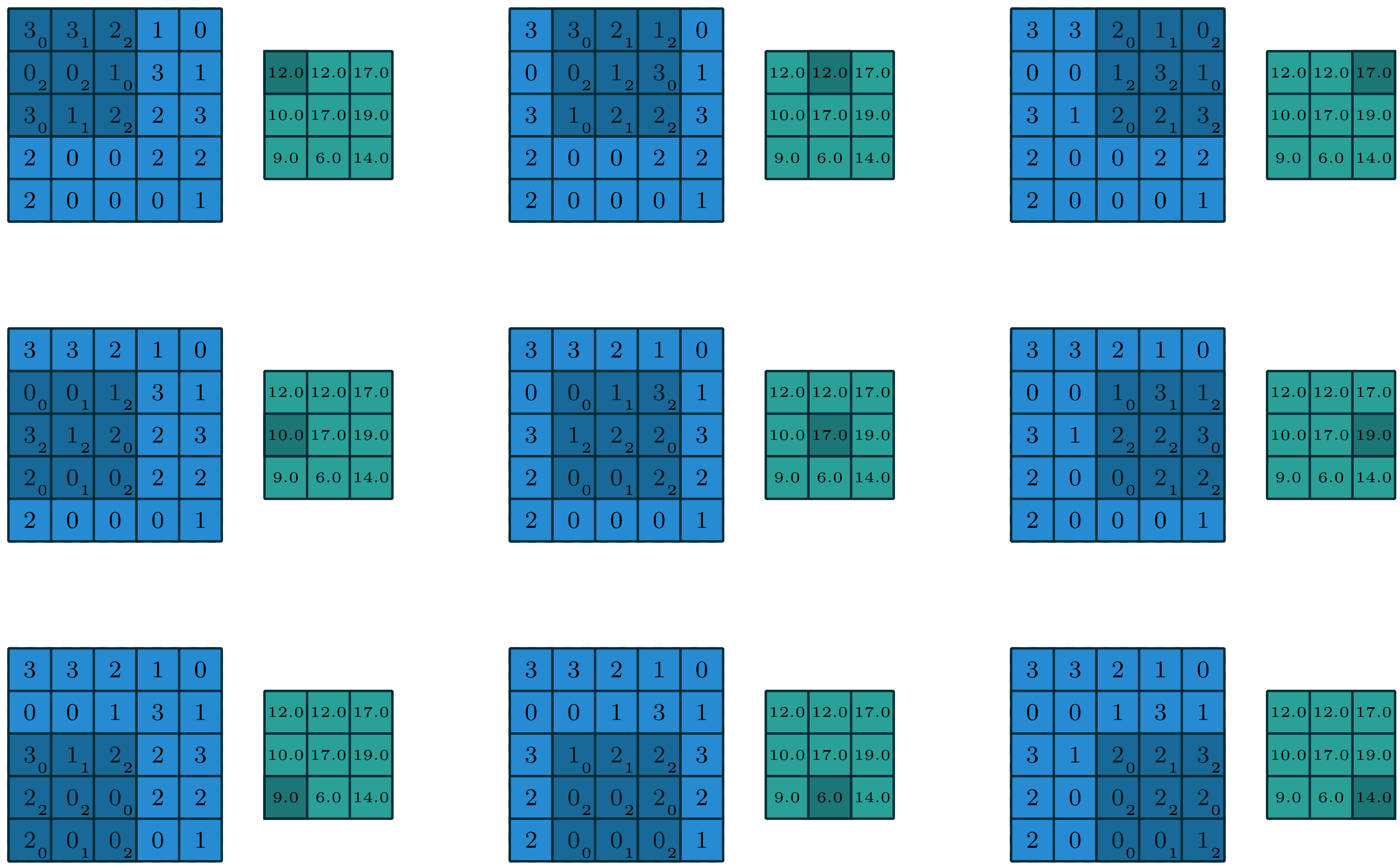
FM

0	1	2
2	2	0
0	1	2

pesos W

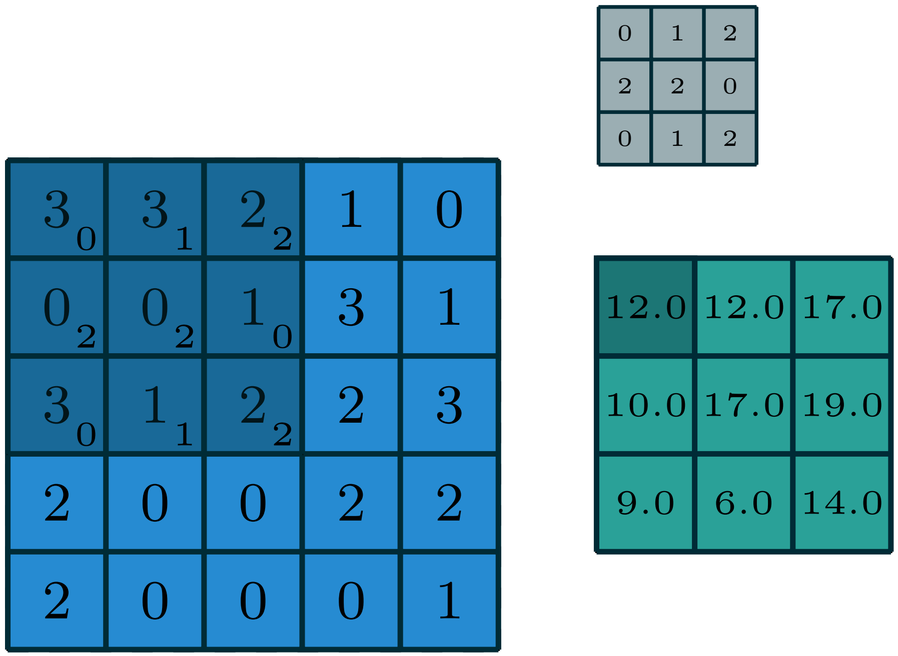
- Asumiremos por simplicidad umbrales de activación 0 y funciones de activación lineales.

Ejemplo



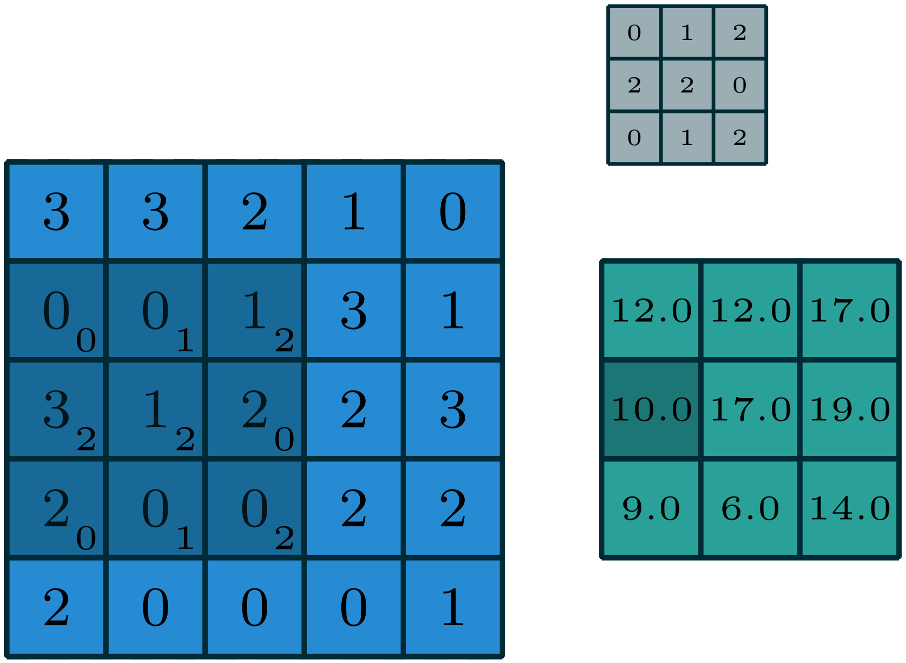
Ejemplo

- Neurona (0,0)



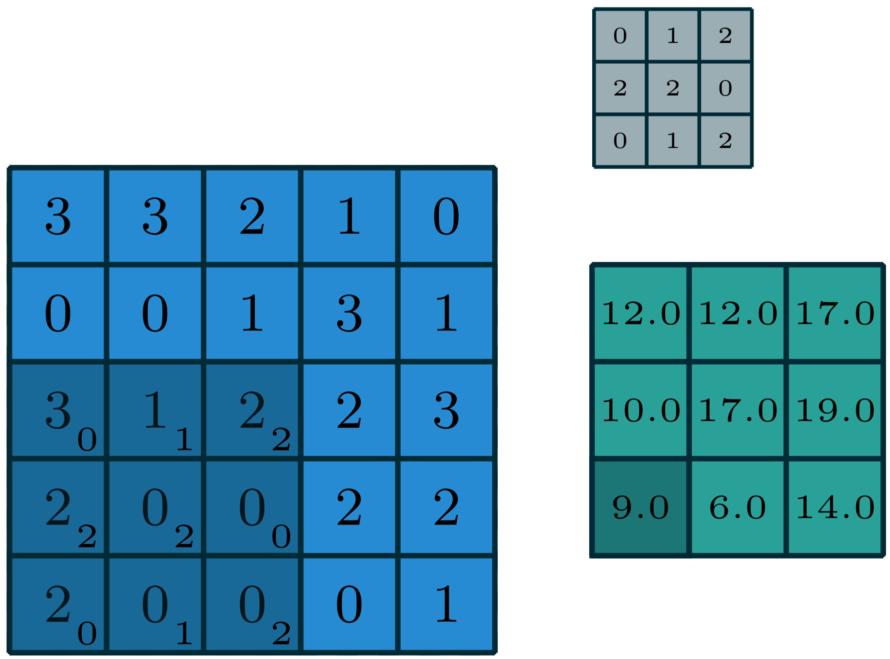
Ejemplo

- Neurona (1,0)



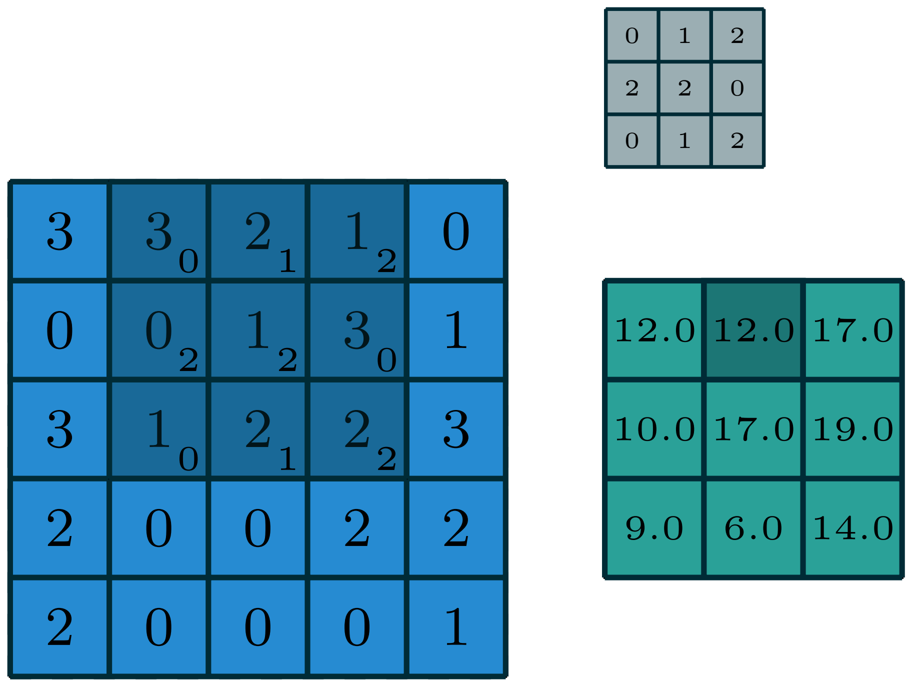
Ejemplo

- Neurona (2,0)

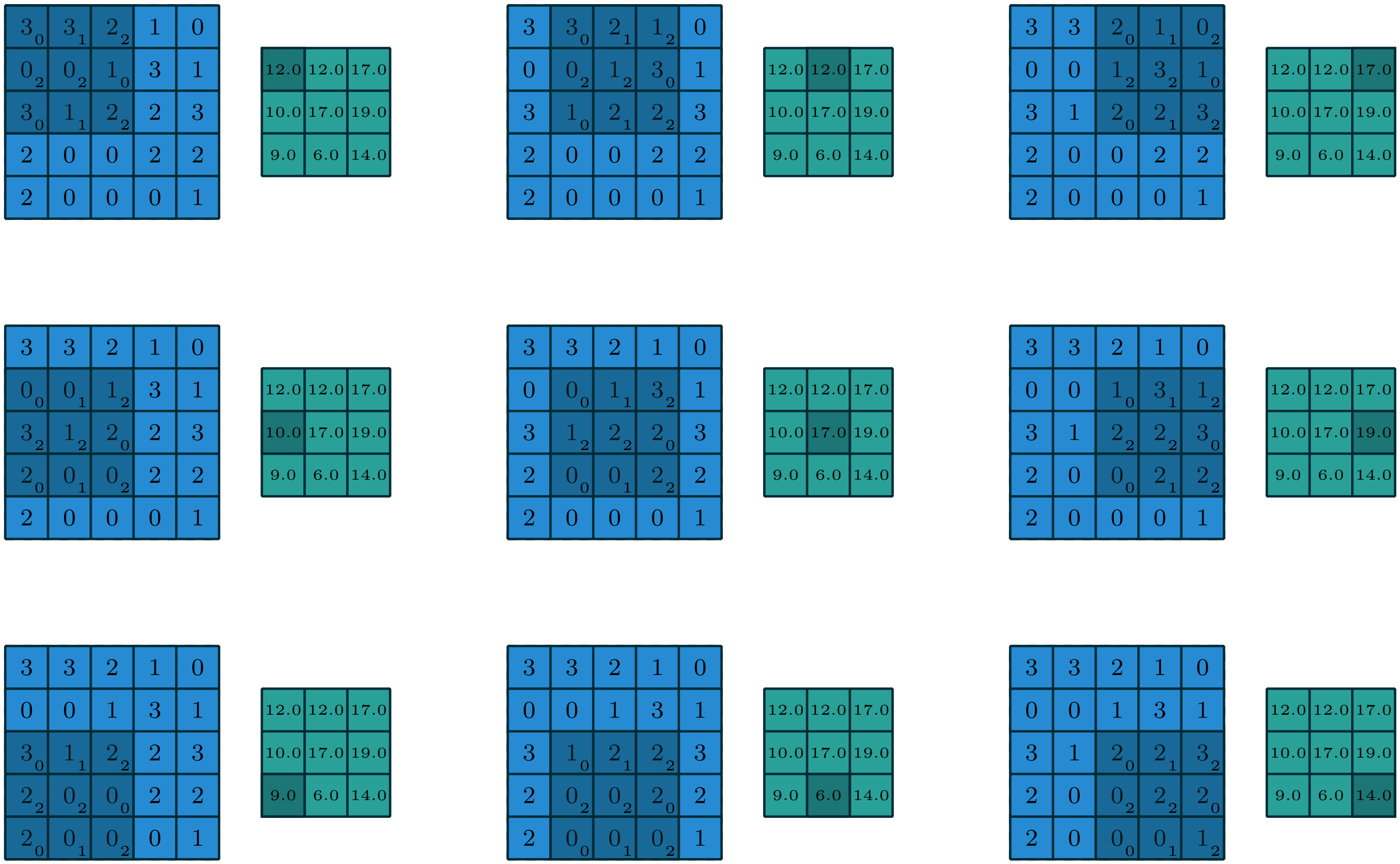


Ejemplo

- Neurona (0,1)



Ejemplo



Entonces …

- Una red convolucional (CNN) se construye sobre 3 ideas fundamentales: conectividad limitada, compartición de pesos y agresiva reducción de dimensionalidad vía pooling.
- La compartición de pesos permite ahorrar aún más parámetros dándonos la oportunidad de usar más unidades y campos receptivos con un buen grado de redundancia (overlap).
- Un grupo de neuronas con pesos compartidos se denominará canal ó filtro y al ser aplicado a volumen de entrada producirá como resultado un mapa de características consistente con la dimensionalidad de la capa.
- En el caso 2D las neuronas de un canal se organizan sobre un arreglo 2D. Sus pesos corresponden a un arreglo 2D (matriz) que tiene la forma del campo receptivo donde se aplicará.
- El número de neuronas de un canal se deduce del tamaño del tamaño del filtro y del stride.