Redes Neuronales Convolucionales

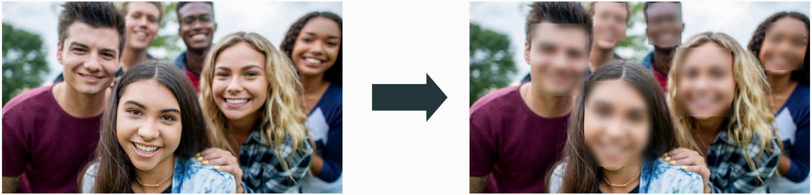
Rafael Villca Poggian

Club de Ciencia de Datos

Contenido

Convolución y Correlación-Cruzada

Ejemplo de Convolución



Convolución

- Es una operación matemática de dos funciones f y g
- Expresa cómo el filtro g modifica a la señal f
- Está definida como:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

- De manera discreta:

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]\, g[n-m]$$

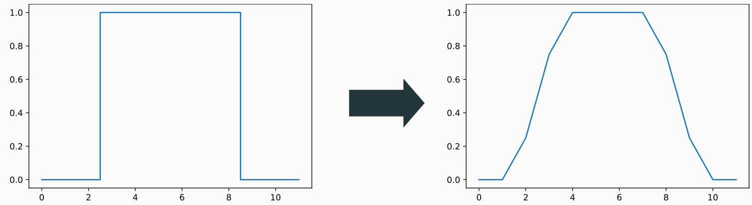
m recorre el conjunto de índices sobre el que está definido el filtro

Convolución 1D

...	0	0	0	1	1	1	1	1	1	0	0	0	...

0.25	0.5	0.25

0	0	0.0											

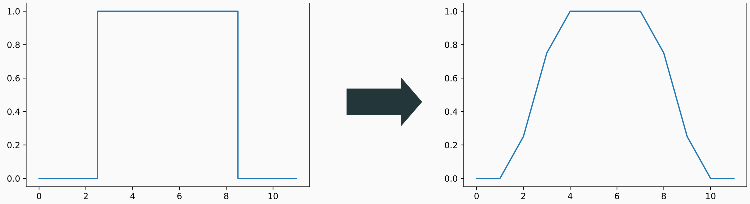


Convolución 1D

...	0	0	0	1	1	1	1	1	1	0	0	0	...

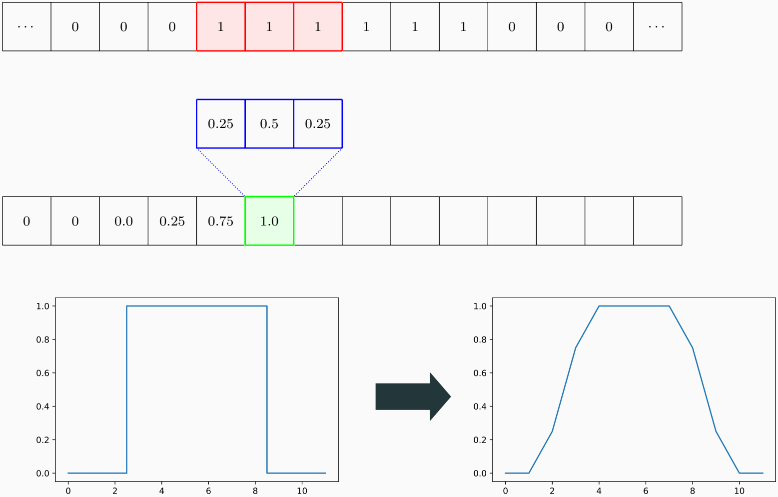
0.25	0.5	0.25

0	0	0.0	0.25										



Convolución 1D

Convolución 1D

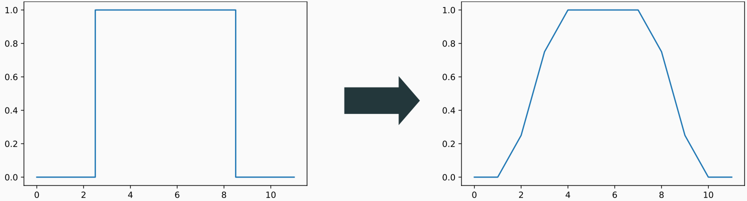


Convolución 1D

...	0	0	0	1	1	1	1	1	1	0	0	0	...

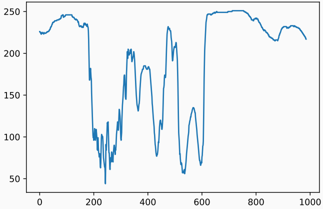
0.25	0.5	0.25

0	0	0.0	0.25	0.75	1.0	1.0	1.0	1.0	0.75	0.25	0.0	0	0

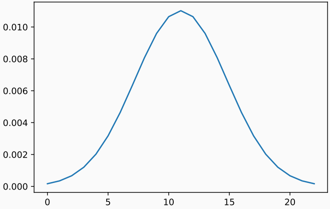


Convolución 1D

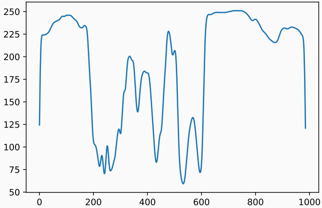
Sea la señal



Y el filtro:



Se obtiene:



Filtro Normal

- Una forma de suavizar señales es aplicando un filtro obtenido de una distribución normal
- Caso bivariante con $\mu = 0$ y $x, y \overset{iid}{\sim} \mathcal{N}$:

$$f\left(x, y\right) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

- Generando un filtro de 23×23 con $\sigma = 3.8$ y $I = \left\{(\mathrm{x}, \mathrm{y}) \in \mathbb{Z}^2\right\}$

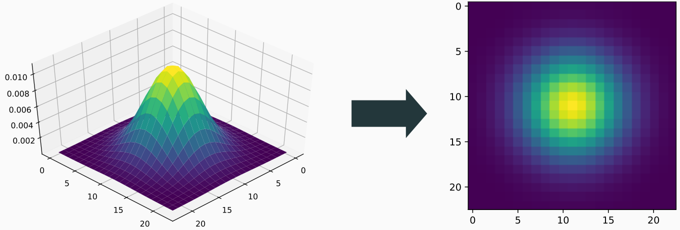
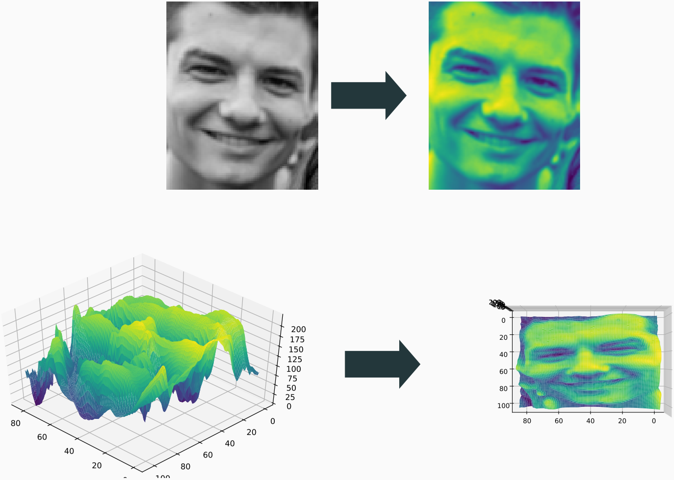


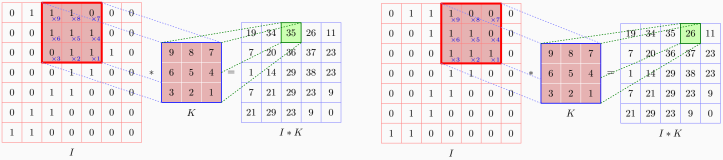
Imagen como señal 2D

- Una imagen se puede interpretar como una señal de dominio $\mathbb{Z}^2$



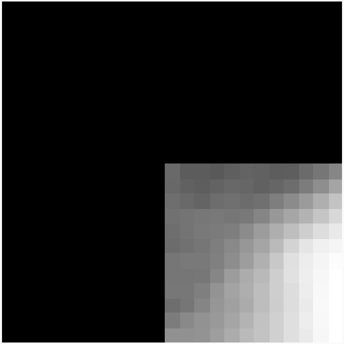
Convolución 2D

- En el caso bidimensional se toma una sección de la imagen de dimensiones $k \times k$.
- $I_{patch} * K = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$
- $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} i & h & g \\ f & e & d \\ c & b & a \end{bmatrix}$
- $(i \cdot 1) + (h \cdot 2) + (g \cdot 3) + (f \cdot 4) + (e \cdot 5) + (d \cdot 6) + (c \cdot 7) + (b \cdot 8) + (a \cdot 9)$



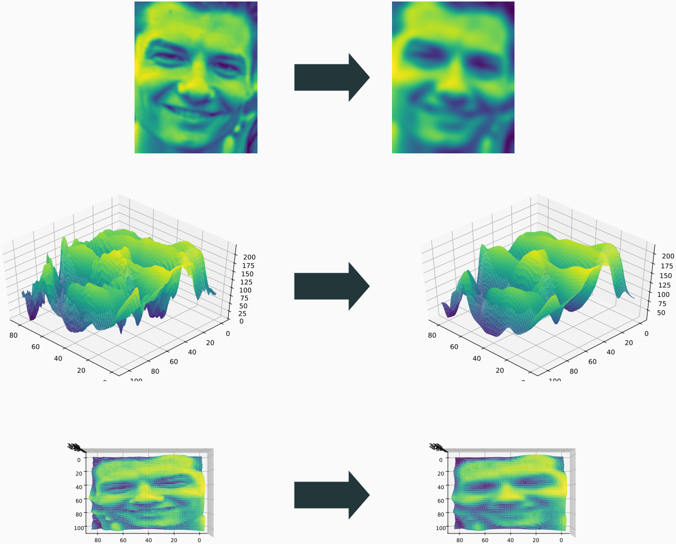
Manipulación de Bordes

Borde cero



Borde reflejado

Convolución 2D



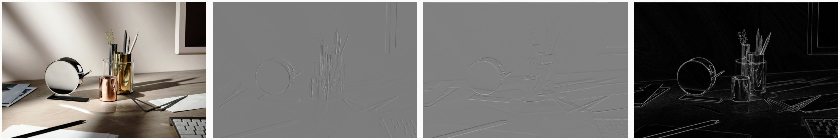
Otros Filtros

···	0	0	0	1	1	1	1	1	1	0	0	0	···

1	0	-1

0	0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	-1.0	-1.0	0.0	0	0

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \& \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$



Correlación Cruzada

- La correlación cruzada se calcula similar a la convolución pero sin invertir los elementos del filtro

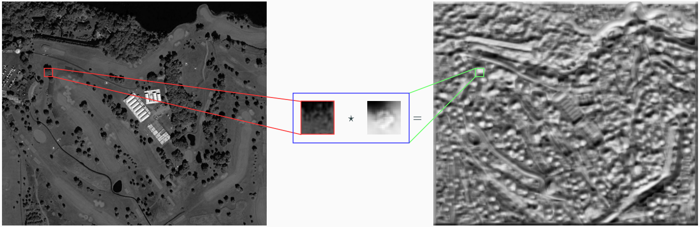
$$I_{patch} \star K = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \star \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

- De manera general

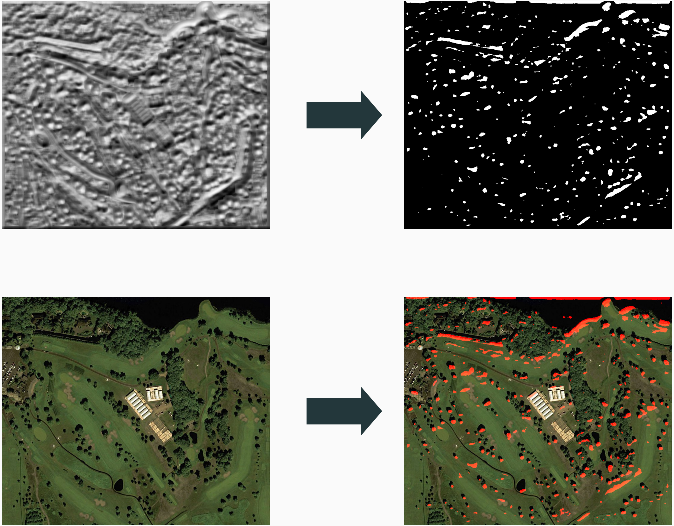
$$(f \star g)(i,j) = \sum_a \sum_b f(a,b) g(i+a, j+b)$$

- Calculando:

Correlación Cruzada



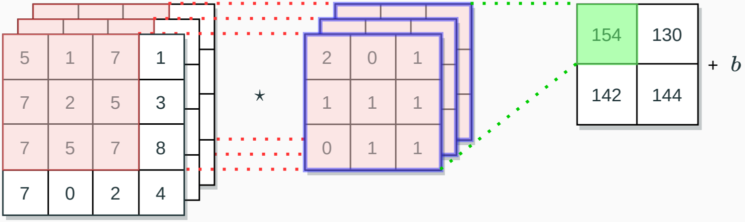
Correlación Cruzada



Capas Convolucionales

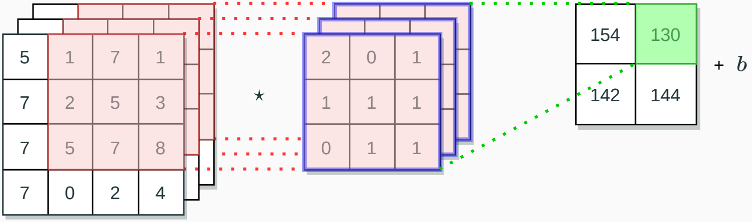
Capas Convolucionales (Correlaciones)

- Se puede aplicar relleno (padding) en los bordes para obtener un capa menos pequeña o igual a la entrada
- Se define como $H_{(l+1,i)} = H_l \star W_i + b_i$
- El filtro W_i se convierte en un tensor de orden 3 con dimensiones $k \times k \times c$ con c el número de canales de la capa anterior.
- Se obtiene un canal por filtro aplicado sobre todos los canales de la capa anterior



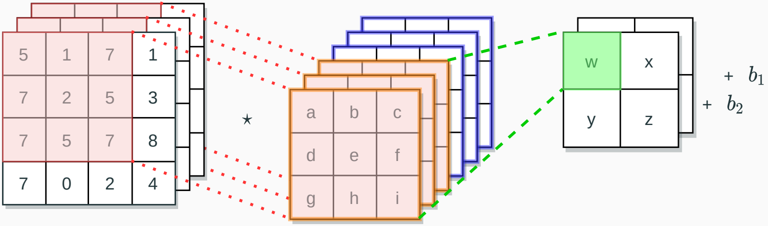
Capas Convolucionales (Correlaciones)

- Se puede aplicar relleno (padding) en los bordes para obtener un capa menos pequeña o igual a la entrada
- Se define como $H_{(l+1,i)} = H_l \star W_i + b_i$
- El filtro W_i se convierte en un tensor de orden 3 con dimensiones $k \times k \times c$ con c el número de canales de la capa anterior.
- Se obtiene un canal por filtro aplicado sobre todos los canales de la capa anterior



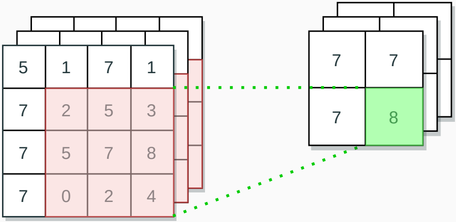
Capas Convolucionales (Correlaciones)

- Se puede aplicar relleno (padding) en los bordes para obtener un capa menos pequeña o igual a la entrada
- Se define como $H_{(l+1,i)} = H_l \star W_i + b_i$
- El filtro W_i se convierte en un tensor de orden 3 con dimensiones $k \times k \times c$ con c el número de canales de la capa anterior.
- Se obtiene un canal por filtro aplicado sobre todos los canales de la capa anterior



Pooling

- La otra operación posible es la agrupación (Pooling)
- Generalmente se aplica Max Pooling
- Se preserva el valor de la activación más grande de una vecindad en la anterior capa



Strides

- Cuando se realiza el barrido por la imagen, el salto entre vecindades se define por el paso (stride)
- Este parámetro, el relleno y el tamaño del filtro o la vecindad definen el tamaño de la salida como:

$$\left(\frac{h_{in} - k_h + p_h + s_h}{s_h}\right) \times \left(\frac{w_{in} - k_w + p_w + s_w}{s_w}\right)$$

h alto, w ancho, p padding, s stride

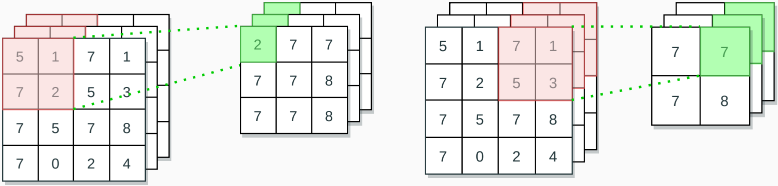
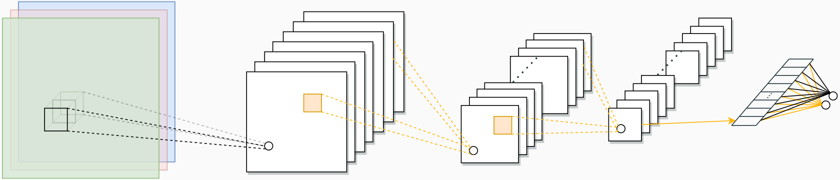


Diagrama de una ConvNet



img 28×28

↓

Conv 3×3, $C_{out} = 6$
$pad = 2$

$tanh(h_1)$

↓

Pool 2×2, $stride = 2$

↓

Conv 5×5, $C_{out} = 16$
$pad = 0$

$tanh(h_3)$

↓

Pool 2×2, $stride = 2$

↓

Fc 120

↓

Fc 84

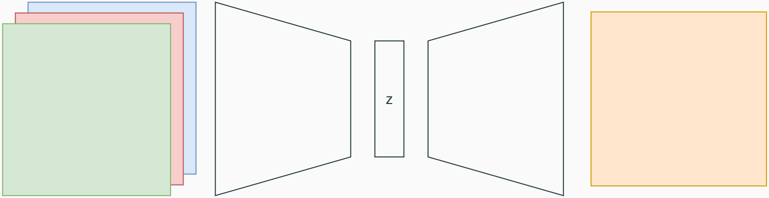
↓

Fc 10

Autoencoders

Encoder-Decoder

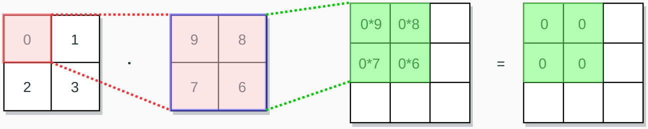
- Al ser las redes neuronales aproximadores universales, se puede utilizar como un codificador o compresor
- De la etapa de codificación se obtiene un vector latente con representaciones sobresalientes de las entradas
- En base a estas representaciones se pueden obtener otras imágenes enfatizando aspectos de la entrada



Convolución Traspuesta

- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

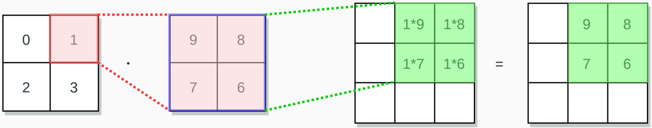
$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$



Convolución Traspuesta

- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

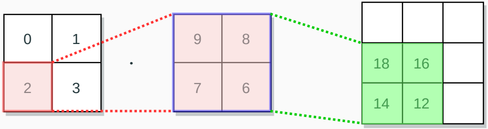
$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$



Convolución Traspuesta

- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

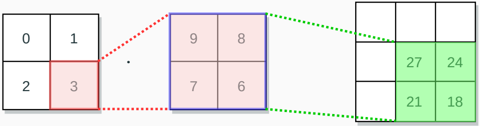
$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$



Convolución Traspuesta

- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$



Convolución Traspuesta

- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$

$$\begin{array}{|c|c|c|}\hline 0 & 0 & \\ \hline 0 & 0 & \\ \hline & & \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline & 9 & 8 \\ \hline & 7 & 6 \\ \hline & & \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline & & \\ \hline & 27 & 24 \\ \hline & 21 & 18 \\ \hline \end{array} + \begin{array}{|c|c|c|}\hline & & \\ \hline 18 & 16 & \\ \hline 14 & 12 & \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 0 & 9 & 8 \\ \hline 18 & 50 & 30 \\ \hline 14 & 33 & 18 \\ \hline \end{array}$$

Convolución Traspuesta

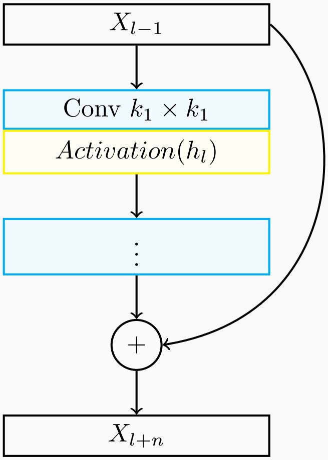
- En el decodificador se aplican deconvoluciones para obtener canales más grandes dada la capa anterior
- Las dimensiones de salida son:

$$((h_{in}-1)\cdot s_h - 2p_{in_h} + k_h + p_{out_h}) \times ((w_{in}-1)\cdot s_w - 2p_{in_w} + k_w + p_{out_w})$$

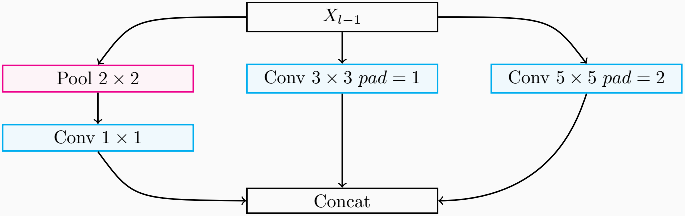
$$\begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} \cdot \begin{bmatrix} 9 & 8 \\ 7 & 6 \end{bmatrix} = \begin{bmatrix} 0 & 9 & 8 \\ 18 & 50 & 30 \\ 14 & 33 & 18 \end{bmatrix}$$

Capas Especiales

Saltos



Capas Concatenadas



Gracias