



Encuentro sincrónico 4

Redes convolucionales y sus aplicaciones

Agenda

Temas:

- Redes neuronales a partir de convoluciones.
- Arquitecturas convolucionales.
- Algunas aplicaciones.



Table of Contents

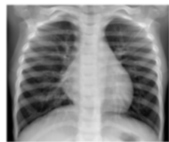
► Redes neuronales convolucionales

► Arquitecturas Convolucionales

► Otras Arquitecturas

Aprendizaje Automático versus Aprendizaje Profundo I

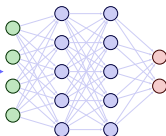
Aprendizaje Automático



Entrada



Características Manuales

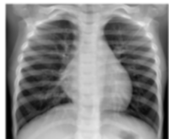


Clasificador

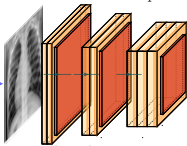
Neumonía
o Normal

Salida

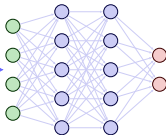
Aprendizaje Profundo



Entrada



Características Automáticas
(Aprendizaje Profundo)



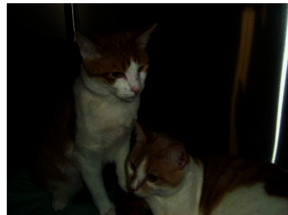
Clasificador

Neumonía
o Normal

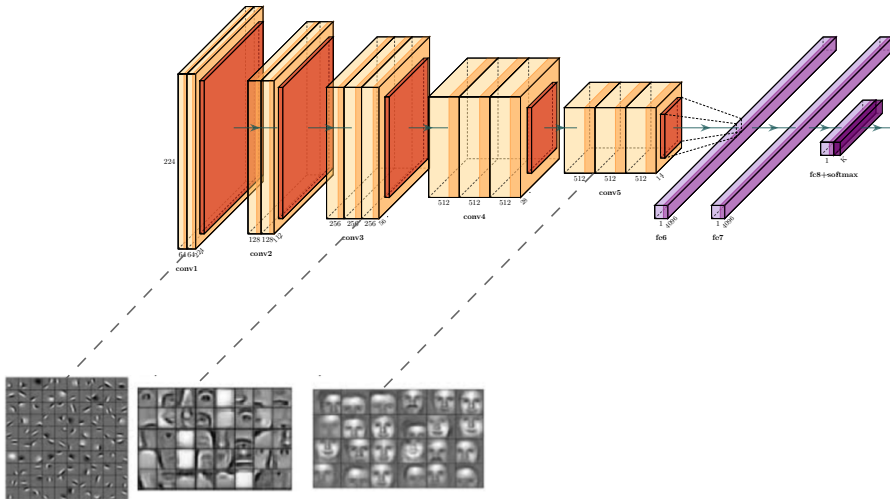
Salida

Aprendizaje Automático versus Aprendizaje Profundo II

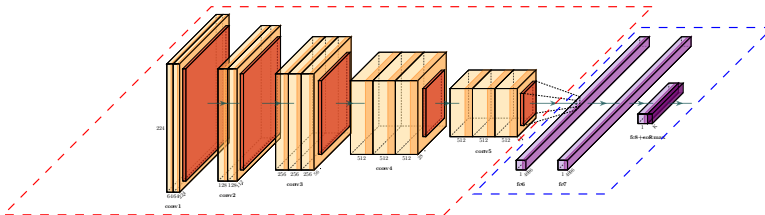
- Requiere que una persona extraiga, manualmente, características desde las imágenes.
- Esas características alimentan un sistema de aprendizaje automático clásico (fully-connected, SVM, KNN).
- Las características extraídas suelen ser ajustadas al problema. Cualquier variación en las entradas puede afectar el desempeño del modelo.



Aprendizaje Automático versus Aprendizaje Profundo III



Aprendizaje Automático versus Aprendizaje Profundo IV



```

model = keras.Sequential([
    keras.layers.Conv2D(filters = 32, kernel_size = (3, 3), padding="valid",
                        strides=(1, 1), activation='relu', input_shape=(28, 28, 1)),
    keras.layers.MaxPooling2D(pool_size = (2, 2), strides=(2, 2)),

    keras.layers.Conv2D(filters = 64, kernel_size = (3, 3), padding="valid",
                        strides=(1, 1), activation='relu'),
    keras.layers.MaxPooling2D(pool_size = (2, 2), strides=(2, 2)),

    keras.layers.Flatten(),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(256, activation='relu')
    keras.layers.Dense(128, activation='relu')
    keras.layers.Dense(10, activation='softmax')
])
  
```



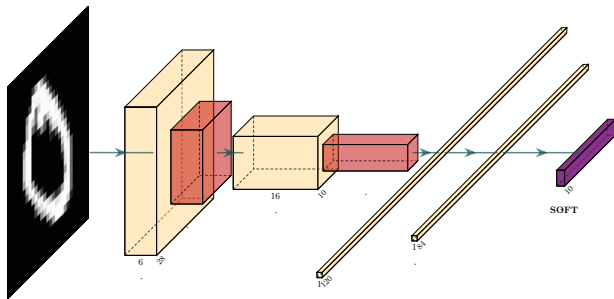
Table of Contents

► Redes neuronales convolucionales

► **Arquitecturas Convolucionales**

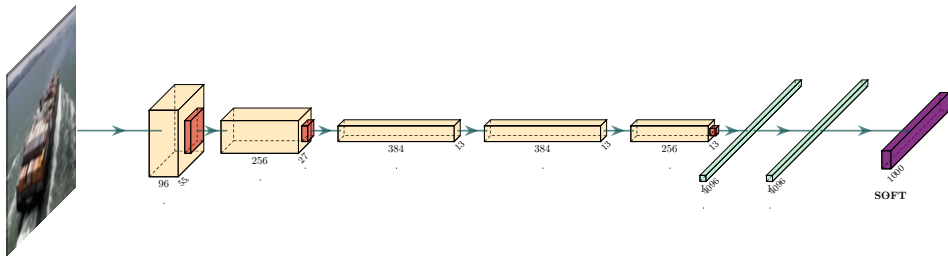
► Otras Arquitecturas

LeNet (Yann LeCun-1998)

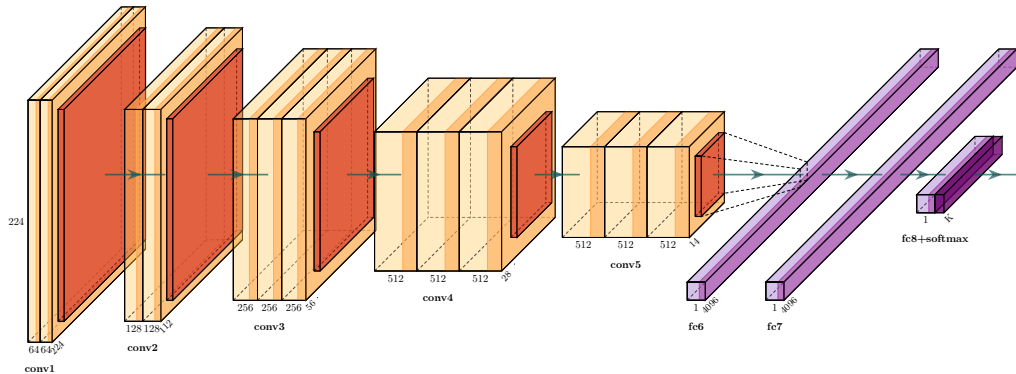


1. Conv1: 5×5 (stride = 1) [Padding="Same"].
2. PoolLayer1: 2×2 (stride = 2).
3. Conv2: 5×5 (stride = 1) [Padding="Same"].
4. PoolLayer2: 2×2 (stride = 2).
5. Dense1: 120.
6. Dense2: 84.
7. Dense3: 10.

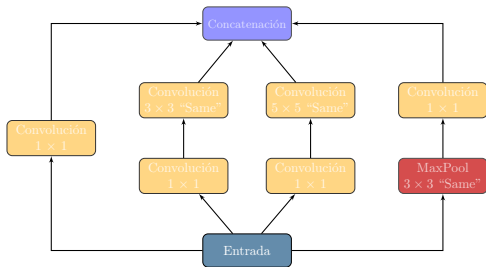
AlexNet (Alex Krizhevsky-2012)



VGG16 (2014)

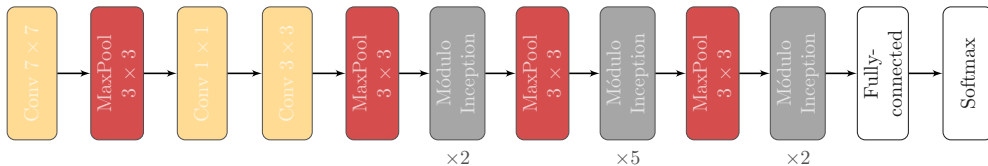


GoogLeNet: inception modules

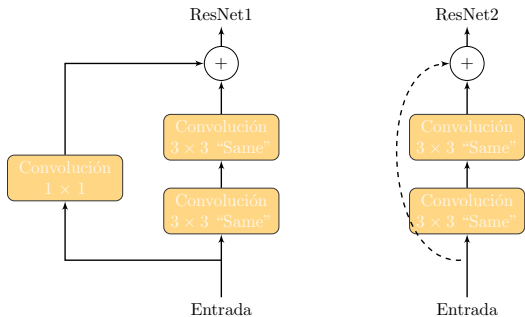


1. Uno de las decisiones a la hora de definir una red convolucional es qué tamaño del filtro utilizar. ¿ 3×3 ? ¿ 5×5 ? ¿Debo usar max pooling?
2. Una solución son los módulos "inception". Combinan los resultados de convoluciones con diferentes tamaños del filtro y una operación de max pooling.

GoogLeNet (2014)



ResNet: residual block



1. Los bloques residuales permiten entrenar redes más profundas.
2. Los saltos en las conexiones permite que los caminos de las derivadas sean más cortos. Así, se reduce el riesgo de gradientes que se desvanecen o explotan.

ResNet18 (2015)

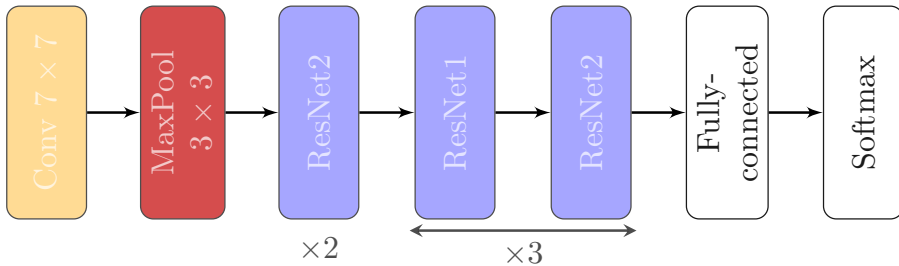


Table of Contents

- ▶ Redes neuronales convolucionales
- ▶ Arquitecturas Convolucionales
- ▶ Otras Arquitecturas

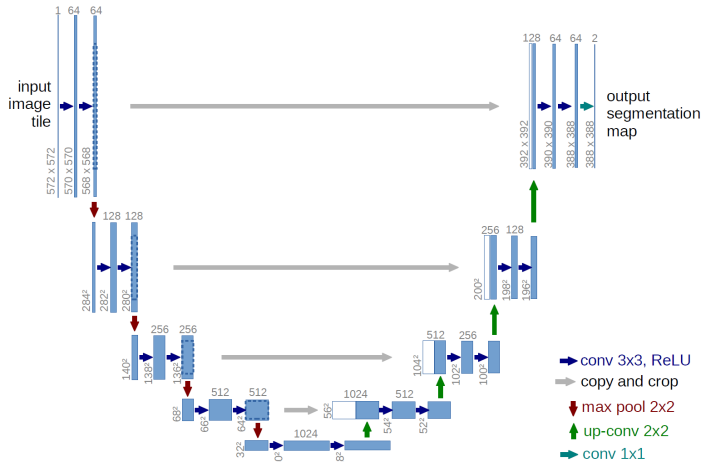
Convolución Transpuesta

$$\begin{array}{|c|c|} \hline 0 & 1 \\ \hline 2 & 3 \\ \hline \end{array} \otimes^T \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 3 & 4 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline 0 & 3 & 4 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 2 & 4 & 0 \\ \hline 6 & 8 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 3 & 6 \\ \hline 0 & 9 & 12 \\ \hline \end{array}$$

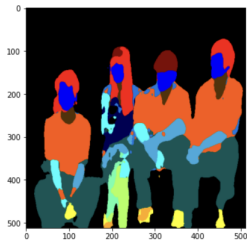
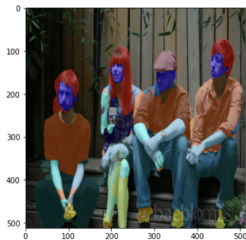
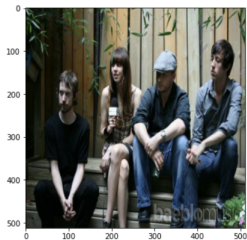
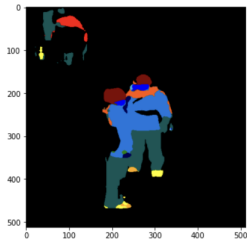
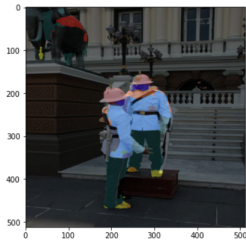
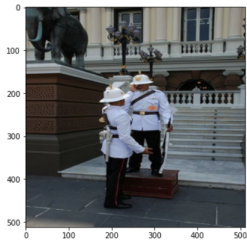
$$= \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline 2 & 10 & 10 \\ \hline 6 & 17 & 12 \\ \hline \end{array}$$

- La convolución transpuesta no es equivalente a la convolución inversa. Equivale a un sobremuestreo.
- Equivale a realizar un padding con $(f-1, f-1)$, donde f es el tamaño del filtro.

UNet




Segmentación Semántica



Segmentación Semántica



Bibliografía

- 
- Ronneberger, O., Fischer, P., Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18 (pp. 234-241). Springer International Publishing.