

## **CNN ( Convolutional Neural Networks) Homework**

**1.- ¿Qué es lo que entendiste por redes neuronales convolucionales en tus propias palabras?**

**2.- ¿Qué respuesta explica mejor la convolución?**

- a) Es el primer paso para usar Redes Neuronales.
- b) Detecta características clave en imágenes, respetando sus límites espaciales.
- c) Es una técnica para estandarizar un dataset
- d) Sirve para comprender los features del modelo y seleccionar las mejores features

**3.- ¿Qué respuesta explica mejor el ReLU?**

- a) Ayuda en la detección de características, aumentando la no linealidad de la imagen, convirtiendo píxeles positivos a cero. Este comportamiento le permite detectar variaciones de atributos.
- b) Se utiliza para encontrar las mejores características considerando su correlación.
- c) Una técnica que te permite encontrar valores atípicos.
- d) Ayuda en la detección mejores features en las imágenes, disminuyendo la no linealidad de la imagen, convirtiendo los píxeles negativos a cero. Este comportamiento le permite detectar variaciones de atributos.

Por lo general, una imagen es muy no lineal, lo que significa valores de píxeles muy variados. En esta situación es muy difícil para el algoritmo pueda hacer predicciones correctas. ReLU viene a disminuir la no linealidad y facilitar el trabajo.

**4. ¿Qué respuesta explica mejor el Pooling?**

- a) Ayuda en la detección de features, incluso si están distorsionadas, además de disminuir el tamaño de los features, lo que resulta en un menor uso de recursos computacionales. También es muy útil para extraer features dominantes.
- b) Disminuya el tiempo de cálculo de las funciones, para disminuir la potencia computacional que se necesita.
- c) Crea un dataset para mejorar la precisión del algoritmo que predice imágenes.
- d) Ayuda en la detección de features distorsionados, con el fin de encontrar atributos dominantes.

## 5. ¿Qué respuesta explica mejor el Flattening?

- a) Es el último paso de CNN.
- b) Transforma imágenes en vectores para facilitar la predicción.
- c) Una vez que tenemos los feature maps agrupados, este componente transforma la información en un vector. Es la información que necesitamos para seguir adelante con las capas de redes neuronales artificiales.
- d) Elimina las funciones innecesarias para que nuestro dataset sea más limpio.

En el procedimiento de **Flattening**, básicamente tomamos los elementos de feature maps aprendidos agrupados y los colocamos en forma vectorial. Esta se convierte en la capa de entrada para la próxima ANN.

## 6. ¿Qué respuesta explica mejor la Fully Connected?

- a) Es el último paso de CNN, donde conectamos los resultados de los componentes anteriores para crear una salida.
- b) Fully Connected actúa colocando diferentes pesos en cada sinapsis para minimizar los errores. No se necesita iteración, ya que podemos obtener los mejores resultados en nuestro primer intento.
- c) Es un componente que conecta diferentes algoritmos para aumentar la precisión.
- d) Fully Connected actúa colocando diferentes pesos en cada sinapsis para minimizar los errores. Este paso se puede repetir hasta que se logre el resultado esperado.

Funciona como una ANN, asignando pesos aleatorios a cada sinapsis, la capa de entrada se ajusta por sus pesos y se pone en una función de activación. La salida de esto luego se compara con los valores reales y el error generado se propaga hacia atrás, es decir, los pesos se reajustan y todos los procesos se repiten. Esto se hace hasta que se minimiza la función de costo.

## 7. ¿Cuáles son los tipos de Pooling? ¿Cuáles son sus características?

- a) **Max. Pooling y Min. Pooling.** Max. Pooling devuelve el valor máximo de la parte cubierta por el kernel y suprimen los ruidos, mientras que la Min. Pooling solo devuelve el valor más pequeño de esa parte.
- b) **Max. Pooling y Average Pooling.** Max. Pooling devuelve el valor máximo de la parte cubierta por el kernel y suprime los ruidos, mientras que Average Pooling solo devuelve la medida de esa parte.
- c) **Max. Pooling y Std Pooling.** Max. Pooling devuelve el valor máximo de la parte cubierta por el kernel, mientras que la Std. Pooling devuelve la desviación estándar de esa parte.
- d) **Max. Pooling y Mean Pooling.** Max. Pooling devuelve el valor máximo de la parte cubierta por el kernel, mientras que Mean Pooling devuelve la medida de esa parte y suprime los ruidos.

Se recomienda utilizar Max Pooling la mayor parte del tiempo.

**8. CNN se divide en dos grandes pasos. Feature Learning y clasificación. ¿Qué sucede en cada paso?**

- a) Durante el feature learning, CNN utiliza algoritmos apropiados, mientras que durante la clasificación cambia el algoritmo para lograr el resultado esperado.
- b) Feature Learning tiene componentes Flattening y Full Connection, con innumerables iteraciones entre ellos antes de pasar a Classification, que utiliza los componentes Convolution, ReLU y Pooling.
- c) Feature Learning tiene componentes Convolution, ReLU y Pooling, con innumerables iteraciones entre ellos antes de pasar a Classification, que utiliza los componentes Flattening y Fully Connected.

Durante el Feature Learning , el algoritmo está aprendiendo sobre su dataset. Componentes como Convolution, ReLU y Pooling funcionan para este paso. Una vez que se conocen los features, la clasificación se realiza mediante los componentes Flattening y Fully Connected.

**9. ¿Cuál es la diferencia entre CNN y ANN?**

- a) CNN usa un algoritmo más simple que ANN.
- b) CNN es la forma más sencilla de utilizar las redes neuronales.
- c) Se complementan entre sí, por lo que para usar ANN, debe comenzar con CNN.
- d) CNN tiene una o más capas de unidades de convolución o pooling, que reciben su entrada de múltiples unidades.

La única diferencia es el componente convolucional, que es lo que hace que CNN sea bueno para analizar y predecir datos como imágenes. Los otros pasos son los mismos.

**10. ¿Cuál es el beneficio de usar CNN en lugar de ANN?**

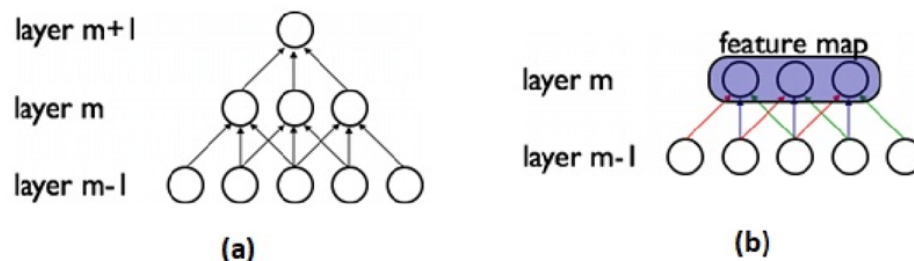
- a) No hay ningún beneficio, ANN siempre es mejor.
- b) Aumenta el número de unidades en la red, lo que significa más parámetros para aprender y aumenta la posibilidad de overfitting. También considera la información detalles pequeños de una imagen. Esta característica es muy importante para lograr una mejor predicción.
- c) Reducir el número de unidades en la red, lo que significa menos parámetros para aprender y menos posibilidades de overfitting. Las CNN también consideran la información de detalles pequeños de un dataset. Esta característica es muy importante para lograr una mejor predicción en datos como imágenes.
- d) CNN tiene mejores resultados ya que tienes más poder de compulsión.

Dado que las imágenes digitales son un montón de píxeles con valores altos, tiene sentido usar CNN para analizarlas. CNN disminuye sus valores, lo cual es mejor para la fase de entrenamiento con menos poder computacional y menos pérdida de información.

### 11. ¿Qué significa "pesos compartidos o Shared Weights" en CNN?

- a) Compartir pesos entre las funciones hace que sea más fácil y rápido predecir la imagen correcta por parte de CNN.
- b) Calcula los pesos de las características y los compara con otros algoritmos para encontrar los mejores parámetros.
- c) Significa que CNN usa los pesos de cada característica para encontrar el mejor modelo para hacer predicciones, compartir los resultados y devolver el promedio.
- d) Es lo que hace que CNN sea "convolucional". Obligando a las neuronas de una capa a compartir pesos, el pase directo se convierte en el equivalente de convolucionar un filtro sobre la imagen para producir una nueva imagen. Luego, en la fase de entrenamiento se convierte en una tarea de aprendizaje de filtros, que deciden qué features debe buscar en los datos.

Esta característica es lo que hace que CNN sea mejor para analizar imágenes que ANN. El componente convolucional de CNN simplifica las estructuras de las imágenes y el algoritmo puede predecir mejor. En lugar de que todas los nodos de una red neuronal estén conectados, unos a otros como sería de manera tradicional (Sparse Connectivity), en una CNN los Shared Weights están conectados solo a las partes importantes de una imagen o del conjunto de datos a aprender.



a) Sparse Connectivity, b) Shared Weights

**12. Con una imagen de entrada que se ha convertido en una matriz de tamaño 12 X 12 junto con un filtro de tamaño 3 x 3 con un stride de 1. Determine el tamaño de la matriz resultante convolucionada. (utilizar la FÓRMULA )**

$$\text{Output\_shape} = C = (N - F + 2P) / S + 1$$

**Donde:**

**C:** is the size of the convoluted matrix.

**N:** is the size of the input matrix.

**f:** is the size of the filter matrix.

**p:** is the Padding amount.

**s:** is the Stride applied.

Aquí:  $n = 12$ ,  $f = 3$ ,  $p = 0$ ,  $s = 1$

**13. Explica los términos "Valid Padding" y "Same Padding" en CNN.**

**14. Listar las arquitecturas de redes convolucionales más conocidas por años desde 2012 (AlexNet) hasta 2020 (Efficient Net) junto con el número de parámetros que tiene, e.g. LeNet aproximadamente 44000**

**15. Notebook 1. Repaso de LeNet**

Correr de nuevo el notebook mostrado en la clase de LeNet de principio a fin para entender el proceso de cómo se entrena una CNN desde scratch utilizando PyTorch

<https://colab.research.google.com/drive/1VYeb2QFKrx4kuslsYH2Yvone0iEZwAOC?usp=sharing>

**16. Notebook 2. (Fine tuning)**

Utilizando la base del notebook mostrado a continuación, entrenamos un clasificador de **perros, gatos, aves, ratones y tiburones** utilizando la librería de FastAI y un modelo pre entrenado a elección, e.g. **resnet18**

- Las imágenes de perros, gatos, aves y ratones pueden ser descargadas con las instrucciones dentro del notebook
- Descargar 100 imágenes por cada categoría.

- Asegurarse que se está corriendo el colab sobre GPU

<https://colab.research.google.com/drive/1MYUM-MRLhTjBfe9TXB5MPgJdPgvrTfe?usp=sharing>

Para saber mas de como funciona FASTAI

Revisar su libro que está en github y en libgen.is, en especial el capítulo de CNN

- [https://github.com/fastai/fastbook/blob/master/06\\_multicat.ipynb](https://github.com/fastai/fastbook/blob/master/06_multicat.ipynb)
- <http://library.lol/main/2668D405E21C8708BB3192CDD92D8DAF>

**Otras referencias.**

- <https://www.kaggle.com/paulburns/fastai-malaria-image-classification-resnet50>
- <https://www.kaggle.com/dataraj/fastai-tutorial-for-image-classification>