

Redes Convolucionales MNIST CIFART-10

Alejandra Verónica López Chiquito
251@alumnos.uaq.mx
Dr. Sebastian Salazar Colores
Deep Learning
10/Octubre/2023

I. INTRODUCCIÓN

Las redes convolucionales son uno de los temas más populares en deep learning. El objetivo de estos modelos se centra en encontrar patrones en los mapas de características que se le proporcionan para así tener la precisión más alta en sus predicciones.

En este ejercicios se planea comparar la una red neuronal convolucional con forma Lenet-5 contra una Densa y una convolucional propuesta. Se abordará de forma introductoria algunos conceptos de una arquitectura de una red neuronal convolucional así como su funcionamiento y la descripción de las bases de datos empleadas en este documento.

II. DESARROLLO

La librería Keras ofrecen algunas bases de datos que pueden ser llamadas con unas cuantas líneas de código con fines de facilitar datos de entrenamiento y validación a las personas que se encuentran aprendiendo herramientas de machine learning o deep learning. En este documento se van a realizar algunas aplicación de redes convolucionales en las bases de datos MNIST y CIFART-10.

Se proponer aplicar a cada una de estas base de datos la arquitectura Lenet-5, trabajada en clase y observar sus resultados, posteriormente aplicar una red Densa en el caso de MNIST y otra red convolucional propuesta en el caso de CIFART-10, y comparar los resultados.

- Lenet - 5: Es uno de los modelos pre-entrenados más sencillos y más utilizado para introducirse en el mundo de las arquitecturas de redes neuronales convolucionales. Utiliza el algoritmo de backpropagation y consta de 5 capas, de ahí viene su nombre.

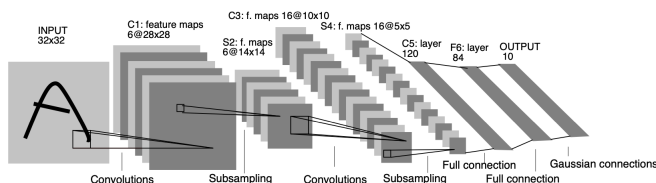


Figura 1. Arquitectura Lenet-5

El presente documento corresponde a un reporte sobre una actividad solicitada en la clase de Deep learning.

1. MNIST

Los datos de MNIST cuenta con 70,000 imagenes de 28x28, en escalas de grises que contienen números dibujados a mano. Una vez que se llaman estás imágenes se asignan 60,000 al conjunto de entrenamiento y 10,000 al de prueba.

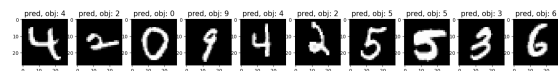


Figura 2. Muestras del dataset MNIST

El objetivo es entrenar una red neuronal convolucional dado el conjunto conjunto de datos guardado en la variable x-train de esta base de datos para lograr predecir el número que está escrito en la imagen. Es esta base de datos se encuentran 10 clases, que son los número del 0 al 9.

Antes de poder entrenar nuestra red, se tienen que pre-procesar los datos. En este caso, se cambió el tamaño de las imagenes de entrada de 28x28 a

El primer método que se aplicó es el de una red densa de 3 capas en total, una de entrada, una intermedia y una de salida. Para las dos primeras de indicó sigmoide como función de activación y para categorizar en la capa de salida se tiene softmax.

```
model = Sequential()
model.add(Dense(64, input_shape=(784,), activation='sigmoid'))
model.add(Dense(64, activation='sigmoid'))
model.add(Dense(10, activation='softmax'))
model.summary()
```

Figura 3. Modelo de red neuronal densa

Para este modelo se hizo un pre-procesamiento al utilizar one hot como método de categorización aplicado a las etiquetas de entrenamiento, de pruebas y de validación. Las métricas en ambos modelos aplicados a este dataset son Adan como optimizador con un learnign rate de 0.001, mean squared error como función de costo y accuracy para medir el rendimiento.

El segundo modelo que se podrá encontrar en el notebook para el dataset MNIST es Lenet-5, un modelo muy conocido y de bastante utilidad para este dataset en específico. Se mostrará la comparación de los resultados para analizar del funcionamiento de estos modelos mencionados en la sección de Resultados.

Model: "sequential_9"

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 26, 26, 6)	60
max_pooling2d_6 (MaxPooling2D)	(None, 13, 13, 6)	0
conv2d_7 (Conv2D)	(None, 11, 11, 16)	880
max_pooling2d_7 (MaxPooling2D)	(None, 5, 5, 16)	0
flatten_3 (Flatten)	(None, 400)	0
dense_20 (Dense)	(None, 120)	48120
dense_21 (Dense)	(None, 84)	10164
dense_22 (Dense)	(None, 10)	850

=====
 Total params: 60074 (234.66 KB)
 Trainable params: 60074 (234.66 KB)
 Non-trainable params: 0 (0.00 Byte)

Figura 4. Modelo Lenet-5

2. CIFART-10

Es una base de datos que contiene 60,000 imágenes con sus etiquetas correspondientes de las cuales 50,000 son dedicadas para el entrenamiento y 10,000 para probar realizar predicciones después de entrenar el modelo y validar los resultados obtenidos.

Es esta base de datos se encuentran 10 clases, las imágenes pueden pertenecer a una de las siguientes clases: airplane, automobile, bird, cat, deer, dog, frog, horse, ship y truck.

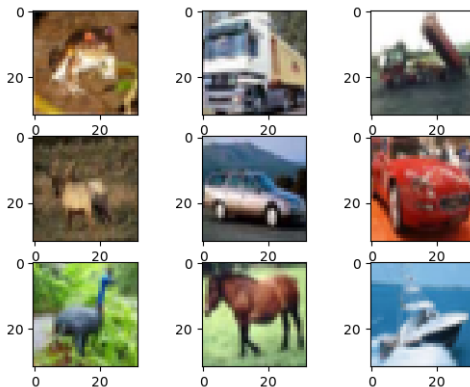


Figura 5. Muestras del dataset CIFART-10

El objetivo es entrenar una red neuronal convolucional con los datos de entrenamiento de esta red, guardados en la variable x-train, con el fin del predecir la clase a la que pertenece la imagen.

El primer modelo que se le aplicará a este dataset será una red convolucional propuesta.

Como ya se mencionó esta base de datos cuenta con 10 clases, por esta razón es que se requiere un pre-procesamiento de las etiquetas. Se utiliza el método to-categorical para normalizar el valor de las etiquetas a categorías one hot. Este

método se aplicó a las etiquetas de entrenamiento, de pruebas y de validación.

```
model.add(Conv2D(filters=6, kernel_size=(3,3), activation='sigmoid', input_shape=(32,32,3)))
model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Conv2D(filters=16, kernel_size=(4,4), activation='sigmoid'))
model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Flatten())

model.add(Dense(120, activation = "sigmoid"))
model.add(Dense(90, activation = "sigmoid"))

model.add(Dense(nClasses, activation = 'softmax'))
```

Figura 6. Modelo convolucional

Teniendo esto se crea el modelo convolucional propuesto para el dataset CIFART-10, el cual consta de 2 capas convolucionales y sus respectivas capas de Pooling. Las dos capas convolucionales tendrán 6 y 16 filtros respectivamente, el primero de 3x3 y el segundo de 4x4, ambas con función de activación sigmoide y únicamente la primera cuenta con el hiperparámetro input-shape de 32x32x3. Para las capas de pooling se aplica el método MaxPooling para obtener el valor máximo en un espacio de 2x2.

Al compilar el modelo se indican 3 parámetros más. Como función de costo se indica mean squared error, como optimizador se indica Adam con un learning rate de 0.001 y la métrica de evaluación será accuracy. Este modelo se ejecutará 150 épocas con un lote de 128 cada época.

Para el segundo modelo que se entrenará con este dataset, se configurará la arquitectura Lenet-5. Es decir un modelo de 5 capas más sus capas complementarias como la de Pooling y Flatten.

III. RESULTADOS

Se esperaban buenos resultados en los modelos aplicados al dataset MNIST debido a que este conjunto de imágenes no es tan sofisticado, empezando porque es en blanco y negro, es decir no tiene 3 canales a analizar (canales RGB) como las imágenes a color.

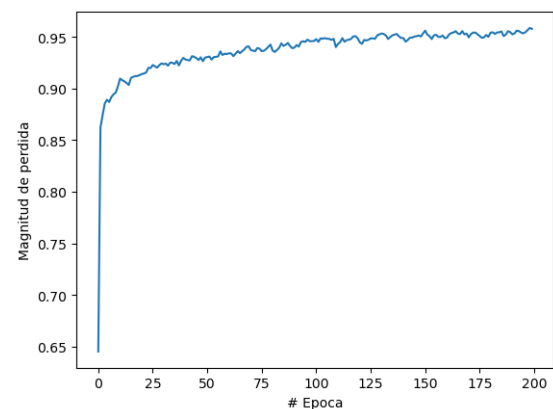


Figura 7. Accuracy obtenido en red densa - MNIST

Para la red densa se tuvo resultados alrededor del 95 por ciento de precisión en el entrenamiento y por tanto en los

resultados. Es un muy buen accuracy y teniendo en cuenta el dataset utilizado, puede ser un excelente resultado.

Aunque es una muy buen resultado, en comparación con el modelo Lenet-5, resulta ser más bajo ya que se demuestra que una red convolucional es más útil en el tratamiento de imágenes ya que en las múltiples pruebas que se le hicieron, los resultados fueron alrededor de 99 por ciento.

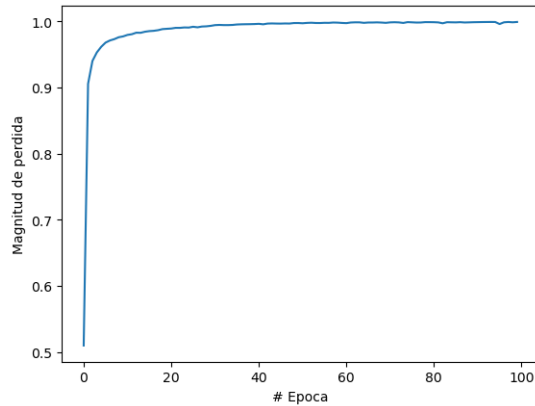


Figura 8. Accuracy obtenido con el modelo Lenet-5 - MNIST

El margen de error es muy bajo, y el entrenamiento fue muy rápido, prácticamente en las primeras 10 épocas el modelo aprendió todo. Esto puede confirmar que una red convolucional es mucho más efectivo que una red densa.

CIFART-10 es un conjunto de datos con más complejidad que MINST, ya que las imágenes tienen 3 canales(RGB), los elementos de cada clase tienen diferente ubicación en la imagen y colores, es decir, podemos encontrar automóviles rojos, blancos, etc.

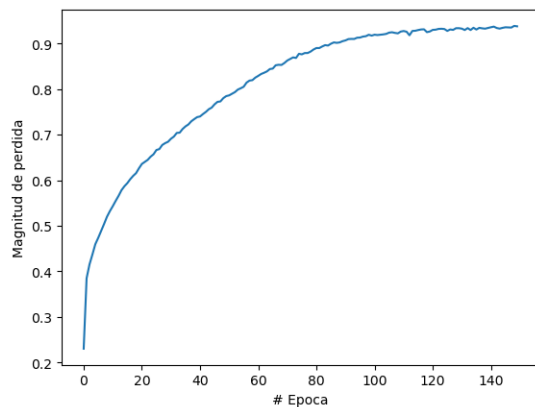


Figura 9. Accuracy obtenido con el modelo Lenet-5 - MNIST

Es por esto que se espera que los modelos puedan no ser tan efectivos como con un dataset en blanco y negro como MINST. Sin embargo, los resultados no fueron tan bajos como se esperaban, ya que con el modelo convolucional propuesto se llegó a obtener un 94 por ciento de accuracy.

Una arquitectura con capas convolucionas es ideal para el encontrar patrones o bordes dentro de un conjunto de imágenes y aunque se habló sobre el tamaño de las imágenes

de CIFART-10, el cual es 32x32x3, podría pensarse que el entrenamiento y predicciones podrían ser nada complicado.

Aplicando el modelo Lenet-5 se obtuvo un 87 por ciento. Un resultado un poco bajo para que un modelo sea confiable, sin embargo brinda información, una red convolucional es ideal para tratar esta base de datos, sin embargo es necesario mejorar al arquitectura para obtener mejores resultados.

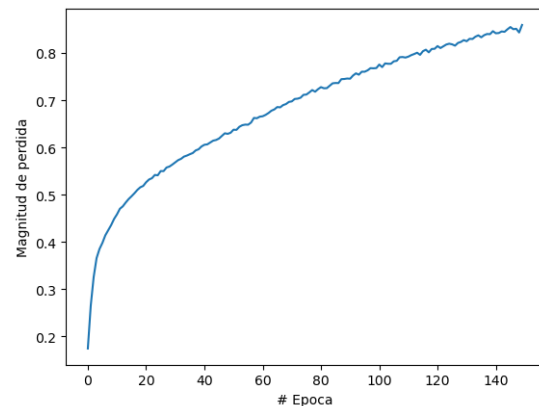


Figura 10. Accuracy obtenido con el modelo Lenet-5 - CIFART-10

A continuación se mostrarán los resultados obtenidos para cada red y modelo respectivamente, esto da una mejor visión de las métricas de cada modelo:

Dataset	Lenet-5	Densa	Propuesta
MNIST	99.99	95.07	-
CIFART-10	87.35	-	93.76

Tabla I

ACCURACY OBTENIDO EN CADA RED Y MODELO.

IV. CONCLUSIÓN

Existen 2 cosas que se pueden remarcar de este ejercicio, la primera es la forma de la gráfica de aprendizaje de la red, sin importar el modelo, la base de datos es bastante influyente en el aprendizaje de los modelos de redes neuronales.

Para el dataset de MINST se observa que el aprendizaje es bastante rápido al principio y estable en el resto del proceso. Para CIFART-10 parece ser un aprendizaje que incrementa poco a poco y podría aparentar que si se le aplican más épocas, podría aprender un poco más.