



Universidad Politécnica
de Madrid

Escuela Técnica Superior de
Ingenieros Informáticos



Máster Universitario en Inteligencia Artificial

Visión por Computador

Práctica 4: Clasificación de imágenes con Redes de Neuronas Profundas

Autores:

Alejandro Francisco Toral

Enrique Martín López

Madrid, 17 de enero de 2021

Índice

1. Introducción a los datasets	2
2. Proceso de diseño	2
2.1. Arquitecturas finales	2
3. Conclusiones	4

Índice de figuras

1. Gráfica con los valores de accuracy y loss obtenidos para Traffic Signs en ffNN	3
2. Gráfica con los valores de accuracy y loss obtenidos para CIFAR-10 en ffNN	3
3. Gráfica con los valores de accuracy y loss obtenidos para Traffic Signs en cNN	4
4. Gráfica con los valores de accuracy y loss obtenidos para CIFAR-10 en cNN	4

Índice de tablas

1. Tabla arquitecturas finales de ffNN	2
2. Tabla arquitecturas finales de ffNN	2
3. Tabla resultados finales de cNN	3
4. Tabla arquitecturas finales de cNN	4

1. Introducción a los datasets

En primer lugar tenemos un dataset de señales de tráfico que cuenta con 43 señales distintas y de cada una distintas imágenes con distintas características, como distinta luminosidad, perspectiva, etc...

Por el otro lado tenemos CIFAR-10 que es un dataset que cuenta con 10 clases distintas de imágenes ligeramente borrosas: aviones, automóviles, pájaros, gatos, ciervos, perros, ranas, caballos, barcos y camiones.

2. Proceso de diseño

El proceso de diseño que se ha seguido para la obtención de las arquitecturas finales que se describen en el apartado de arquitecturas finales ha sido ir valorando cuánto *bias* y cuánto *variance* había en las arquitecturas que diseñábamos y en función de eso ir tomando acciones al respecto para intentar minimizar ambas.

Por ejemplo, si veíamos que había una gran cantidad de *variance* intentábamos aplicar técnicas de regularización o si esto no era suficiente cambiar la arquitectura y bajar el número de neuronas y así evitar el *overfitting*. Si por el contrario el *bias* no era el deseado, tratábamos de cambiar la arquitectura metiendo más neuronas o capas, aplicar distintos optimizadores o cambiar el inicializador. A continuación se van a describir las arquitecturas finales obtenidas.

2.1. Arquitecturas finales

En primer lugar se van a describir las arquitecturas finales de fNN que se han obtenido, el rendimiento de dichas arquitecturas y las gráficas que representan el *accuracy* y el *loss* obtenidos. **Comentario:** En la columna *Structure*, se describe la estructura de las capas ocultas, ya que se da por hecho que el tamaño de la capa de entrada es el correspondiente al de la capa *Flatten* y la capa de salida es el número de clases que se consideran.

Notas:

La estructura notada como **A** corresponde a una arquitectura:
300, 150, 75

La estructura notada como **B** corresponde a una arquitectura:
2500, 1250, 675, 125, 50

Dataset	Structure	Batch Size	Activ.	Inicial.	Batch Norm	Regul.	Optimizer	LR	Epochs
Traffic	A	64	relu	He (Uniform)	Yes	Dropout (rate=0.01)	Adam	0.001	150
CIFAR-10	B	128	relu	He (Uniform)	Yes	Dropout (rate=0.05)	Adam	0.001	200

Tabla 1: Tabla arquitecturas finales de fNN

Los resultados obtenidos son:

Dataset	Train Acc.	Validation Acc.	Test Acc.
Traffic	94,5 %	78,2 %	81,1 %
CIFAR-10	100,0 %	81,0 %	83,9 %

Tabla 2: Tabla arquitecturas finales de fNN

Las gráficas obtenidas son las siguientes:

La gráfica para el dataset de Traffic Signs es el siguiente:

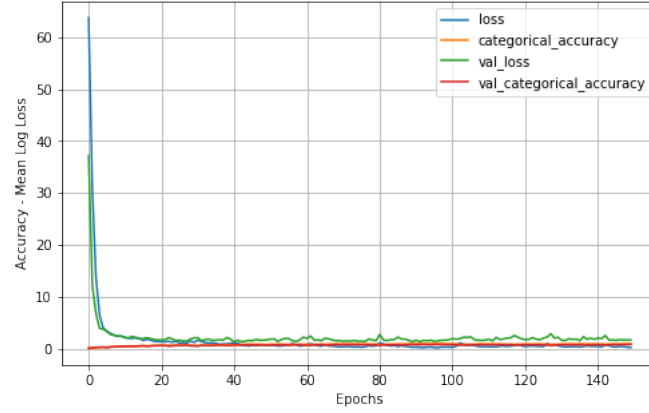


Figura 1: Gráfica con los valores de accuracy y loss obtenidos para Traffic Signs en fFNN

La gráfica para el dataset de CIFAR-10 es el siguiente:

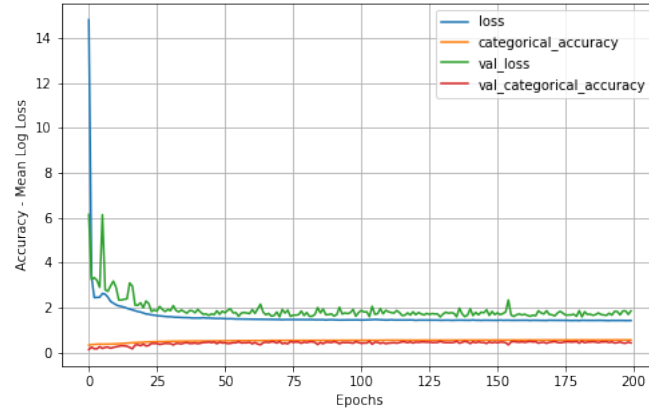


Figura 2: Gráfica con los valores de accuracy y loss obtenidos para CIFAR-10 en fFNN

Por último se van a describir las arquitecturas finales de cNN que se han obtenido y al igual que antes, el rendimiento y gráficas correspondientes.

Notas:

La estructura notada como **A** corresponde a una arquitectura:

(224, 224, 48), (112, 112, 96), (56, 56, 192), (28, 28, 32), (25, 25, 43), 43

La estructura notada como **B** corresponde a una arquitectura:

(32, 32, 32), (16, 16, 64), (8, 8, 128), (4, 4, 32), (1, 1, 10), 10

Dataset	Structure	Batch Size	Activ.	Inicial.	Batch Norm	Regul.	Optimizer	LR	Epochs
Traffic	A	64	relu	He (Uniform)	Yes	Dropout (rate=0.1) and L2(0.0001)	Adam	0.001	150
CIFAR-10	B	64	relu	He (Uniform)	Yes	Dropout (rate=0.2) and L2(0.0001)	Adam	0.001	200

Tabla 3: Tabla resultados finales de cNN

Los resultados obtenidos son:

Dataset	Train Acc.	Validation Acc.	Test Acc.	Training Time
Traffic	55,8 %	42,9 %	42,8 %	918 seconds
CIFAR-10	90,4 %	80,1 %	80,1 %	706 seconds

Tabla 4: Tabla arquitecturas finales de cNN

La gráfica para el dataset de Traffic Signs es el siguiente:

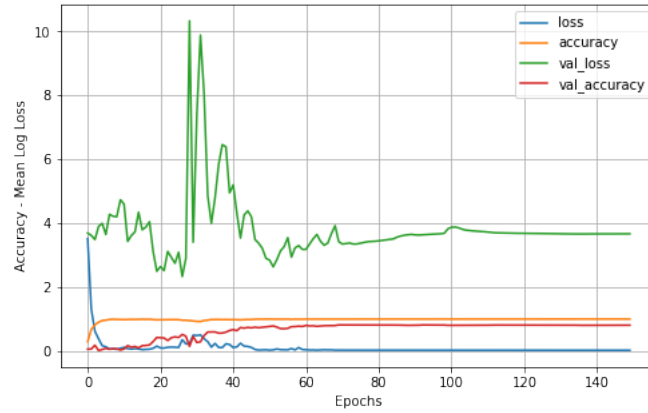


Figura 3: Gráfica con los valores de accuracy y loss obtenidos para Traffic Signs en cNN

La gráfica para el dataset de CIFAR-10 es el siguiente:

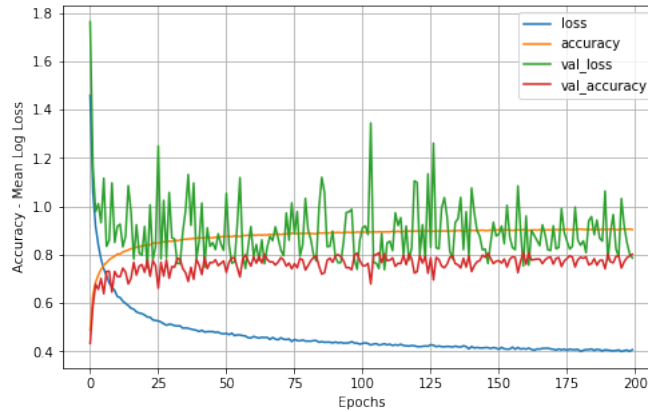


Figura 4: Gráfica con los valores de accuracy y loss obtenidos para CIFAR-10 en cNN

3. Conclusiones

Se ha podido comprobar como utilizar cNN para la clasificación de imágenes permite obtener un considerable mayor porcentaje de acierto a la vez que un menor tiempo de entrenamiento frente a las ffNN. En el dataset de CIFAR-10 se han notado especialmente estas ventajas, ya que los resultados obtenidos con ffNN eran especialmente malos. También se ha podido comprobar como aplicar técnicas de regularización ayuda en gran medida a bajar en bastante cantidad el *overfitting* que había en un principio sin aplicar la regularización.