



UNIVERSITAT  
POLITÀCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

# Redes neuronales

**Grau en Enginyeria Informàtica**  
**Aprendizaje automático**

Iñaki Díez Lambies

Andrea Gascó Pau

Grupo: L1-4CO21

## Contenido

1. Introducción.....	3
2. Experimentos .....	<b>¡Error! Marcador no definido.</b>
3. Evaluación.....	4

## 1. Introducción

En esta práctica se nos propone la tarea de entrenar un clasificador basado en redes neuronales para minimizar el error de clasificación.

Para esta tarea, se nos ha provisto del conjunto de datos de *FashionMNIST*.

## 2. Preparación

Para la realización de este experimento hemos realizado un script para poder probar los diferentes valores.

Los parámetros que hemos ido variando en los diferentes experimentos han sido:

El número de neuronas, que debe de ser un valor lo suficientemente grande como para que el modelo se ajuste de forma correcta, sin caer en un número elevado que nos podría llevar a un sobreajuste del modelo.

El número de épocas, que se refiere al número de veces que el conjunto de datos de entrenamiento pasa a través de la red neuronal durante el proceso de entrenamiento. Este valor se determina principalmente por la precisión y la capacidad del modelo para generalizar los datos. En general, un mayor número de épocas puede llevar a un modelo más preciso, ya que los pesos del modelo se actualizan más. Sin embargo, puede llevar a un sobreajuste de los datos de entrenamiento.

El factor de aprendizaje, que sirve para controlar la magnitud de las actualizaciones de los pesos del modelo. Este se aplica a los gradientes calculados durante el entrenamiento y se utiliza para actualizar los pesos del modelo. En general, cuanto más complejo es el modelo o mayor cantidad de datos tenemos, el factor de aprendizaje será pequeño, para evitar convergencias hacia una solución o sobreajuste del modelo.

El número de capas, que viene determinado por la complejidad del conjunto de datos. Al igual que el número de neuronas, no hay que caer en un número innecesario de capas, pues llevaría a un sobreajuste de los datos de entrenamiento.

El tamaño de batch, que es el número de entrenamientos que se utilizan en cada iteración (época) durante el entrenamiento. Depende de factores, como la complejidad del modelo o la cantidad de datos de entrenamiento. Este valor, influye en la precisión del modelo y en la velocidad del entrenamiento. En general, un valor más elevado puede llevar a un entrenamiento más rápido, ya que se realizan menos iteraciones para completar una época. Sin embargo, un tamaño de batch grande puede llevar a que el modelo sea menos preciso, ya que la actualización de pesos se basa en más datos. En contra posición, un tamaño de batch pequeño, puede llevar a un entrenamiento más lento, pero más preciso.

A raíz de esto, hemos analizado la tasa de acierto y de pérdida del modelo. El primero, es la proporción de las predicciones correctas del modelo en relación con el total de predicciones realizadas. Se usa como medida del rendimiento. La segunda, se calcula como la diferencia entre las predicciones del modelo y los valores reales para un conjunto de datos de entrenamiento o validación. Es objetivo es la minimización de este valor.

### 3. Experimentos

Para encontrar la mejor tasa de acierto hemos realizado dos experimentos. Hay que tener en cuenta que cada vez que se realizan los experimentos, los resultados que se obtienen pueden diferir un poco a los resultados obtenidos con los mismos datos en otra ejecución de este.

En este primer experimento hemos analizado los siguientes valores: batch de (32, 64, 128), número de capas de (2, 4, 8), número de neuronas por capa de (128, 256, 512, 1024), factor de aprendizaje de (0.1, 0.01, 0.001) y 12 épocas.

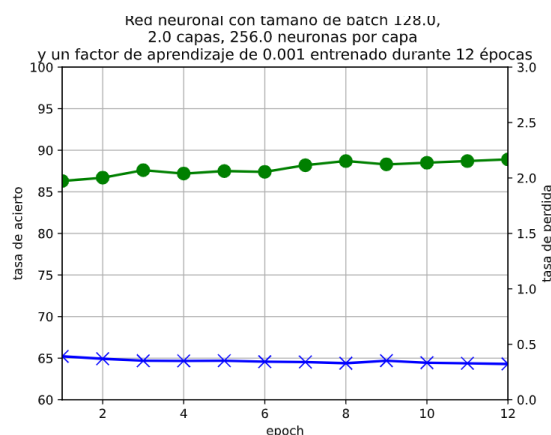
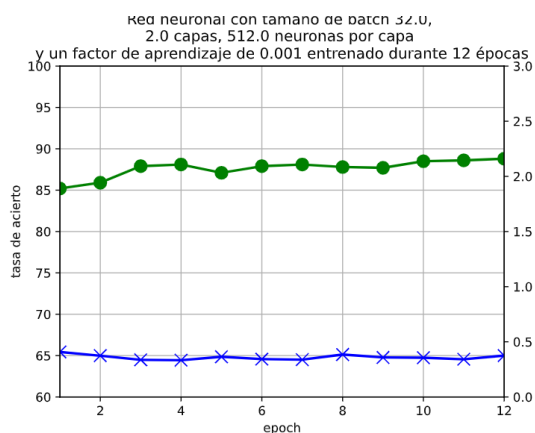
A raíz del primer experimento se han extraído las siguientes conclusiones a través del análisis, comprensión y comparación de los diferentes resultados. En primer lugar, se puede observar que cuanto mayor es el número de épocas, mayor es la tasa de acierto hasta llegar a un valor cercano al 10, 11 o 12 donde ya empieza a descender.

Por otro lado, un factor de aprendizaje elevado, como 0.1, no da una tasa de acierto elevada, siendo mejor utilizar un factor de aprendizaje más bajo, como 0.001. Si utilizamos un valor inferior, como 0.0001, también nos encontramos, por lo general, peores resultados. Además, se ha observado que el tamaño de batch no afecta significativamente en los resultados obtenidos.

En último lugar, se puede observar que los mejores resultados se encuentran con una proporción de capas y número de neuronas similares. Ejemplos de esta proporción son:

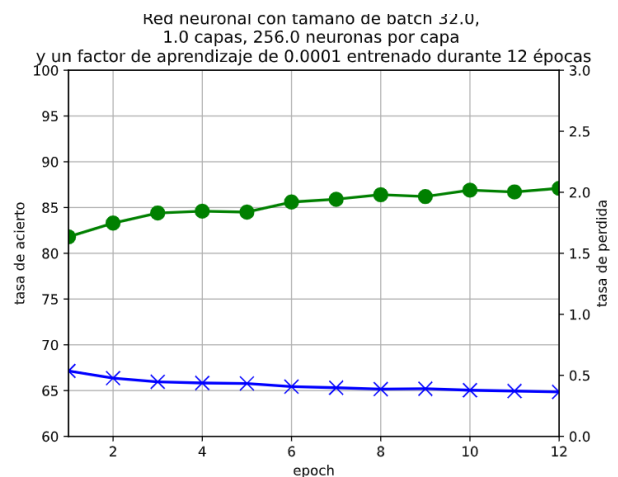
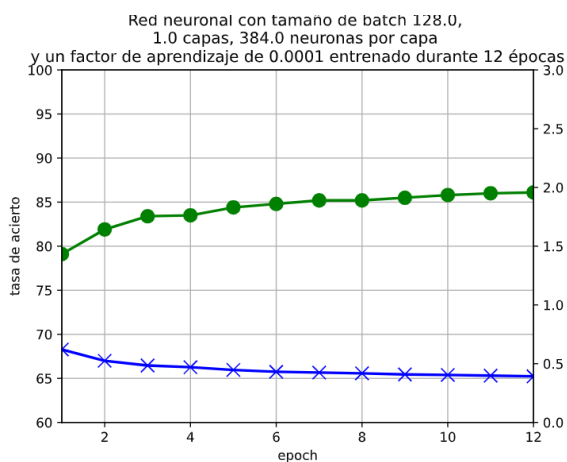
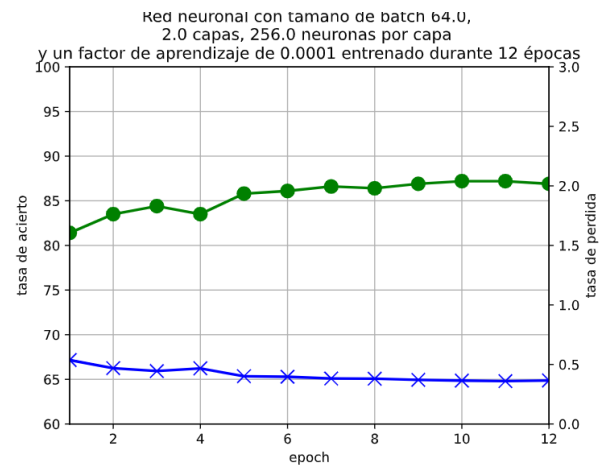
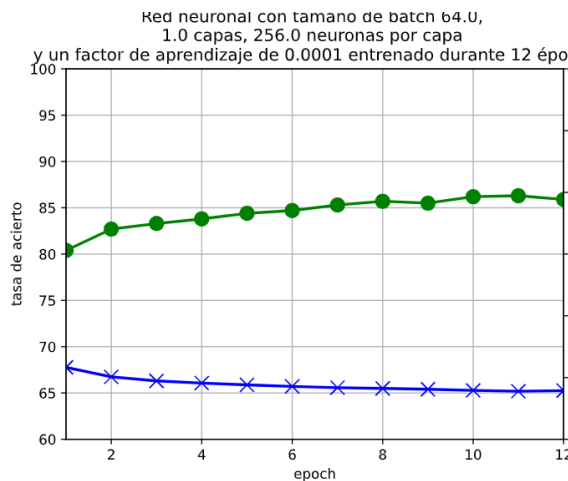
- 2 capas con 256 neuronas por capa
- 1 capa con 512 neuronas por capa
- 2 capas con 386 neuronas por capa

Teniendo en cuenta todo esto, en este experimento se ha obtenido un mejor acierto con un batch de 32, número de capas 2, número de neuronas 512, un factor de aprendizaje de 0.001 y número de épocas 11 (32 2 512 0.001 11) y una mejor tasa de pérdida con un batch de 128, número de capas 2, número de neuronas 256, un factor de aprendizaje de 0.001 y número de épocas 12 (128 2 256 0.001 12). A continuación, se pueden observar las gráficas de estas dos donde se ve su evolución por cada época.



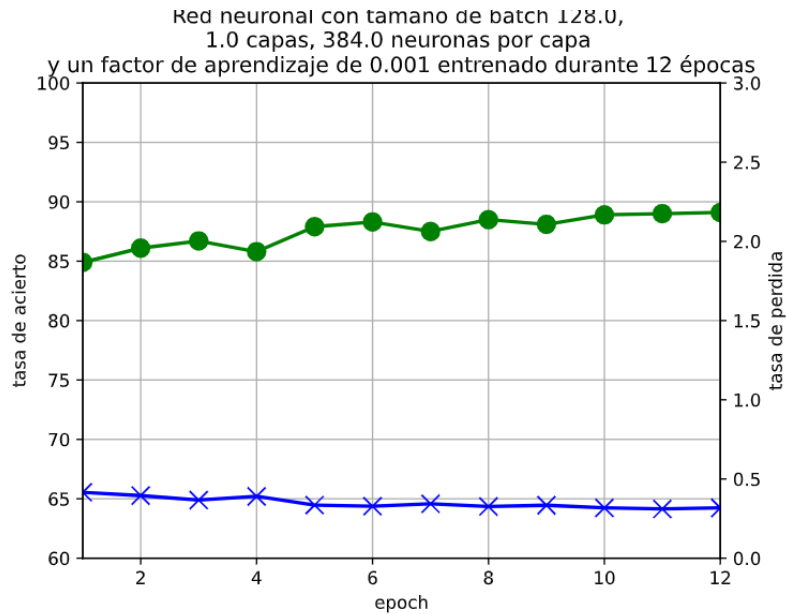
En el segundo experimento hemos analizado los siguientes valores: batch de (32, 64, 128), número de capas de (1, 2, 3), número de neuronas por capa de (256, 384, 512), factor de aprendizaje de (0.001, 0.001) y 12 épocas.

A raíz de este hemos conseguido diversos valores que han empatado con una tasa de acierto de 84.4. Aquí se pueden apreciar las distintas gráficas que muestran los resultados.



Como podemos observar, todas estas redes neuronales tienen en común un factor de aprendizaje de 0.0001. Es decir, tienen un entrenamiento más lento que el anteriormente testado pero que acaba por obtener una mayor precisión en sus clasificaciones.

En cuanto al factor de pérdida del modelo, podemos encontrar que el mejor valor lo ostenta aquel con un valor de batch de 128, 1 capa, 384 neuronas por capa y un factor de aprendizaje de 0.001 en su época 11. Además, este modelo presenta una tasa de acierto para nada despreciable de un 89.0 sin caer en tasas de pérdida de los modelos anteriores como 0.41 o incluso 0.43. Aquí a continuación podemos observar su representación gráfica.



#### 4. Conclusiones

Los resultados obtenidos son muy satisfactorios en cuanto a las tasas de acierto obtenidas en los segundos experimentos. Cualquiera de los resultados obtenidos sería válido pero hemos decidido que el mejor sería aquel con un tamaño de batch de 128, 1 capa, 384 neuronas por capa y un factor de aprendizaje de 0.001 entrenado durante 12 épocas.

Nos hemos decidido por este debido a que las predicciones del modelo están más cercanas a los valores reales para el conjunto de datos de entrenamiento. Esto sumado a que apenas perdemos tasa de acierto (89.0 frente a 89.4) nos encontramos con un modelo un poco más completo.