



Universidad
Rey Juan Carlos

Práctica 4. Redes Neuronales

Aprendizaje Automático

**Grado en Ingeniería de Robótica
Software**

Curso 2023/2024

Ejercicio 1. Regresión.

Se dispone de un conjunto de datos almacenados en el archivo “housing.csv”. Este conjunto de datos consta de variables numéricas con características de casas del estado de California (variables de la 1 a la 8) y una variable con el precio de cada una de las casas en cientos de miles de dólares (variable 9).

Resuelva los siguientes apartados:

1.1. Primero, divida el conjunto de datos en dos partes: conjunto de datos de entrenamiento (80 %) y conjunto de datos de test (20 %). Con esto, genere un modelo de una red neuronal con una capa oculta entrenada con el conjunto de datos de entrenamiento. Para ello, utilice la función de activación ReLU en la capa oculta, un tamaño de *batch* de 5, 100 *epochs*, un optimizador *Adam* y una función de pérdidas de error cuadrático medio (*mean squared error*). Ten en cuenta que, para este caso, no hay función de activación en la capa de salida. ¿Cuál es el error cuadrático medio del conjunto de datos de entrenamiento? ¿Cuál es el error cuadrático medio del conjunto de datos de test? ¿Se está produciendo sobreajuste (*overfitting*)?

1.2. Genere varios modelos de redes neuronales profundas (con más de una capa oculta) entrenadas con el conjunto de datos de entrenamiento. ¿Cuál es el error cuadrático medio del conjunto de datos de entrenamiento y de test para cada una de ellas? ¿Qué sucede al aumentar la profundidad de las redes neuronales?

Ejercicio 2. Clasificación binaria.

Se dispone de un conjunto de datos almacenados en el archivo “sonar.csv”. Este conjunto de datos consta de variables numéricas de medidas de sonar de elementos (variables de la 1 a la 60) y una variable con el tipo de elemento que es: “M” para elementos metálico y “R” para elementos de piedra (variable 61).

Resuelva los siguientes apartados:

2.1. Primero, divida el conjunto de datos en dos partes: conjunto de datos de entrenamiento (80 %) y conjunto de datos de test (20 %). Con esto, genere un modelo de una red neuronal con una capa oculta entrenada con el conjunto de datos de entrenamiento. Para ello, utilice la función de activación ReLU en la capa oculta, un tamaño de *batch* de 10, 250 *epochs*, un optimizador *Adam* y una función de pérdidas de entropía cruzada binaria (*binary cross entropy*). Ten en cuenta que, para este caso, la función de activación de la capa de salida es una sigmoide. ¿Cuál es la precisión (*accuracy*) del conjunto de datos de entrenamiento? ¿Cuál es la precisión (*accuracy*) del conjunto de datos de test? ¿Se está produciendo sobreajuste (*overfitting*)?

2.2. Genere varios modelos de redes neuronales profundas (con más de una capa oculta) entrenadas con el conjunto de datos de entrenamiento. ¿Cuál es la precisión (*accuracy*) del conjunto de datos de entrenamiento y de test para cada una de ellas? ¿Qué sucede al aumentar la profundidad de las redes neuronales?

Ejercicio 3. Clasificación multiclase.

Se dispone de un conjunto de datos almacenados en el archivo “iris.csv”. Este conjunto de datos consta de variables numéricas de medidas varias partes de flores de la especie iris (variables de la 1 a la 4) y una variable con el tipo de especie de flor iris que es: “Iris-setosa”, “Iris-versicolor” e “Iris-virginica” (variable 5).

Resuelva los siguientes apartados:

3.1. Primero, divida el conjunto de datos en dos partes: conjunto de datos de entrenamiento (80 %) y conjunto de datos de test (20 %). Con esto, genere un modelo de una red neuronal con una capa oculta entrenada con el conjunto de datos de entrenamiento. Para ello, utilice la función de activación ReLU en la capa oculta, un tamaño de batch de 5, 200 *epochs*, un optimizador *Adam* y una función de pérdidas de entropía cruzada (*cross entropy*). Ten en cuenta que, para este caso, la función de activación de la capa de salida debería ser una *softmax*. Sin embargo, la función de pérdidas “CrossEntropyLoss” de Pytorch ya tiene en cuenta esa función de activación internamente. ¿Cuál es la precisión (*accuracy*) del conjunto de datos de entrenamiento? ¿Cuál es la precisión (*accuracy*) del conjunto de datos de test? ¿Se está produciendo sobreajuste (*overfitting*)?

3.2. Genere varios modelos de redes neuronales profundas (con más de una capa oculta) entrenadas con el conjunto de datos de entrenamiento. ¿Cuál es la precisión (*accuracy*) del conjunto de datos de entrenamiento y de test para cada una de ellas? ¿Qué sucede al aumentar la profundidad de las redes neuronales?