**Respuestas**

**Ejercicio # 4**

¿Qué conjunto de datos se utiliza en este código y qué problema de aprendizaje automático se aborda?

Es el conjunto de datos de las viviendas de California, aborda el entrenamiento el modelo de aprendizaje automático con los datos de entrenamiento y validación; asimismo las métricas ayudan a visualizar la perdida y el error a lo largo del modelo

¿Qué hace la función train\_test\_split() de scikit-learn y por qué se utiliza en este código?

Copitot

La función `train\_test\_split()` de la biblioteca scikit-learn es esencial para la validación de modelos en el aprendizaje automático. Permite dividir datos en conjuntos de entrenamiento y prueba, lo cual es crucial para evaluar el rendimiento de un modelo con datos no vistos. Los parámetros más comunes incluyen `test\_size` que define la proporción del conjunto de datos a utilizar como conjunto de prueba, `train\_size` que especifica la proporción del conjunto de datos a utilizar como conjunto de entrenamiento, `random\_state` que es la semilla para el generador de números aleatorios, asegurando la reproducibilidad de los resultados, y `shuffle` que decide si se mezclan los datos antes de dividirlos.

¿Por qué es importante estandarizar los datos antes de entrenar un modelo de redes neuronales?

Mejorar la eficiencia del entrenamiento, evitar el sesgo de características, mejora en la estabilidad numérica, facilita el aprendizaje de característica.

¿Qué arquitectura de red neuronal se utiliza en este código y cuántas capas tiene?

Tiene 2 capas de red neuronal y es una red de feedforward simple con una capa oculta

¿Qué función de activación se utiliza en la capa oculta de la red neuronal y por qué se elige esa función?

ReLU

¿Qué función de pérdida se utiliza para compilar el modelo y qué métricas se utilizan para evaluar su rendimiento?

La función de perdida es regresión de cuadrático medio (MSE), y las métricas que se utilizan son rendimiento de rendimiento

¿Cuántas épocas se utilizan para entrenar el modelo y por qué se elige ese número?

10 epocas.

**Ejercicio # 5**

¿Qué conjunto de datos se utiliza en este código y qué problema de aprendizaje automático se aborda?

El problema de aprendizaje automático que se aborda es la clasificación de imágenes. El objetivo es entrenar un modelo para que pueda reconocer y clasificar correctamente los dígitos del 0 al 9 en las imágenes.

¿Por qué es importante normalizar los valores de píxeles de las imágenes antes de entrenar el modelo?

Para tener un rango consistente, convergencia mas rápida, evita el desbordamiento numerico

¿Qué arquitectura de red neuronal se utiliza en este código y cuántas capas tiene?

Un modelo Sequential y tiene 3 capas

¿Cuál es la función de activación utilizada en la capa oculta y en la capa de salida de la red neuronal, y por qué se eligen esas funciones?

¿Qué función de pérdida se utiliza para compilar el modelo y qué métricas se utilizan para evaluar su rendimiento?

El método model.evaluate() devuelve la pérdida y las métricas de evaluación calculadas sobre los datos proporcionados

¿Cuántas épocas se utilizan para entrenar el modelo y por qué se elige ese número?

10 epocas; entre mas complejo el modelo requiere mas epocas

¿Qué significa la función model.summary(), y que información proporciona?

La función model.summary() es una función útil en bibliotecas de aprendizaje profundo como Keras y TensorFlow. Proporciona un resumen conciso del modelo que has construido.

Capas del modelo: Muestra todas las capas del modelo en orden, junto con los nombres de las capas. Forma de salida de cada capa: Para cada capa, muestra la forma de salida. Esto es útil para entender cómo las dimensiones de tus datos cambian a medida que pasan por el modelo. Número de parámetros en cada capa: Muestra el número de parámetros (pesos y sesgos) que se aprenden en cada capa. Número total de parámetros: Al final del resumen, muestra el número total de parámetros en el modelo. Número de parámetros entrenables y no entrenables: También muestra cuántos de los parámetros son entrenables y cuántos no lo son. Un parámetro no entrenable es uno que no se actualiza durante el entrenamiento. Por ejemplo, en una capa de normalización por lotes, los parámetros de media y varianza se consideran no entrenables ya que se derivan de los datos.