

Título de la tarea: Monitorización del entrenamiento de redes neuronales profundas

Ciclo formativo y módulo: Curso especialización en Inteligencia Artificial y Big Data
- Programación de Inteligencia Artificial

¿Qué contenidos o resultados de aprendizaje trabajaremos?

Resultados de aprendizaje

- ✓ **RA1.** Caracteriza lenguajes de programación valorando su idoneidad en el desarrollo de inteligencia artificial.
- ✓ **RA2.** Desarrolla aplicaciones de Inteligencia artificial utilizando entornos de modelado.

Contenidos

- 1.- Evaluación de un modelo de machine learning
- 2.- Sobre entrenamiento u "overfitting" .
 - 2.1.- Evitar el sobre entrenamiento.
 - 2.2.- Decaimiento de los pesos.
 - 2.3.- Descarte o "dropout".
- 3.- TensorBoard.
- 4.- Técnicas avanzadas de mejora en aprendizaje automático.
 - 4.1.- Normalización por lotes.
 - 4.2.- Optimización de hiperparámetros.

Siguiente »

1.- Descripción de la tarea.




Caso práctico

Lorena está en pleno proceso de ajuste y mejora de su modelo de deep learning y ha descubierto lo útil que es el kit de monitorización TensorBoard que se llama desde el entrenamiento con el método fit de Keras y que permite visualizar de forma gráfica el comportamiento del modelo. Le ha sorprendido lo sencillo que es cargar la extensión en el notebook en el que está trabajando, y cómo basta con incluir, en fit, el parámetro de callbacks `tensorboard_callback`.



[LookStudio](#) (Licencia de Freepik)

¿Qué te pedimos que hagas?

- ✓ **Apartado 1: Inicializa un proyecto de regresión con TensorBoard.**
 - Inicia un nuevo notebook, preferiblemente en Google Colab. Para guiarte en el proceso, puedes utilizar este [cuaderno-guía](#)  con algunas sugerencias de fragmentos de código indicados en las celdas de texto, pero tendrás que escribir el código en la celda de código correspondiente y ejecutarlo.
 - Importa la librerías y paquetes necesarios
 - Carga la extensión de TensorBoard.
 - Crea un dataset de 1000 casos de variables x e y basados en añadir ruido a puntos aleatorios de la recta de ecuación $y = 0,4x + 1$
 - Reservar el 80% de datos para el entrenamiento y el 20% para validación o test.
- ✓ **Apartado 2: Crea una red neuronal profunda de dos capas con Loss MSE y optimizador SGD.**
 - Crea un modelo basado en una red neuronal de dos capas.
 - Configura el entrenamiento utilizando la función de coste del error medio cuadrático (MSE).
 - Configura el optimizador SGD
 - En el método fit, incluye el parámetro `callbacks = (tensorboard_callback)`
- ✓ **Apartado 3: Lanza TensorBoard.**
 - Utiliza la función mágica que lanza TensorBoard: `%tensorboard --logdir logs/scalars`.
 - Explora las representaciones de las métricas
 - ¿A qué conclusiones llegas?
- ✓ **Apartado 4: Crea el modelo predictivo.**
 - Genera el modelo predictivo con el método `predict()`.
 - Predice los valores para $x = 60$, $x = 25$ y $x = 2$.
 - Compara los valores predichos con los calculados de forma exacta con la ecuación de la recta $y = 0,4x + 1$ y analiza los resultados.

NOTA IMPORTANTE

Es necesario entregar el notebook con los resultados ya ejecutados, y es conveniente entregar una presentación de apoyo con las capturas de pantalla de los principales pasos. En dicha presentación, incluir enlace abierto al notebook con el código para que cualquier persona con el enlace pueda acceder a él.

[« Anterior](#) [Siguiente »](#)

2.- Información de interés.



Recursos necesarios y recomendaciones

Recursos necesarios

- ✓ Ordenador personal con, al menos, 4 Gigabytes de memoria RAM
- ✓ Conexión a Internet.
- ✓ Navegador web.
- ✓ [Cuaderno-guía con anotaciones de código sugeridas](#) .

Recomendaciones

- ✓ Antes de abordar la tarea:
 - ➔ lee con detenimiento la unidad, consulta los enlaces para saber más, examina el material proporcionado por el profesor y aclara las dudas que te surjan con él.
 - ➔ Realiza el examen online de la unidad, y consulta nuevamente las dudas que te surjan. Solo cuando lo tengas todo claro, debes abordar la realización de la tarea.
- ✓ No olvides elaborar el documento explicativo y adjuntar el archivo del notebook que se haya generado.



Indicaciones de entrega

Una vez realizada la tarea, el envío se realizará a través de la plataforma. El archivo se nombrará siguiendo las siguientes pautas:

Apellido1_Apellido2_Nombre_PIA06_Tarea

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna **Begoña Sánchez Mañas para la sexta unidad del MP de PIA**, debería nombrar esta tarea como...

sanchez_manas_begona_PIA06_Tarea

« Anterior Siguiente »

3.- Evaluación de la tarea.

Criterios de evaluación implicados

Criterios de evaluación RA1

- ✓ a. Se ha identificado la estructura de un programa informático.
- ✓ b. Se han valorado características en los lenguajes de programación adecuadas al tipo de aplicaciones a implementar.
- ✓ d. Se han valorado características de los lenguajes de programación para el desarrollo de Inteligencia Artificial.

Criterios de evaluación RA2

- ✓ b. Se han caracterizado entornos de modelo de aplicaciones de inteligencia artificial.
- ✓ c. Se ha definido el modelo que se quiere implementar según el problema planteado.
- ✓ d. Se ha implementado la aplicación de Inteligencia Artificial.
- ✓ e. Se han evaluado los resultados obtenidos.

¿Cómo valoramos y puntuamos tu tarea?

Rúbrica de la tarea	
Apartado 1: Se inicializa un proyecto de regresión con TensorBoard.	2,5 puntos
Apartado 2: Se crea una red neuronal profunda de dos capas con Loss MSE y optimizador SGD.	2,5 puntos
Apartado 3: Se lanza TensorBoard.	2,5 puntos
Apartado 4: Se crea el modelo predictivo.	2,5 puntos
Redacción clara y correcta, sin errores ortográficos	Se restan 0,1 puntos por cada error ortográfico o expresiones incorrectas.

NOTA IMPORTANTE

Aquellos apartados/subapartados en los que las capturas de pantalla no sean claras o no tengan como fondo de pantalla la plataforma con tu usuario mostrando claramente la foto de tu perfil, no serán corregidos.