

Hoja de trabajo #1

Material de Apoyo

1. Ciclo For en Python

for i in range(inicio, fin, paso):

Código a ejecutar en cada iteración

2. Obtener la precisión y la pérdida de una red neuronal con Keras

Tras entrenar un modelo con `.fit()`, este retorna un objeto `history`. Puedes acceder a la precisión y la pérdida usando:

history = modelo.fit(...)

precisión = history.history['accuracy']

3. Incrementos Logarítmicos

Usar `np.logspace` para generar números en una escala logarítmica. Mayor información en [este enlace](#).

array = np.logspace(np.log10(inicio), np.log10(fin), num=cantidad_de_pasos)

4. Visualización de Métricas

Se recomienda el uso de bibliotecas como `matplotlib` o `seaborn` para graficar y analizar resultados. Por ejemplo, para graficar la precisión y pérdida durante el entrenamiento:

import matplotlib.pyplot as plt

plt.plot(history.history['accuracy'])

```
plt.plot(history.history['loss'])  
plt.title('Model Accuracy & Loss')  
plt.ylabel('Value')  
plt.xlabel('Epoch')  
plt.legend(['Accuracy', 'Loss'], loc='upper left')  
plt.show()
```

5. Documentación y Recursos

Recuerda consultar la documentación oficial de [Keras](#) y [TensorFlow](#) para obtener más detalles sobre funciones y métodos específicos.