

Evaluación final

Para resolver los problemas siguientes, debes acceder a las bases de datos que se encuentran en el siguiente repositorio:

https://drive.google.com/drive/folders/1q_1GiS_MXf60s9agziMkjamX1MFINMDj?usp=sharing

- 1) Empleando una red neuronal densamente conectada, debes resolver el problema de predecir el precio de vehículos. La base de datos correspondiente se denomina “**car data.csv**”, y fue tomada originalmente del sitio Kaggle (<https://www.kaggle.com/nehalbirla/vehicle-dataset-from-cardekho>).

Esta base de datos está constituida por la información de 301 vehículos, donde la información de cada vehículo está representada por 9 atributos, los cuales se enlistan a continuación:

1. Car_Name
2. Year
3. Selling_Price
4. Present_Price
5. Kms_Driven
6. Fuel_Type
7. Seller_Type
8. Transmission
9. Owner

De estos 9 atributos, el número 3, **Selling Price**, representa la etiqueta de cada una de las instancias.

Tu implementación debe cumplir con los siguientes requisitos:

- Elimina la columna **Car_Name**, ya que sus elementos no son significativos para realizar la predicción del precio del vehículo.
- Sustituye los valores de la columna **Year**, por el número de años que han pasado luego de la construcción del vehículo. Por ejemplo, si el valor del atributo **Year** de algún vehículo es **2015**, tienes que reemplazar ese valor por **6**, ya que **2021-2015=6**.
- Los valores de los atributos Full_Type, Seller_Type y Transmission, están codificados como strings, por lo tanto, es necesario convertir dichos strings a su representación numérica.

Para lograrlo, utiliza la siguiente línea de código de la librería Pandas:
`df=pd.get_dummies(data=df,drop_first=True)`.

- Tomando en cuenta lo anterior, debes construir dos conjuntos de datos: entrenamiento y validation. El conjunto de entrenamiento debe estar constituido por 210 instancias, mientras que el conjunto de validación debe estar compuesto por 91 instancias.
- Como una guía, considera que las instancias de los conjuntos de entrenamiento y prueba, deben tener los siguientes atributos:
 1. Year
 2. Present_Price
 3. Kms_Driven
 4. Owner
 5. Fuel_Type_Diesel
 6. Fuel_Type_Petrol
 7. Seller_Type_Individual
 8. Transmission_Manual
- Utiliza la función **mae** (Mean Average Error) como métrica de evaluación del desempeño de tu modelo.
- Despliega las gráficas de tu modelo correspondientes a la función de pérdida y a la métrica de evaluación.
- Debes guardar el modelo que tenga el mejor desempeño en el conjunto de validación, para ello, emplea **callbacks**.
- Finalmente, asegúrate de que el rendimiento de tu MEJOR MODELO en el conjunto de validación con base a la métrica de evaluación **mae**, sea menor al 0.5. Si este rendimiento no se logra, no obtendrás los 30 puntos de este ejercicio.
- TODOS LOS REQUISITOS PREVIAMENTE DESCRITOS, DEBEN DE REALIZARSE MEDIANTE CÓDIGO, NADA PUEDE REALIZARSE DE FORMA MANUAL.

NOTA: En un mensaje privado de Slack, compárteme el notebook (s) que contenga la solución a este problema. También, tienes que compartirme el modelo que tenga el mejor desempeño en el conjunto de validación.

Valor: 30 puntos

- 2) Implementa una red neural convolucional que resuelva el problema de clasificación del clima. Para resolver este problema, tienes que emplear la base de datos contenida en el archivo **"Weather_Dataset.zip"**. Esta base de datos fue tomada del siguiente sitio web:

<https://data.mendeley.com/datasets/4drtfjtfy/1>

La base de datos en cuestión contiene 4 clases:

1. Cloudy
2. Rain
3. Shine
4. Runrise

Tu implementación debe cumplir con los siguientes requisitos:

- Construye dos conjuntos de datos: entrenamiento y validación. La única restricción que debes de cuidar en la generación de estos conjuntos, es que cada clase del conjunto de entrenamiento debe tener máximo 150 instancias. ES OBLIGATORIO QUE LOS CONJUNTOS DE ENTRENAMIENTO Y VALIDACIÓN LOS GENERES MEDIANTE CÓDIGO, NO PUEDES HACERLO DE FORMA MANUAL.
- Algunas de las imágenes de la base de datos están dañadas, por lo tanto, debes de diseñar una estrategia que te permita solventar este problema mediante código, NO PUEDES ELIMINAR LAS IMÁGENES DAÑADAS DE FORMA MANUAL.
- Debes desplegar las gráficas de precisión y pérdida.
- Debes guardar el modelo que tenga el mejor desempeño en el conjunto de validación, para ello, emplea **callbacks**.
- Además de utilizar la métrica acc para evaluar el desempeño de tu modelo en el conjunto de validación, debes desplegar la **matriz de confusión** que muestre el rendimiento de tu MEJOR MODELO en el conjunto de validación. También tienes que mostrar el valor de la métrica **F1** con respecto al rendimiento de tu MEJOR MODELO en el conjunto de validación.

Los requisitos anteriores deben de cumplirse para que tengas derecho a que tu implementación reciba una calificación. Considerando lo anterior, los valores que puede tener tu implementación son los siguientes:

- a) Si la precisión de validación alcanzada por tu MEJOR MODELO es menor a 0.80, obtendrás 0 puntos.

- b) Si la precisión de validación alcanzada por tu MEJOR MODELO corresponde a un valor entre 0.80 - 0.85, obtendrás 10 puntos.
- c) Si la precisión de validación alcanzada por tu MEJOR MODELO es mayor a 0.85 y menor o igual a 0.95, obtendrás 20 puntos.
- d) Si la precisión de validación alcanzada por tu MEJOR MODELO es mayor a 0.95, obtendrás 30 puntos.

NOTA: En un mensaje privado de Slack, compárteme el notebook (s) que contenga la solución a este problema. También, tienes que compartirme el modelo que tenga el mejor desempeño en el conjunto de validación.

Valor máximo: 30 puntos

- 3) Implementa una red neural convolucional que resuelva el problema de clasificación de construcciones arquitectónicas. Para resolver este problema, tienes que emplear la base de datos contenida en el archivo ***"Architectural_Heritage_Elements_Dataset_128(creative_commons).zip"***. Esta base de datos fue tomada del siguiente sitio web: <https://old.datahub.io/dataset/architectural-heritage-elements-image-dataset/resource/aed19d35-9e7a-4ec3-9234-187dffb5484f>

Es necesario señalar que la base de datos original fue modificada para los fines del presente examen.

La base de datos en cuestión contiene 7 clases:

1. altar
2. bell_tower
3. column
4. domn(outerior)
5. gargoyle
6. stained_glass
7. vault

Tu implementación debe cumplir con los siguientes requisitos:

- Construye tres conjuntos de datos: entrenamiento, validación y prueba. Para construir estos conjuntos, debes de cuidar que cada clase del conjunto de entrenamiento tenga como máximo 500 instancias. Por otra parte, cada clase del conjunto de validación debe tener mínimo 200 instancias, mientras que cada clase del conjunto de prueba, debe contener por lo menos 100 instancias. ES OBLIGATORIO QUE LOS CONJUNTOS DE ENTRENAMIENTO, VALIDACIÓN Y PRUEBA, LOS GENERES MEDIANTE CÓDIGO, NO PUEDES HACERLO DE FORMA MANUAL.
- Debes desplegar las gráficas de precisión y pérdida en los conjuntos de entrenamiento y validación.
- Debes guardar el modelo que tenga el mejor desempeño en el conjunto de validación, para ello, emplea **callbacks**.
- Además de utilizar la métrica **acc** para evaluar el desempeño de tu MEJOR MODELO en el conjunto de prueba, debes desplegar la **matriz de confusión** que muestre el rendimiento de tu MEJOR MODELO en el conjunto de prueba. También tienes que mostrar el valor de la métrica **F1** con respecto al rendimiento de tu MEJOR MODELO en el conjunto de prueba.

Los requisitos anteriores deben de cumplirse para que tengas derecho a que tu implementación reciba una calificación. Considerando lo anterior, los valores que puede tener tu implementación son los siguientes:

- a) Si la precisión en el conjunto de prueba alcanzada por tu MEJOR MODELO es menor a 0.80, obtendrás 0 puntos.
- b) Si la precisión en el conjunto de prueba alcanzada por tu MEJOR MODELO corresponde a un valor entre 0.80 - 0.85, obtendrás 10 puntos.
- c) Si la precisión en el conjunto de prueba alcanzada por tu MEJOR MODELO es mayor a 0.85 y menor o igual a 0.90, obtendrás 20 puntos.
- d) Si la precisión en el conjunto de prueba alcanzada por tu MEJOR MODELO es mayor a 0.90 y menor o igual a 0.95, obtendrás 30 puntos.
- e) Si la precisión en el conjunto de prueba alcanzada por tu MEJOR MODELO es mayor a 0.95, obtendrás 40 puntos.

NOTA: En un mensaje privado de Slack, compárteme el notebook (s) que contenga la solución a este problema. También, tienes que compartirme el modelo que tenga el mejor desempeño en el conjunto de validación.

Valor máximo: 40 puntos

A considerar: la calificación mínima aprobatoria es de 70 puntos.