

Desarrolle un predictor del precio del bitcoin usando una red neuronal recurrente (RNN, por sus siglas en inglés). El archivo con los datos se puede descargar en [este enlace](#). Este se puede cargar con [pandas.read_feather](#).

```
import pandas as pd
filename = "bitcoin_open_prices.feather"
pd.read_feather(filename)
```

En Google Colab, se puede descargar directamente el archivo con:

```
!wget
https://github.com/fjme95/datasets/blob/main/bitcoin_open_prices.feather?raw=
=true -O bitcoin_open_prices.feather
```

Realice los siguientes pasos para la predicción:

- Obtenga el primer 70% de los datos para el entrenamiento y los 30% restantes para las pruebas.
- Escale los datos de entrenamiento usando `sklearn.preprocessing.MinMaxScaler`.
- Considere `X_train` desde el primer elemento al penúltimo del dataset de entrenamiento y `y_train` del segundo al último.
- Entrene una **red secuencial** con las siguientes capas:
 - LSTM con activación sigmoide y dimensionalidad del espacio de salida de 4.
 - Capa densa con salida de dimensión 1.

Para la red neuronal, utilice el optimizador adam y el error cuadrático medio como función de pérdida.

Una vez que obtenga las predicciones, regrese éstas al espacio original (`sklearn.preprocessing.MinMaxScaler.inverse_transform`) y realice una gráfica comparando los datos reales con los predichos por la red.

El entregable deberá ser un archivo .ipynb que contenga:

1. Exploración de los datos
2. Los pasos del algoritmo
3. Evaluación de la predicción
4. Gráfica comparando los datos reales con los arrojados por el modelo.
5. Conclusiones.

Extra:

- ¿El modelo “sobre ajustó” los datos?
 - En caso de ser afirmativa la respuesta, agregué una capa de Dropout al modelo y pruebe con diferentes optimizadores y funciones de activación.