Opportunities@MeLi

Code Exercise

# Data Scientist

### Description:

In the context of Mercadolibre's Marketplace an algorithm is needed to predict if an item listed in the marketplace is new or used.

Your tasks involve the data analysis, designing, processing and modeling of a machine learning solution to predict if an item is new or used and then evaluate the model over held-out test data.

To assist in that task a dataset is provided in `MLA\_100k\_checked\_v3.jsonlines` and a function `build\_dataset` to read that dataset in `new\_or\_used.py`.

For the evaluation, you will use the accuracy metric in order to get a result of 0.86 as minimum. Additionally, you will have to choose an appropriate secondary metric and also elaborate an argument on why that metric was chosen.

### The deliverables are:

* The file, including all the code needed to define and evaluate a model.
* A document with an explanation on the criteria applied to choose the features, the proposed secondary metric and the performance achieved on that metrics. Optionally, you can deliver an EDA analysis with other format like .ipynb

**Solución Test MeLi**

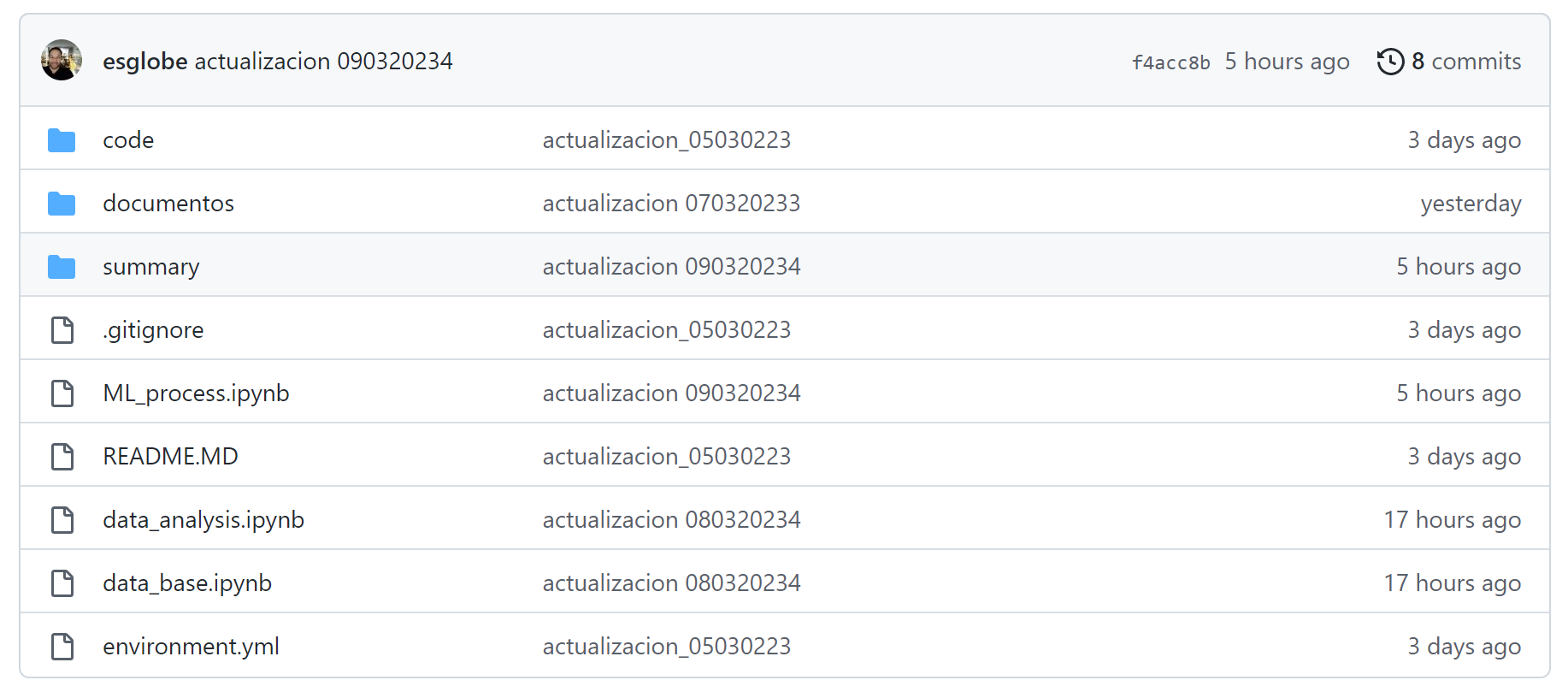
Javier Martínez

**Resumen**

Los desarrollos se encuentran disponibles en el repositorio público *test\_mercado\_libre* (https://github.com/esglobe/test\_mercado\_libre). La solución fue dividida en tres secciones, la primera, destinada a la creación de una base de datos tabular (data\_base.ipynb). Para luego dar paso al análisis de datos (data\_analysis.ipynb) y finalmente, el proceso para el ajuste o entrenamiento de modelos (ML\_process.ipynb). En la búsqueda del mejor modelo (accuracy mayor a 0.86 como mínimo) se genero un proceso que aplica los algoritmos de Decision Tree, Random Forest, Gradient boosting y eXtreme Gradient Boosting desde la percepción de la clasificación binaria (new=1 y used=0). Para cada uno de los algoritmos se implementó una grilla en los parámetros del modelo con el propósito de evaluar la variación en el desempeño (performance). En definitiva, el mejor modelo encontrado alcanzó un accuracy de 0.8651 tras la aplicación del algoritmo eXtreme Gradient Boosting con n\_estimators = 300. Las tres variables (features) más significativas fueron el initial\_quantily, listing\_type\_id\_frre y accepts\_mercadopago.

**El repositorio test\_mercado\_libre**

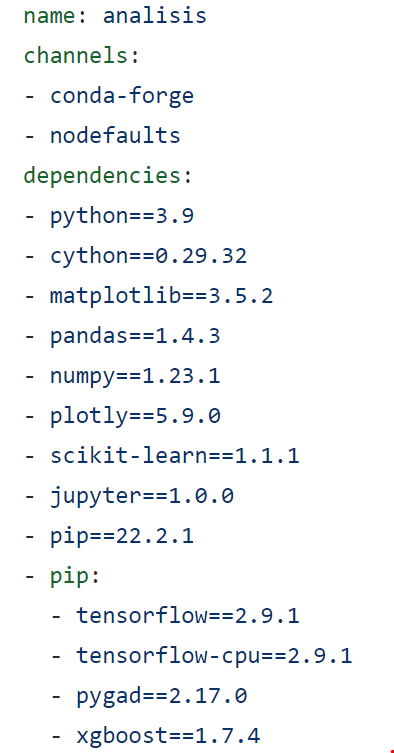
En la búsqueda de la solución se creó el repositorio público test\_mercado\_libre (https://github.com/esglobe/test\_mercado\_libre). El cual dispone, en su rama master, de la siguiente estructura:



Tal que;

1. **code:** Contiene el código new\_or\_used.py para la lectura de los datos.
2. **documentos:** Dispone de los criterios para el desarrollo del test (Opportunities@MeLi - CodeExercise DS\_ML.docx).
3. **summary:** Tiene el resumen tras el entrenamiento de los modelos.
4. **ML\_process.ipynb:** El proceso creado para el ajuste o entrenamiento de los modelos.
5. **README.MD**: Comandos para la creación del conda environment.
6. **data\_analysis.ipynb:** Los desarrollos destinados al estudio de los datos.
7. **data\_base.ipynb:** Mecanismo de procesamiento de la información.
8. **environment.yml:** Creación del conda environment (analisis).

El ambiente de desarrollo local dispone de las siguientes librerías y dependencias:



**Procesamiento de la base de datos**