* ¿Con qué propósito utilizamos los objetos OneHotEncoder y MinMaxScaler?

Estos dos objetos tienen como objetivo preparar los datos, no obstante, ambos se usan en diferente tipo de datos:

·        Datos categóricos: Para los datos categóricos, es decir, aquellas columnas con respuestas con “respuestas de opción multiple” el OneHotEncoder lo que hace es transformar esas respuestas en columnas con uno en caso de esta presentes y cero en caso contrario, pasando a ser columnas numéricas binarias. Para quienes vengan de la analítica de datos se preguntarán ¿Qué diferencia hay entre OneHotEncoder y get dummies de la librería pandas? Y es que estas dos funciones son muy semejantes entre sí, pues hacen lo mismo, pero cada una tiene un objetivo, get dumies se usa mas en un EDA o bien para hacer gráficos con las columnas binarias mientras que OneHotEncoding esa dirigido a los procesos de machine learning

·        Datos numéricos: Al tener una columna con datos categóricos, estos pueden ser bastante dispersos, por lo cual, trabajar con ellos de esta forma nos traería muchos problemas a la hora de hacer un proceso de machine learning, pues la similitud, predicción u objetivo a cumplir tendría una probabilidad demasiado baja, por ello es que la funcion MinMaxScaler nos permite llevar estos datos numéricos a un rango entre 1 y 0, así se mejorara la probabilidad, pues estos números ya no estarían en un amplio rango, en pocas palabras, se redimensionan estos datos a un rango entre 1 y 0 para mejorar el rendimiento.

* ¿Cuál es el beneficio de utilizar un Pipeline en proceso de creación de un modelo de ML?

El usar pipeline nos permite ahorrarnos el código para juntar todas las transformaciones que hemos realizado anteriormente o bien, los procesos mencionados en la respuesta anterior y con ello aplicar el entrenamiento de nuestro modelo. Es más simple trabajar con los pipelines que hacer todo por separado y después andar unión cada punto como se hace con pandas, sin mencionar a la partición de test y training que a mi consideración, aunque sea más fácil por la experiencia y uso de pandas es más laborioso, lleva más líneas de código a mi parecer.

* ¿Cuál es la diferencia entre los métodos readStream() y writeStream()?

Estos métodos se usan en procesos de strremaing, conforme lo indican sus nombres

·        readStream se encarga de leer los datos en tiempo real, es como ver la ingesta de datos en tiempo real, pero este ya en procesos de machine learnig que es precisamente lo que hacen las plataformas de streaming por atrás, estar monitoreando nuestra actividad para hacer recomendaciones en tiempo real.

·        WriteStream por otro lado nos permite ya hacer la manipulación de los datos captados al momento, es decir el pos a la lectura de los datos

Estos procesos son completamente distintos pero complementarios entre sí, pues Write stream depende de ReadStream y ReadStream necesita de WriteStream para poder transformar los datos en información y conocimiento. Un caso que se me ocurre en esto y esta presenta hoy en día es la bolsa de valores o las divisas, estas van cambiando en tiempo real con la compra, venta y movimiento del mercado, lo cual entraría a readStream pero de nada sirve solo leer los datos si no se presentan de forma correcta, por lo que estas transformaciones ya sea get dumies para hacer los gráficos y tener un análisis en tiempo real o OnetoEncoder para construir un modelo de compra/venta optimizado se encuentra en la parte de WriteStream.

* ¿Qué métricas podemos obtener del evaluador de la clase MulticlassClassificationEvaluator?

Precision, Recall, Accuracy, Precision by label, Recall by label, F1-measure by label, Hamming Loss, Subset Accuracy, F1 Measure, Micro precision, Micro recall, Micro F1 Measure