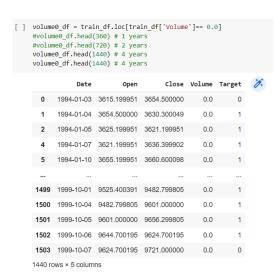
- Realizado para resolver el challenge
  - He utilizado una red neuronal recurrente para poder tener en cuenta los valores anteriores del Ibex para calcular los próximos valores.
  - He escogido una arquitectura Long short-term memory (LSTM) porque he observado que de momento son las que mejores resultados están dando con el objetivo de predecir la bolsa.
- Idea para resolver el challenge (No realizado).
  - Utilizaría los datos de Twitter para evaluar si el estado de ánimo era bueno o malo, teniendo en cuenta los adjetivos positivos y negativos observados.
    - Utilizando el TFID Vectorizer de Sklearn
    - Con los datos obtenidos entrenar un clasificador de tipo Support Vector Machine con Kernel lineal y Classificador de Bayes, quedándome con el que me diera mejores resultados.
  - Después entrenaría una red nueva para mejorar los resultados de la LSTM con los estimador de estado emocional de Twitter, teniendo los valores de estos como entrada.

En el proceso de analizar los datos se ha detectado:

 Que los datos no estaban completos, pero solo eran 100 datos de 6500 por lo que se podían eliminar sin peligro.

```
print(train df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6554 entries, 0 to 6553
Data columns (total 8 columns):
                Non-Null Count Dtype
     Column
     Date
                6554 non-null
                               object
                6421 non-null
                               float64
    Open
    High
                6421 non-null
                               float64
                6421 non-null
                              float64
    Low
     Close
                6421 non-null
                               float64
    Adi Close 6421 non-null
                              float64
    Volume
                6421 non-null
                               float64
    Target
                6554 non-null
dtypes: float64(6), int64(1), object(1)
memory usage: 409.8+ KB
None
```

 Que en los primeros años no habían valores del volumen hasta el 2000, por lo que si se utiliza, tenemos que eliminar esos datos.



En el proceso de analizar los datos se ha observado:

 Que la mayoría de datos estaban correlacionados entre si y tenían valores muy parecidos.

```
con el feature del volumen.
print(train df.describe())
      6421,000000
                                                                                                                            print(train df['Target'].corr(train df['Adj Close']))
      8936.540448
                                                 8934.970624
                                                              print(train_df['Open'].corr(train_df['Adj Close']))
                                                                                                                             print(train df['Target'].corr(train df['Open']))
                                                              print(train df['Open'].corr(train df['Close']))
                 2877.300049
                 7817,200195
                                                              print(train df['Open'].corr(train df['High']))
                                                                                                                             print(train df['Target'].corr(train df['High']))
                                                              print(train df['Open'].corr(train df['Low']))
                                                                                                                             print(train_df['Target'].corr(train_df['Low']))
                           10441.200195
                                                              print(train df['Open'].corr(train df['Volume']))
                          15868.599609 15945.700195 15945.683594
                                                                                                                             print(train df['Target'].corr(train df['Close']))
                                                                                                                             print(train df['Target'].corr(train df['Volume']))
                    Target
                                                              0.9991869125864032
                6421.000000
                                                              0.9991869122480536
                                                                                                                             -0.017413997649211612
     1.231845e+08
                  0.499274
                                                              0.9996620595980537
     0.000000e+00
                  0.000000
                                                                                                                             -0.017777092372831756
                                                              0.9995332248169949
                                                                                                                             -0.017758741226730563
                  1.000000
                                                              0.13415866925589653
                                                                                                                             -0.01765471331507807
                                                                                                                             -0.01741399386196035
                                                                                                                             0.0003054249426972032
```

Que el valor objetivo estaba

también correlacionado con los

features, pero en menor grado

#### Decisiones y resultados:

- Teniendo en cuenta los valores anteriores, se ha decidido que el feature que tenía más sentido para entrenar la red era el valor de inicio de la bolsa de cada día. A pesar de esto, se ha decidido añadir el valor del volumen y el valor de cierre de cada día para intentar mejorar los resultados, eliminando los datos no completos.
- Se han normalizado todos los datos entre 0 y 1.
- Los datos se han pasado en batches de 10 días para que la red pudiera tener en cuenta los 9 días anteriores a la predicción ( el periodo de tiempo podría ser optimizado)
- Se ha dividido el data set de entrenamiento en 70% para entrenar y 30% para validar.
- Se ha obteniendo f-score en la validación de 0.516.

• A continuación se muestra la matriz de confusión y el grafico de evolución de la loss, donde se observa una Pareto.

```
[ ] print(metrics.confusion_matrix(y_val,y_pred))
[[297 394]
      [344 394]]
```

