

## Caixa Bank – Nuwe -28/05/22 -Data Science Challenge – Sergi Ponsà Cobas

- 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.

# Caixa Bank – Nuwe -28/05/22 -Data Science Challenge – Sergi Ponsà Cobas

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

```
[ ] print(train_df.info())
```

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

```
[ ] volume0_df = train_df.loc[train_df['Volume']== 0.0]
#volume0_df.head(360) # 1 years
#volume0_df.head(720) # 2 years
volume0_df.head(1440) # 4 years
volume0_df.head(1440) # 4 years
```

	Date	Open	Close	Volume	Target
0	1994-01-03	3615.199951	3654.500000	0.0	0
1	1994-01-04	3654.500000	3630.300049	0.0	1
2	1994-01-05	3625.199951	3621.199951	0.0	1
4	1994-01-07	3621.199951	3636.399902	0.0	1
5	1994-01-10	3655.199951	3660.600098	0.0	1
...	...	...	...	...	...
1499	1999-10-01	9525.400391	9482.799805	0.0	1
1500	1999-10-04	9482.799805	9601.000000	0.0	1
1501	1999-10-05	9601.000000	9656.299805	0.0	1
1502	1999-10-06	9644.700195	9624.700195	0.0	1
1503	1999-10-07	9624.700195	9721.000000	0.0	0

1440 rows x 5 columns

# Caixa Bank – Nuwe -28/05/22 -Data Science Challenge – Sergi Ponsà Cobas

En el proceso de analizar los datos se ha observado:

- Que la mayoría de datos estaban correlacionados entre si y tenían valores muy parecidos.
- Que el valor objetivo estaba también correlacionado con los features, pero en menor grado con el feature del volumen.

```
[ ] print(train_df.describe())
```

	Open	High	Low	Close	Adj Close
count	6421.000000	6421.000000	6421.000000	6421.000000	6421.000000
mean	8936.540448	9005.235576	8858.340567	8934.978558	8934.970624
std	2732.102441	2749.009324	2712.511028	2731.032625	2731.030170
min	2865.100098	2877.300049	2833.600098	2865.100098	2865.097168
25%	7732.399902	7817.200195	7641.500000	7727.799805	7727.791992
50%	9329.700195	9404.599609	9243.000000	9331.000000	9331.000000
75%	10525.500000	10590.299805	10441.200195	10523.400391	10523.400391
max	15999.200195	16040.400391	15868.599609	15945.700195	15945.683594

  

	Volume	Target
count	6.421000e+03	6421.000000
mean	8.218074e+07	0.527644
std	1.231845e+08	0.499274
min	0.000000e+00	0.000000
25%	0.000000e+00	0.000000
50%	1.966000e+05	1.000000
75%	1.773980e+08	1.000000
max	7.894902e+08	1.000000

```
[ ] print(train_df['Open'].corr(train_df['Adj Close']))
print(train_df['Open'].corr(train_df['Close']))
print(train_df['Open'].corr(train_df['High']))
print(train_df['Open'].corr(train_df['Low']))
print(train_df['Open'].corr(train_df['Volume']))
```

```
0.9991869125864032
0.9991869122480536
0.9996620595980537
0.9995332248169949
0.13415866925589653
```

```
[ ] print(train_df['Target'].corr(train_df['Adj Close']))
print(train_df['Target'].corr(train_df['Open']))
print(train_df['Target'].corr(train_df['High']))
print(train_df['Target'].corr(train_df['Low']))
print(train_df['Target'].corr(train_df['Close']))
print(train_df['Target'].corr(train_df['Volume']))
```

```
-0.017413997649211612
-0.017777092372831756
-0.017758741226730563
-0.01765471331507807
-0.01741399386196035
0.0003054249426972032
```

## Caixa Bank – Nuwe -28/05/22 -Data Science Challenge – Sergi Ponsà Cobas

### 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 obtenido 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]]
```

