act2_uresti

September 11, 2025

---

1 Actividad Modulo 2

1.0.1 Segunda entrega

Yose Miguel Sotomayor Carneado A0150908

[ ]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt


from sklearn.datasets import fetch_kddcup99
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, ConfusionMatrixDisplay
from sklearn.preprocessing import StandardScaler, LabelEncoder, PowerTransformer

from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression

import warnings
warnings.filterwarnings("ignore")
```

---

[3]:
```python
data = fetch_kddcup99(as_frame=True)

X = data.data
y = data.target.astype(str)
```

[4]:
```python
encoder = LabelEncoder()

y = encoder.fit_transform(y)
np.unique(y)
```

```
[4]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
            17, 18, 19, 20, 21, 22])
```

```
[60]: pd.DataFrame(y).value_counts()
```

```
[60]: 0
      18    280790
      9     107201
      11     97278
      0       2203
      17      1589
      5       1247
      15      1040
      21      1020
      20       979
      14       264
      10       231
      3         53
      1         30
      6         21
      22        20
      4         12
      16        10
      7          9
      2          8
      8          7
      13         4
      12         3
      19         2
      Name: count, dtype: int64
```

```
[5]: X
```

```
[5]:        duration protocol_type  service   flag src_bytes dst_bytes land  \
     0             0        b'tcp'  b'http'  b'SF'       181      5450    0
     1             0        b'tcp'  b'http'  b'SF'       239       486    0
     2             0        b'tcp'  b'http'  b'SF'       235      1337    0
     3             0        b'tcp'  b'http'  b'SF'       219      1337    0
     4             0        b'tcp'  b'http'  b'SF'       217      2032    0
     …           …            …       …     …      …         …  …
     494016        0        b'tcp'  b'http'  b'SF'       310      1881    0
     494017        0        b'tcp'  b'http'  b'SF'       282      2286    0
     494018        0        b'tcp'  b'http'  b'SF'       203      1200    0
     494019        0        b'tcp'  b'http'  b'SF'       291      1200    0
     494020        0        b'tcp'  b'http'  b'SF'       219      1234    0

            wrong_fragment urgent hot  … dst_host_count dst_host_srv_count  \
0                      0       0   0  …              9                    9
1                      0       0   0  …             19                   19
2                      0       0   0  …             29                   29
3                      0       0   0  …             39                   39
4                      0       0   0  …             49                   49
…                …     …   ..  …             …                 …
494016                 0       0   0  …             86                  255
494017                 0       0   0  …              6                  255
494018                 0       0   0  …             16                  255
494019                 0       0   0  …             26                  255
494020                 0       0   0  …              6                  255

        dst_host_same_srv_rate dst_host_diff_srv_rate  \
0                          1.0                    0.0
1                          1.0                    0.0
2                          1.0                    0.0
3                          1.0                    0.0
4                          1.0                    0.0
…                          …                      …
494016                     1.0                    0.0
494017                     1.0                    0.0
494018                     1.0                    0.0
494019                     1.0                    0.0
494020                     1.0                    0.0

        dst_host_same_src_port_rate dst_host_srv_diff_host_rate  \
0                              0.11                         0.0
1                              0.05                         0.0
2                              0.03                         0.0
3                              0.03                         0.0
4                              0.02                         0.0
…                               …                           …
494016                         0.01                        0.05
494017                         0.17                        0.05
494018                         0.06                        0.05
494019                         0.04                        0.05
494020                         0.17                        0.05

        dst_host_serror_rate dst_host_srv_serror_rate dst_host_rerror_rate  \
0                        0.0                      0.0                  0.0
1                        0.0                      0.0                  0.0
2                        0.0                      0.0                  0.0
3                        0.0                      0.0                  0.0
4                        0.0                      0.0                  0.0
…                      …                      …                  …
494016                   0.0                     0.01                  0.0
494017                   0.0                     0.01                  0.0
```

```
494018                 0.06                     0.01                     0.0
494019                 0.04                     0.01                     0.0
494020                  0.0                     0.01                     0.0

       dst_host_srv_rerror_rate
0                           0.0
1                           0.0
2                           0.0
3                           0.0
4                           0.0
…                         …
494016                      0.0
494017                      0.0
494018                      0.0
494019                      0.0
494020                      0.0

[494021 rows x 41 columns]
```

[10]:

```
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 494021 entries, 0 to 494020
Data columns (total 41 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   duration                     494021 non-null  object
 1   protocol_type                494021 non-null  object
 2   service                      494021 non-null  object
 3   flag                         494021 non-null  object
 4   src_bytes                    494021 non-null  object
 5   dst_bytes                    494021 non-null  object
 6   land                         494021 non-null  object
 7   wrong_fragment               494021 non-null  object
 8   urgent                       494021 non-null  object
 9   hot                          494021 non-null  object
 10  num_failed_logins            494021 non-null  object
 11  logged_in                    494021 non-null  object
 12  num_compromised              494021 non-null  object
 13  root_shell                   494021 non-null  object
 14  su_attempted                 494021 non-null  object
 15  num_root                     494021 non-null  object
 16  num_file_creations           494021 non-null  object
 17  num_shells                   494021 non-null  object
 18  num_access_files             494021 non-null  object
 19  num_outbound_cmds            494021 non-null  object
 20  is_host_login                494021 non-null  object
 21  is_guest_login               494021 non-null  object
 22  count                        494021 non-null  object
 23  srv_count                    494021 non-null  object
 24  serror_rate                  494021 non-null  object
 25  srv_serror_rate              494021 non-null  object
 26  rerror_rate                  494021 non-null  object
 27  srv_rerror_rate              494021 non-null  object
 28  same_srv_rate                494021 non-null  object
 29  diff_srv_rate                494021 non-null  object
 30  srv_diff_host_rate           494021 non-null  object
 31  dst_host_count               494021 non-null  object
 32  dst_host_srv_count           494021 non-null  object
 33  dst_host_same_srv_rate       494021 non-null  object
 34  dst_host_diff_srv_rate       494021 non-null  object
 35  dst_host_same_src_port_rate  494021 non-null  object
 36  dst_host_srv_diff_host_rate  494021 non-null  object
 37  dst_host_serror_rate         494021 non-null  object
 38  dst_host_srv_serror_rate     494021 non-null  object
 39  dst_host_rerror_rate         494021 non-null  object
 40  dst_host_srv_rerror_rate     494021 non-null  object
dtypes: object(41)
memory usage: 154.5+ MB
```

[11]:
```
X.isna().sum()
```

[11]:
```
duration                       0
protocol_type                  0
service                        0
flag                           0
src_bytes                      0
dst_bytes                      0
land                           0
wrong_fragment                 0
urgent                         0
hot                            0
num_failed_logins              0
logged_in                      0
num_compromised                0
root_shell                     0
su_attempted                   0
num_root                       0
num_file_creations             0
num_shells                     0
num_access_files               0
num_outbound_cmds              0
is_host_login                  0
is_guest_login                 0
count                          0
```

```
srv_count                      0
serror_rate                    0
srv_serror_rate                0
rerror_rate                    0
srv_rerror_rate                0
same_srv_rate                  0
diff_srv_rate                  0
srv_diff_host_rate             0
dst_host_count                 0
dst_host_srv_count             0
dst_host_same_srv_rate         0
dst_host_diff_srv_rate         0
dst_host_same_src_port_rate    0
dst_host_srv_diff_host_rate    0
dst_host_serror_rate           0
dst_host_srv_serror_rate       0
dst_host_rerror_rate           0
dst_host_srv_rerror_rate       0
dtype: int64
```

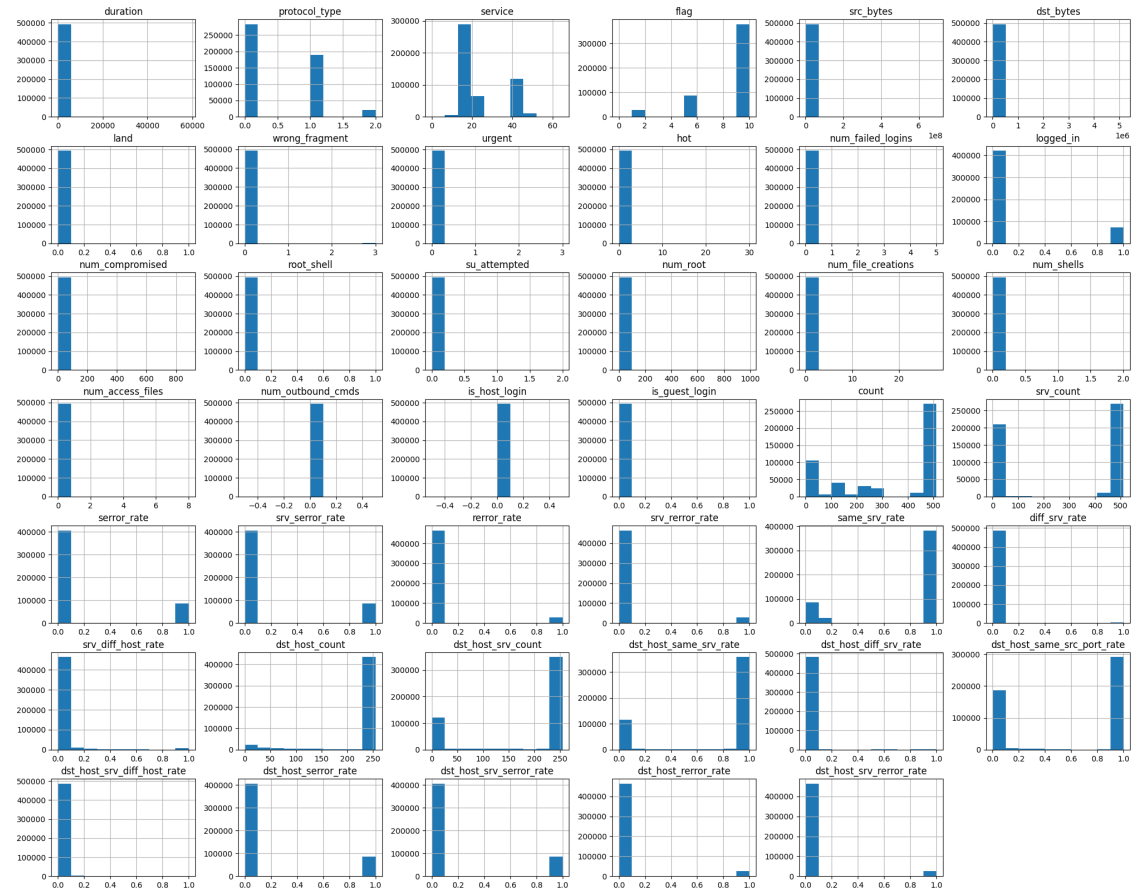
[19]:
```
# MAKE NUM CATEGORICAL and astype float

X['protocol_type'] = encoder.fit_transform(X['protocol_type'])
X['service'] = encoder.fit_transform(X['service'])
X['flag'] = encoder.fit_transform(X['flag'])

def make_float(x):
    return float(x)

for col in X.columns:
    X[col] = X[col].apply(make_float)

X
```

[19]:
```
        duration  protocol_type  service  flag  src_bytes  dst_bytes  land  \
0            0.0            1.0     22.0   9.0      181.0     5450.0   0.0
1            0.0            1.0     22.0   9.0      239.0      486.0   0.0
2            0.0            1.0     22.0   9.0      235.0     1337.0   0.0
3            0.0            1.0     22.0   9.0      219.0     1337.0   0.0
4            0.0            1.0     22.0   9.0      217.0     2032.0   0.0
…            …              …        …     …          …  …
494016       0.0            1.0     22.0   9.0      310.0     1881.0   0.0
494017       0.0            1.0     22.0   9.0      282.0     2286.0   0.0
494018       0.0            1.0     22.0   9.0      203.0     1200.0   0.0
494019       0.0            1.0     22.0   9.0      291.0     1200.0   0.0
494020       0.0            1.0     22.0   9.0      219.0     1234.0   0.0
```

```
        wrong_fragment  urgent  hot  …  dst_host_count  dst_host_srv_count  \
0                  0.0     0.0  0.0  …             9.0                 9.0
1                  0.0     0.0  0.0  …            19.0                19.0
2                  0.0     0.0  0.0  …            29.0                29.0
3                  0.0     0.0  0.0  …            39.0                39.0
4                  0.0     0.0  0.0  …            49.0                49.0
…                    …       …    …  …               …                   …
494016             0.0     0.0  0.0  …            86.0               255.0
494017             0.0     0.0  0.0  …             6.0               255.0
494018             0.0     0.0  0.0  …            16.0               255.0
494019             0.0     0.0  0.0  …            26.0               255.0
494020             0.0     0.0  0.0  …             6.0               255.0

        dst_host_same_srv_rate  dst_host_diff_srv_rate  \
0                          1.0                     0.0
1                          1.0                     0.0
2                          1.0                     0.0
3                          1.0                     0.0
4                          1.0                     0.0
…                            …                       …
494016                     1.0                     0.0
494017                     1.0                     0.0
494018                     1.0                     0.0
494019                     1.0                     0.0
494020                     1.0                     0.0

        dst_host_same_src_port_rate  dst_host_srv_diff_host_rate  \
0                              0.11                         0.00
1                              0.05                         0.00
2                              0.03                         0.00
3                              0.03                         0.00
4                              0.02                         0.00
…                                 …                            …
494016                         0.01                         0.05
494017                         0.17                         0.05
494018                         0.06                         0.05
494019                         0.04                         0.05
494020                         0.17                         0.05

        dst_host_serror_rate  dst_host_srv_serror_rate  dst_host_rerror_rate  \
0                       0.00                      0.00                   0.0
1                       0.00                      0.00                   0.0
2                       0.00                      0.00                   0.0
3                       0.00                      0.00                   0.0
4                       0.00                      0.00                   0.0
…                          …                         …                     …
494016                  0.00                      0.01                   0.0
```

```
494017                 0.00                      0.01                   0.0
494018                 0.06                      0.01                   0.0
494019                 0.04                      0.01                   0.0
494020                 0.00                      0.01                   0.0

        dst_host_srv_rerror_rate
0                            0.0
1                            0.0
2                            0.0
3                            0.0
4                            0.0
…                              …
494016                       0.0
494017                       0.0
494018                       0.0
494019                       0.0
494020                       0.0

[494021 rows x 41 columns]
```

[27]:
```
X.hist(figsize=(25, 20))
plt.show()
```



[20]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
↪random_state=42)
```

Como es un problema de clasificacion multiclase vamos a usar onehotencoder para las clases de la y

[54]:
```
# Columnas categóricas
cat_cols = ['protocol_type', 'service', 'flag']

# Columnas numéricas
num_cols = [c for c in X.columns if c not in cat_cols]

preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), num_cols)
    ]
)

transformer = Pipeline(
        steps=[
            ("power", PowerTransformer(method="yeo-johnson", standardize=True)),
        ]
    )

clf = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('transform', transformer),
    ('classifier', LogisticRegression(max_iter=1000, penalty='l2',␣
↪multi_class='auto' ))
])

clf.fit(X_train, y_train)
```

[54]:
```
Pipeline(steps=[('preprocessor',
                 ColumnTransformer(transformers=[('num', StandardScaler(),
                                                  ['duration', 'src_bytes',
                                                   'dst_bytes', 'land',
                                                   'wrong_fragment', 'urgent',
                                                   'hot', 'num_failed_logins',
                                                   'logged_in',
                                                   'num_compromised',
                                                   'root_shell', 'su_attempted',
                                                   'num_root',
                                                   'num_file_creations',
                                                   'num_shells',
                                                   'num_access_files',
                                                   'num_outbound_cmds',
                                                   'is_host_login',
                                                   'is_guest_login', 'count',
                                                   'srv_count', 'serror_rate',
                                                   'srv_serror_rate',
                                                   'rerror_rate',
                                                   'srv_rerror_rate',
                                                   'same_srv_rate',
                                                   'diff_srv_rate',
                                                   'srv_diff_host_rate',
                                                   'dst_host_count',
                                                   'dst_host_srv_count',
…])])),
                ('transform', Pipeline(steps=[('power', PowerTransformer())])),
                ('classifier',
                 LogisticRegression(max_iter=1000, multi_class='auto'))])
```

[55]:
```
y_pred = clf.predict(X_test)
```

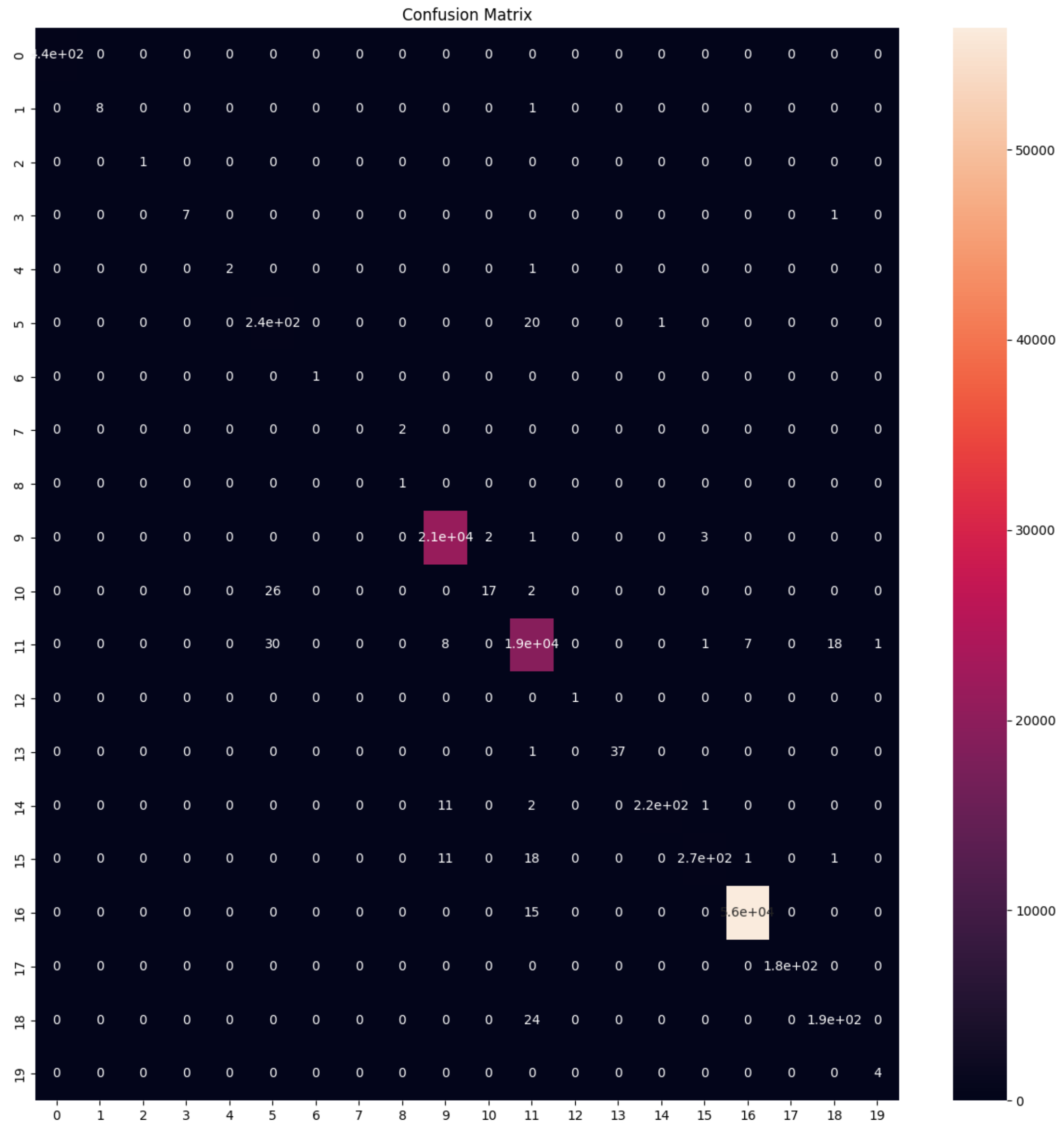
[56]:
```
print(classification_report(y_pred, y_test))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       435
           1       0.89      1.00      0.94         8
           2       1.00      1.00      1.00         1
           3       0.88      1.00      0.93         7
           4       0.67      1.00      0.80         2
           5       0.92      0.81      0.86       300
           6       1.00      1.00      1.00         1
           7       0.00      0.00      0.00         0
           8       1.00      0.33      0.50         3
           9       1.00      1.00      1.00     21318
          10       0.38      0.89      0.53        19
          11       1.00      1.00      1.00     19373
          12       1.00      1.00      1.00         1
          14       0.97      1.00      0.99        37
          15       0.94      1.00      0.97       223
          17       0.90      0.98      0.94       278
          18       1.00      1.00      1.00     56395
          20       1.00      1.00      1.00       185
          21       0.89      0.91      0.90       214
          22       1.00      0.80      0.89         5

    accuracy                           1.00     98805
```

```
   macro avg       0.87      0.89      0.86     98805
weighted avg       1.00      1.00      1.00     98805
```

Podemos observar que los resultados son bastante buenos.

[57]:
```python
from sklearn.metrics import confusion_matrix
import seaborn as sns
plt.figure(figsize=(15, 15))
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True)
plt.title("Confusion Matrix")
plt.show()
```



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	.4e+02	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	8	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
4	0	0	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	2.4e+02	0	0	0	0	0	20	0	0	1	0	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	2.1e+04	2	1	0	0	0	3	0	0	0	0
10	0	0	0	0	0	26	0	0	0	0	17	2	0	0	0	0	0	0	0	0
11	0	0	0	0	0	30	0	0	0	8	0	1.9e+04	0	0	0	1	7	0	18	1
12	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	1	0	37	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	11	0	2	0	0	2.2e+02	1	0	0	0	0
15	0	0	0	0	0	0	0	0	0	11	0	18	0	0	0	2.7e+02	1	0	1	0
16	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0	.6e+0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.8e+02	0	0
18	0	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	1.9e+02	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4

[58]:
```python
from sklearn.metrics import f1_score
print('F1 Macro' ,f1_score(y_test, y_pred, average='macro'))
print('F1 Micro' ,f1_score(y_test, y_pred, average='micro'))
print('F1 Weighted' ,f1_score(y_test, y_pred, average='weighted'))
```

```
F1 Macro 0.8621861403907598
F1 Micro 0.997874601487779
F1 Weighted 0.9978189895431658
```

Vemos que el modelo a grandes rasgos tiene unos buenos f1-score para evaluar el modelo con muchas clases y MUY mal balanceadas!