

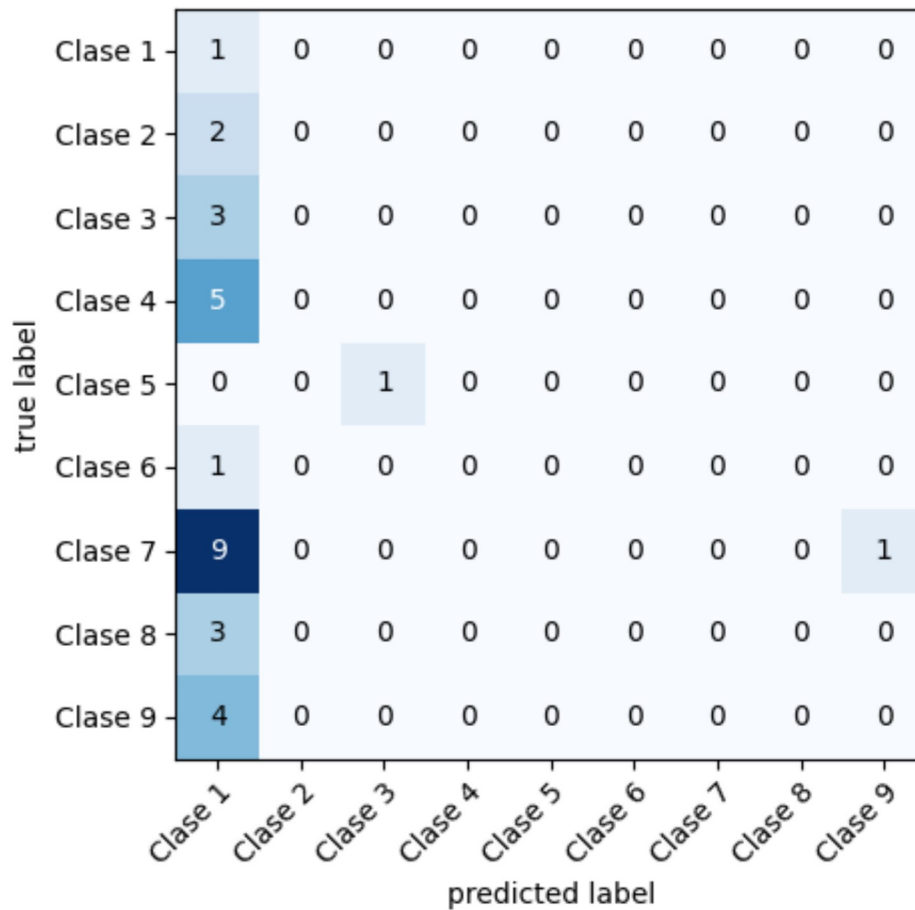
```
In [22]: #IMPORTACION DE LIBRERIAS PARA LA MUESTRA DE LA MATRIZ DE EFECTIVIDAD
from sklearn.metrics import confusion_matrix, f1_score, roc_curve, precision_sco
from sklearn import metrics
from mlxtend.plotting import plot_confusion_matrix
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
#DATOS SOBRE LAS FOTOGRAFIAS Y SU TAMAÑO DE GESTIONAMIENTO
width_shape = 100
height_shape = 100
batch_size = 1
#NUMERO DE CLASES
names = ['Clase 1', 'Clase 2', 'Clase 3', 'Clase 4', 'Clase 5',
         'Clase 6', 'Clase 7', 'Clase 8', 'Clase 9']
#FUENTE DE DATOS PARA TEST
test_data_dir = './Dataset/test'
#IMAGEDATAGENERATOR PARA LA CONFIGURACION DE RECUPERACION DE INFORMAION
test_datagen = ImageDataGenerator()
#CONFIGURACION DE FUNCION PARA GENERAR DATOS PARA EL TEST
test_generator = test_datagen.flow_from_directory(
    test_data_dir,
    target_size=(width_shape, height_shape),
    batch_size = batch_size,
    class_mode='categorical',
    shuffle=False)
#INFO DEL MODELO GENERADO A RAZA DEL ENTRENAMIENTO
custom_Model= load_model("./modelo/modelo.h5")
#GENERACION DE LAS PREDICCIONES
predictions = custom_Model.predict(test_generator)
y_pred = np.argmax(predictions, axis=1)
y_real = test_generator.classes
```

Found 30 images belonging to 10 classes.

30/30 [=====] - 2s 78ms/step

```
In [23]: matc=confusion_matrix(y_real, y_pred)
```

```
In [24]: #GRAFICACION DE LOS DATOS Y SUS PREDICCIONES
plot_confusion_matrix(conf_mat=matc, class_names = names, show_normed=False)
plt.tight_layout()
```



```
In [25]: #TABLA PARA VERIFICACION DE PARAMETROS DE CLASIFICACION
print(metrics.classification_report(y_real,y_pred, digits = 7))
```

	precision	recall	f1-score	support
1	0.0357143	1.0000000	0.0689655	1
2	0.0000000	0.0000000	0.0000000	2
3	0.0000000	0.0000000	0.0000000	3
4	0.0000000	0.0000000	0.0000000	5
5	0.0000000	0.0000000	0.0000000	1
6	0.0000000	0.0000000	0.0000000	1
7	0.0000000	0.0000000	0.0000000	10
8	0.0000000	0.0000000	0.0000000	3
9	0.0000000	0.0000000	0.0000000	4
accuracy			0.0333333	30
macro avg	0.0039683	0.1111111	0.0076628	30
weighted avg	0.0011905	0.0333333	0.0022989	30

```
C:\entornoPy\env\lib\site-packages\sklearn\metrics\_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
```

```
    _warn_prf(average, modifier, msg_start, len(result))
```

```
C:\entornoPy\env\lib\site-packages\sklearn\metrics\_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
```

```
    _warn_prf(average, modifier, msg_start, len(result))
```

```
C:\entornoPy\env\lib\site-packages\sklearn\metrics\_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
```

```
    _warn_prf(average, modifier, msg_start, len(result))
```

In []:

In []: