



DETECCIÓN DE RETINOPATÍA DIABÉTICA Y SUS ETAPAS

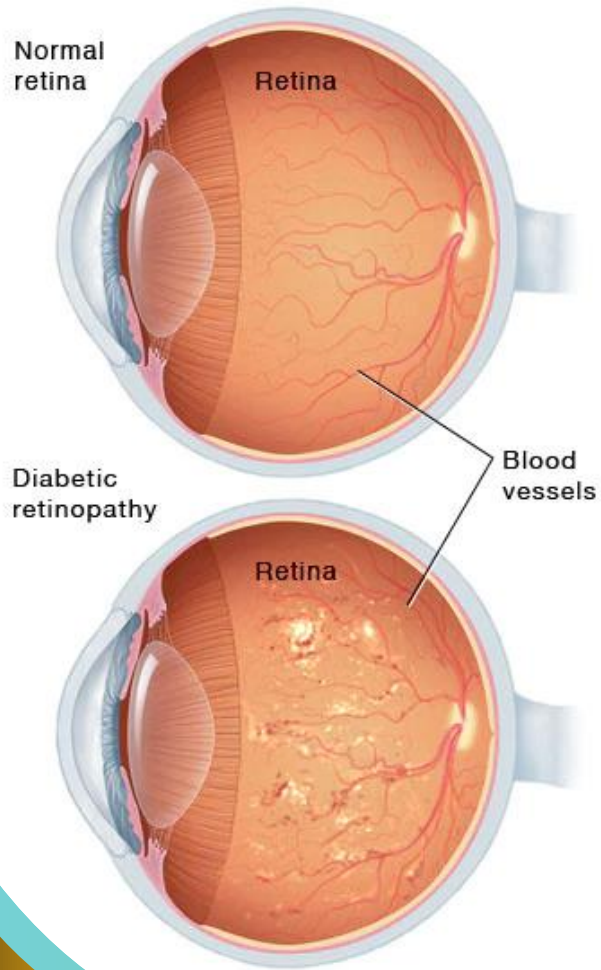
Nelson Alexis Cáceres Carreño
Jenny Marcela Santamaría Rincón

Inteligencia Artificial II



01

Detección de la
retinopatía diabética y
sus etapas



La retinopatía diabética es la principal causa de ceguera en la población mundial, esta enfermedad es una complicación de la diabetes que afecta los ojos y es causada por el daño a los vasos sanguíneos que van al tejido sensible a la luz que se encuentra en el fondo del ojo (retina).

Conjunto de datos



0 -> Sin
retinopatía
diabética



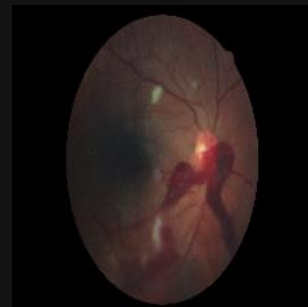
1 -> leve



2 -> moderada

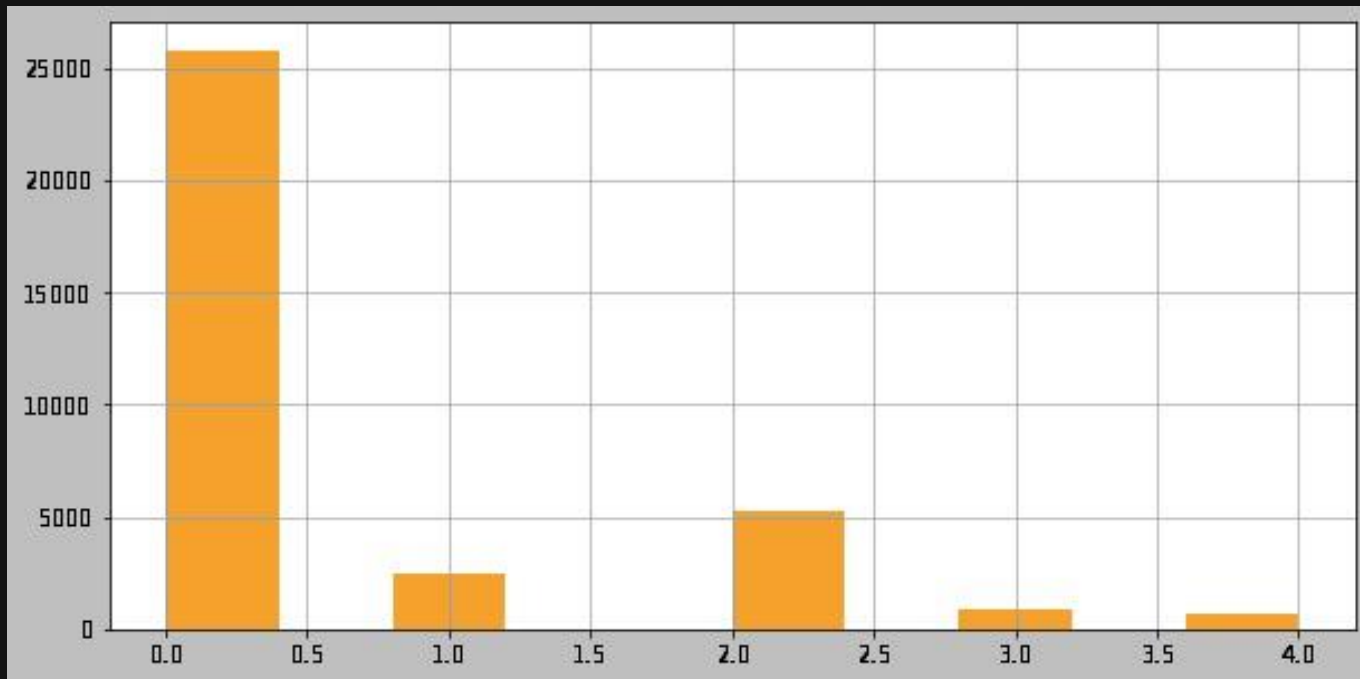


3 -> Grave



4 -> Retinopatía
diabética
proliferativa

Datos originales



Conjunto de datos



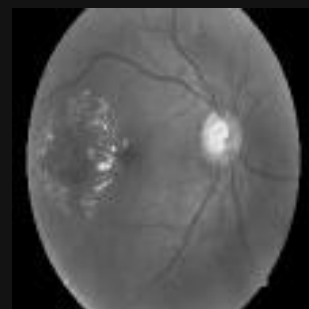
0 -> Sin
retinopatía
diabética



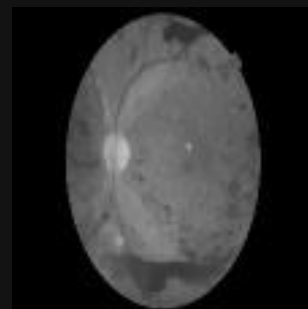
1 -> leve



2 -> moderada



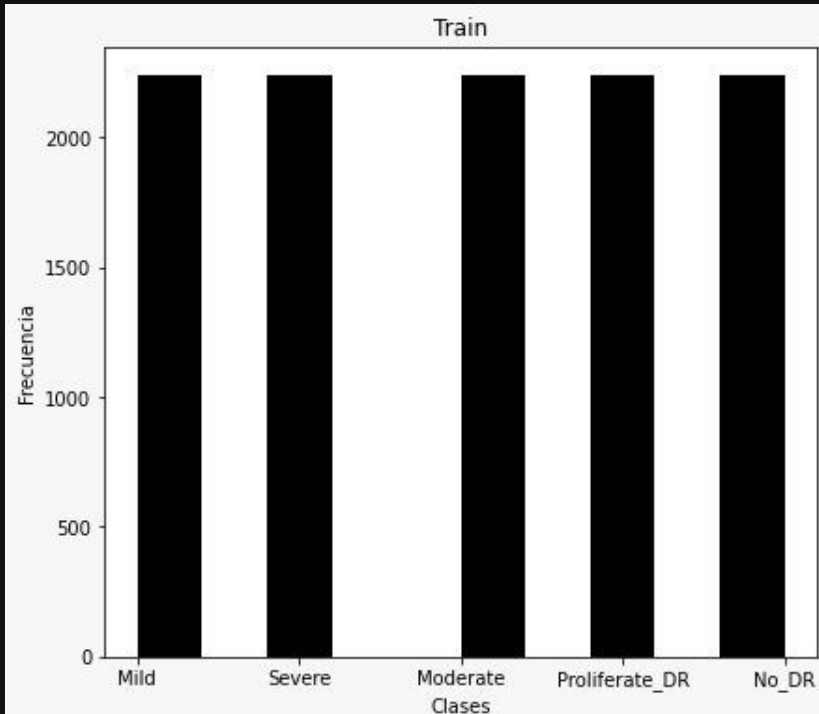
3 -> Grave



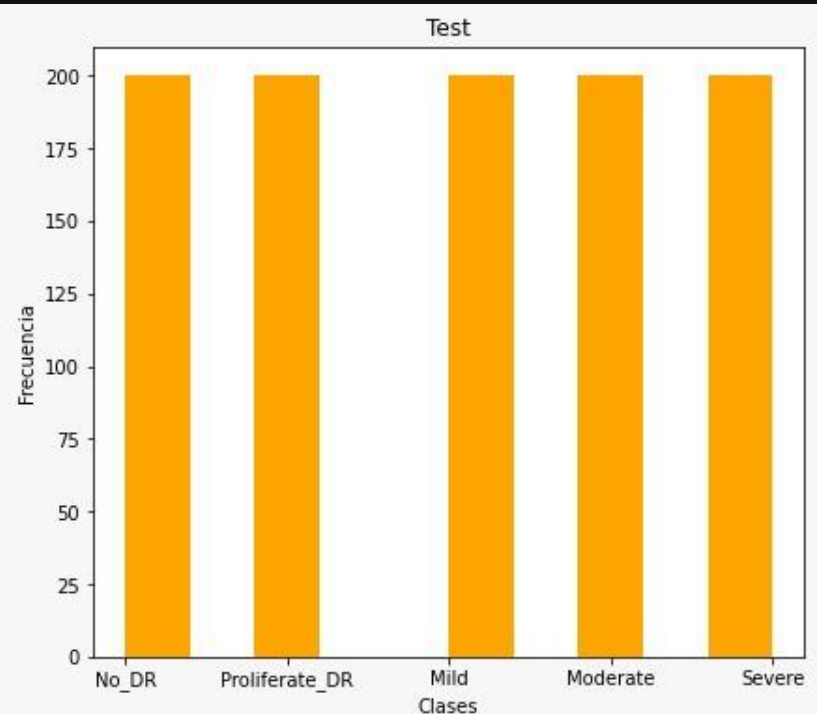
4 -> Retinopatía
diabética
proliferativa

Data augmentation

Train



Test



CALLBACKS

01

```
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(  
    monitor='val_f1_score',  
    factor=0.5,  
    patience=4,  
    verbose=1,  
    mode='max',  
    cooldown=4,  
    min_lr=0.0000001)
```

02

```
early = tf.keras.callbacks.EarlyStopping(  
    monitor='val_f1_score',  
    mode='max',  
    patience=20)
```

03

```
checkpoint_path = 'save_model/checkpointXception.h5'  
checkpoint = tf.keras.callbacks.ModelCheckpoint(  
    filepath=checkpoint_path,  
    save_weights_only=True,  
    monitor='val_f1_score',  
    mode='max',  
    verbose=1,  
    save_best_only=True)
```

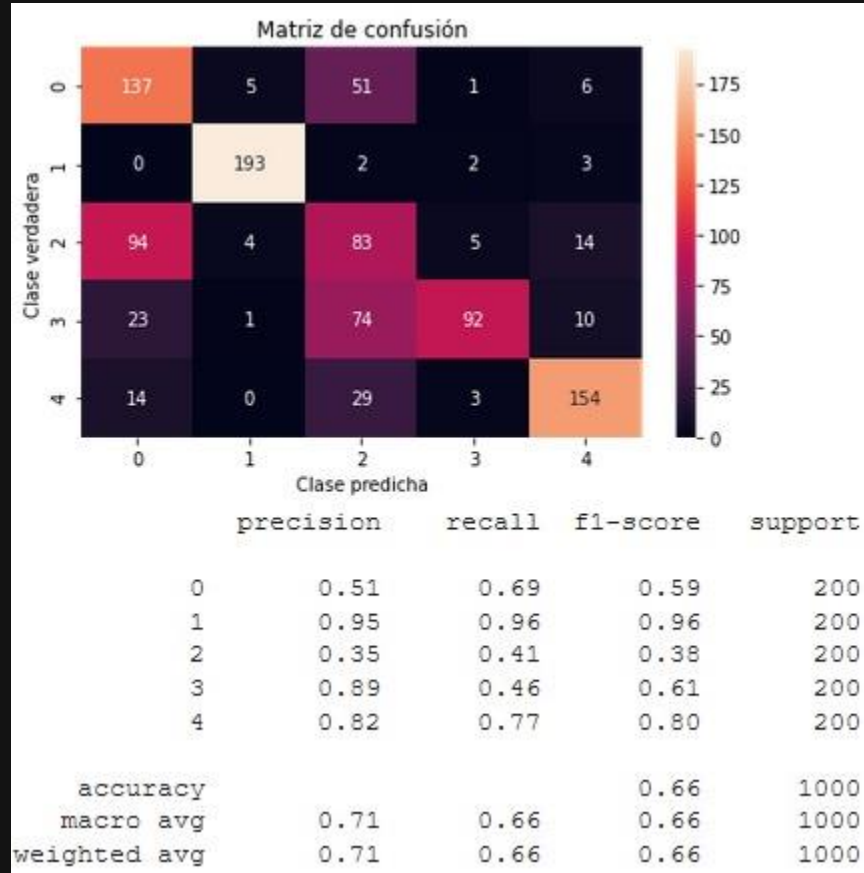

CREACIÓN DEL MODELO

Layer (type)	Output Shape	Param #
exception (Functional)	(None, 4, 4, 2048)	20861480
global_max_pooling2d (Global)	(None, 2048)	0
flatten (Flatten)	(None, 2048)	0
dense (Dense)	(None, 256)	524544
dropout_1 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 256)	65792
dense_2 (Dense)	(None, 5)	1285
Total params: 21,453,101		
Trainable params: 21,398,573		
Non-trainable params: 54,528		
None		
number of layers to train: 160		

CREACIÓN DEL MODELO

```
opt = tf.keras.optimizers.Adam(lr=0.0001)
model_on_Xc1.compile(optimizer=opt,
                      loss='categorical_crossentropy',
                      metrics=['accuracy', tfa.metrics.F1Score(num_classes=5, threshold=0.5, average='weighted')])
history3 = model_on_Xc1.fit(train_generator,
                             epochs=100,
                             validation_data=test_generator,
                             callbacks=callbacks_list)
```

Matriz de confusión

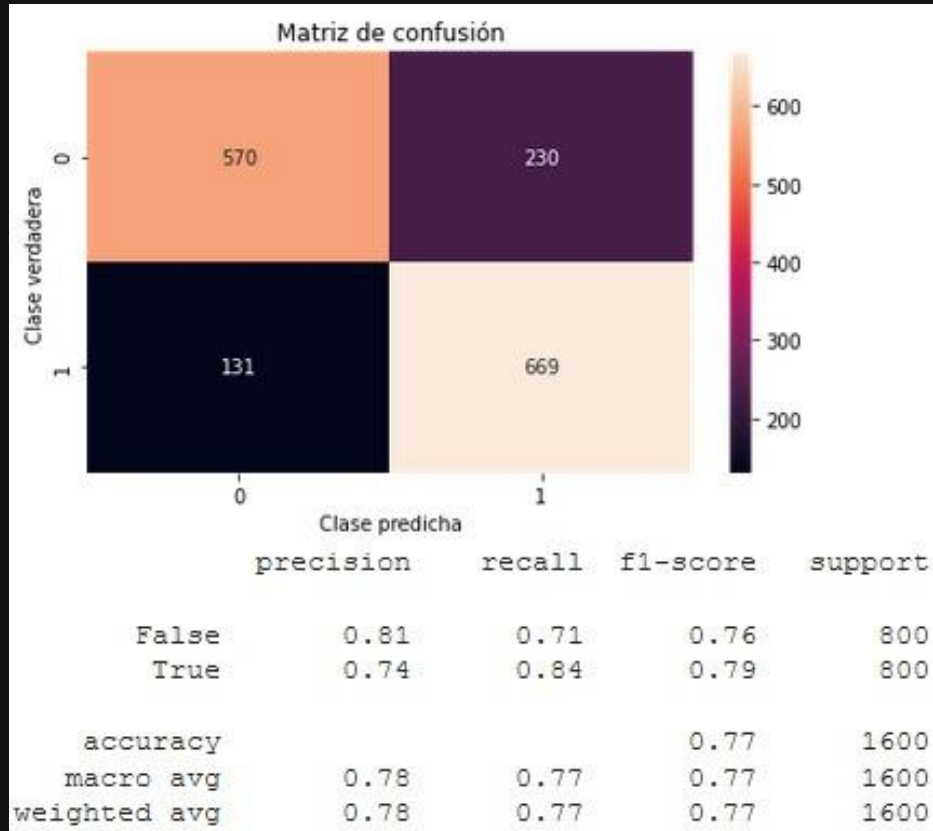




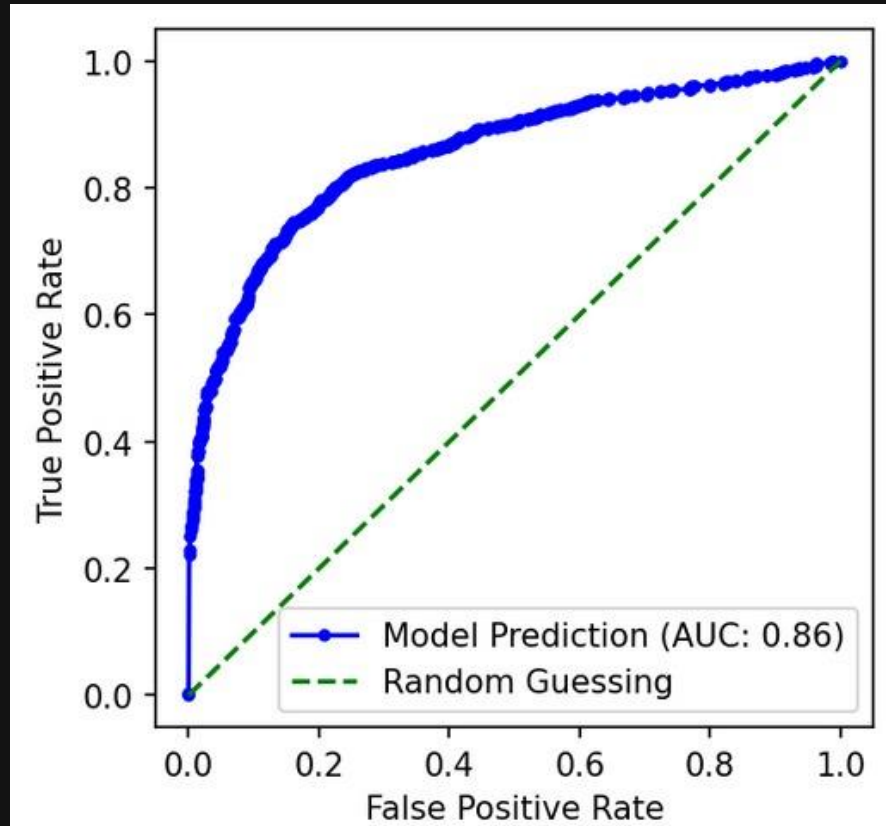
02

Detección de retinopatía diabética

Matriz de confusión



Curva ROC y AUC





Conclusiones

El algoritmo propuesto identifica eficazmente si una persona padece o no de retinopatía diabética, sin embargo, a la hora de predecir la etapa en la cual se encuentra la enfermedad hay un gran conflicto con la clase moderada, ya que, no logra reconocer de forma satisfactoria esta clase y tampoco es precisa su clasificación, provocando a su vez una disminución en la precisión de la clasificación de las muestras sanas.

→ ● Trabajo futuro

Existen diferentes técnicas de procesamiento de imágenes que pueden ser utilizadas y podrían mejorar la tasa de verdaderos positivos y reducir los falsos negativos, además, se recomienda buscar un conjunto de imágenes más amplio, que no presente tanto ruido, fallos de iluminación y desenfoques que dificulten la detección de zonas específicas.