23 de Mayo de 2025

Instituto Politécnico Nacional

“Escuela Superior de Cómputo”

Práctica 2 “Practica radiografías.”

Profesor: Luis Alberto Ibáñez Zamora

Alumno:

Ignacio Cortés Atzin Maxela

email:

ignacio.cortes.atzin.maxela@gmail.com

Grupo: 7BM2

Generar una tabla comparativa con los valores encontrados al entrenar la red neuronal: accuracy, loss, agregar el id de cada proceso, el consumo de memoria ram y el tiempo de ejecución individual de cada lote y el tiempo total de haber entrenado en el clasificador en:

- CPU google colab
- GPU google colab
- CPU de forma local en entorno visual studio code, jupiternote book o algun compilador de python

Adjuntar su colab o notebook

1) CPU Google colab

```
/usr/local/lib/python3.11/dist-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pas
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
Model: "sequential_1"
```

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 64)	640
max_pooling2d (MaxPooling2D)	(None, 74, 74, 64)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 64)	36,928
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 64)	0
conv2d_3 (Conv2D)	(None, 15, 15, 64)	36,928
flatten (Flatten)	(None, 14400)	0
dense (Dense)	(None, 64)	921,664
dense_1 (Dense)	(None, 2)	130

```
Total params: 1,033,218 (3.94 MB)
Trainable params: 1,033,218 (3.94 MB)
Non-trainable params: 0 (0.00 B)
```

```
Epoch 1/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 106s 1s/step - accuracy: 0.5802 - loss: 0.6478 - val_accuracy: 0.8285 - val_loss: 0.3943
Epoch 2/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 102s 1s/step - accuracy: 0.9000 - loss: 0.2471 - val_accuracy: 0.8013 - val_loss: 0.4639
Epoch 3/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 106s 1s/step - accuracy: 0.9362 - loss: 0.1776 - val_accuracy: 0.7869 - val_loss: 0.5645
Epoch 4/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 143s 1s/step - accuracy: 0.9442 - loss: 0.1310 - val_accuracy: 0.7163 - val_loss: 1.2380
Epoch 5/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 141s 1s/step - accuracy: 0.9635 - loss: 0.0861 - val_accuracy: 0.8317 - val_loss: 0.4478
Epoch 6/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 137s 1s/step - accuracy: 0.9687 - loss: 0.0816 - val_accuracy: 0.7917 - val_loss: 1.0835
Epoch 7/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 104s 1s/step - accuracy: 0.9739 - loss: 0.0631 - val_accuracy: 0.8141 - val_loss: 0.5907
Epoch 8/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 145s 1s/step - accuracy: 0.9799 - loss: 0.0519 - val_accuracy: 0.7885 - val_loss: 0.9294
Epoch 9/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 143s 1s/step - accuracy: 0.9844 - loss: 0.0468 - val_accuracy: 0.7740 - val_loss: 1.1272
Epoch 10/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 139s 1s/step - accuracy: 0.9811 - loss: 0.0512 - val_accuracy: 0.8045 - val_loss: 0.9052
Epoch 11/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 139s 1s/step - accuracy: 0.9914 - loss: 0.0256 - val_accuracy: 0.8045 - val_loss: 0.8983
Epoch 12/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 146s 1s/step - accuracy: 0.9917 - loss: 0.0277 - val_accuracy: 0.7853 - val_loss: 1.0038
Epoch 13/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 142s 1s/step - accuracy: 0.9890 - loss: 0.0242 - val_accuracy: 0.7869 - val_loss: 1.2680
Epoch 14/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 138s 1s/step - accuracy: 0.9939 - loss: 0.0180 - val_accuracy: 0.7740 - val_loss: 0.7077
Epoch 15/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 146s 1s/step - accuracy: 0.9857 - loss: 0.0356 - val_accuracy: 0.7917 - val_loss: 1.3878
Epoch 16/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 142s 1s/step - accuracy: 0.9933 - loss: 0.0208 - val_accuracy: 0.8077 - val_loss: 1.0476
Epoch 17/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 138s 1s/step - accuracy: 0.9960 - loss: 0.0108 - val_accuracy: 0.8301 - val_loss: 0.9132
Epoch 18/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 102s 1s/step - accuracy: 0.9920 - loss: 0.0177 - val_accuracy: 0.8205 - val_loss: 1.2271
Epoch 19/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 146s 1s/step - accuracy: 0.9972 - loss: 0.0066 - val_accuracy: 0.8590 - val_loss: 1.0944
Epoch 20/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 141s 1s/step - accuracy: 0.9876 - loss: 0.0305 - val_accuracy: 0.8253 - val_loss: 0.9960
<keras.src.callbacks.history.History at 0x7c15f1cd9ed0>
```

```
[24] test_loss, test_acc = model.evaluate(x_testReshaped, y_test)
     print(test_acc)
```

```
20/20 ━━━━━━━━━━━━━━━━━━━━ 7s 306ms/step - accuracy: 0.6895 - loss: 1.8772
0.8253205418586731
```

```
[25] test_loss, test_acc = model.evaluate(x_trainReshaped, y_train)
```

```
84/84 ━━━━━━━━━━━━━━━━━━━━ 36s 428ms/step - accuracy: 0.9998 - loss: 0.0032
```

Tabla comparativa

	Epoch	Accuracy	Loss	Validation Accuracy	Validation Loss	Process ID	Mean Batch Time (s)	Mean RAM Usage (MB)	Total Training Time (s)
0	1	0.990679	0.025234	0.849359	0.975588	173	1.123732	2238.749061	635.800742
1	2	0.996271	0.011775	0.782051	1.544594	173	1.123732	2238.749061	635.800742
2	3	0.996644	0.007564	0.842949	0.942124	173	1.123732	2238.749061	635.800742
3	4	0.995526	0.012445	0.780449	2.096662	173	1.123732	2238.749061	635.800742
4	5	0.994034	0.017128	0.836538	0.937387	173	1.123732	2238.749061	635.800742

2) GPU Google colab

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 64)	640
max_pooling2d (MaxPooling2D)	(None, 74, 74, 64)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 64)	36,928
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 64)	0
conv2d_3 (Conv2D)	(None, 15, 15, 64)	36,928
flatten (Flatten)	(None, 14400)	0
dense (Dense)	(None, 64)	921,664
dense_1 (Dense)	(None, 2)	130

Total params: 1,033,218 (3.94 MB)
Trainable params: 1,033,218 (3.94 MB)
Non-trainable params: 0 (0.00 B)

```
Epoch 1/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 14s 85ms/step - accuracy: 0.5974 - loss: 0.6234 - val_accuracy: 0.8237 - val_loss: 0.4147
Epoch 2/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 36ms/step - accuracy: 0.9093 - loss: 0.2390 - val_accuracy: 0.8301 - val_loss: 0.4349
Epoch 3/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 35ms/step - accuracy: 0.9499 - loss: 0.1436 - val_accuracy: 0.8141 - val_loss: 0.4463
Epoch 4/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 34ms/step - accuracy: 0.9612 - loss: 0.0995 - val_accuracy: 0.8205 - val_loss: 0.4173
Epoch 5/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 33ms/step - accuracy: 0.9676 - loss: 0.0851 - val_accuracy: 0.7708 - val_loss: 1.1049
Epoch 6/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 34ms/step - accuracy: 0.9742 - loss: 0.0734 - val_accuracy: 0.8237 - val_loss: 0.7116
Epoch 7/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 37ms/step - accuracy: 0.9734 - loss: 0.0749 - val_accuracy: 0.7420 - val_loss: 1.4611
Epoch 8/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 33ms/step - accuracy: 0.9725 - loss: 0.0701 - val_accuracy: 0.8638 - val_loss: 0.5155
Epoch 9/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 34ms/step - accuracy: 0.9818 - loss: 0.0467 - val_accuracy: 0.8045 - val_loss: 0.8809
Epoch 10/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 32ms/step - accuracy: 0.9919 - loss: 0.0277 - val_accuracy: 0.7788 - val_loss: 1.5858
Epoch 11/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 37ms/step - accuracy: 0.9868 - loss: 0.0394 - val_accuracy: 0.7756 - val_loss: 1.2852
Epoch 12/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 33ms/step - accuracy: 0.9888 - loss: 0.0319 - val_accuracy: 0.7885 - val_loss: 1.3723
Epoch 13/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 34ms/step - accuracy: 0.9894 - loss: 0.0358 - val_accuracy: 0.8077 - val_loss: 1.1026
Epoch 14/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 35ms/step - accuracy: 0.9794 - loss: 0.0483 - val_accuracy: 0.7997 - val_loss: 1.3542
Epoch 15/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 36ms/step - accuracy: 0.9939 - loss: 0.0165 - val_accuracy: 0.8189 - val_loss: 0.8438
```

```
Epoch 16/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 33ms/step - accuracy: 0.9906 - loss: 0.0248 - val_accuracy: 0.7917 - val_loss: 1.5509
Epoch 17/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 32ms/step - accuracy: 0.9985 - loss: 0.0060 - val_accuracy: 0.7740 - val_loss: 1.8738
Epoch 18/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 34ms/step - accuracy: 1.0000 - loss: 0.0020 - val_accuracy: 0.7756 - val_loss: 2.5196
Epoch 19/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 5s 36ms/step - accuracy: 0.9999 - loss: 8.0220e-04 - val_accuracy: 0.8397 - val_loss: 1.2790
Epoch 20/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 3s 34ms/step - accuracy: 0.9915 - loss: 0.0286 - val_accuracy: 0.8317 - val_loss: 1.0936
<keras.src.callbacks.history.History at 0x79bf824c2850>
```

```
[17] test_loss, test_acc = model.evaluate(x_testReshaped, y_test)
     print(test_acc)
```

```
20/20 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.7176 - loss: 2.1455
0.8317307829856873
```

```
[18] test_loss, test_acc = model.evaluate(x_trainReshaped, y_train)
```

```
84/84 ━━━━━━━━━━━━━━━━━━━━ 1s 16ms/step - accuracy: 0.9986 - loss: 0.0041
```

Tabla comparativa

	Epoch	Accuracy	Loss	Validation Accuracy	Validation Loss	Process ID	Mean Batch Time (s)	Mean RAM Usage (MB)	Total Training Time (s)
0	1	0.994407	0.018079	0.798077	1.895166	1297	0.035333	2903.154985	25.339713
1	2	0.996644	0.007815	0.786859	2.412797	1297	0.035333	2903.154985	25.339713
2	3	0.989933	0.030102	0.745192	2.020682	1297	0.035333	2903.154985	25.339713
3	4	0.989933	0.032986	0.817308	1.328164	1297	0.035333	2903.154985	25.339713
4	5	0.997763	0.006683	0.772436	1.759777	1297	0.035333	2903.154985	25.339713

3) 3) CPU en Visual Studio

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 64)	640
max_pooling2d (MaxPooling2D)	(None, 74, 74, 64)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 64)	36,928
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 64)	0
conv2d_3 (Conv2D)	(None, 15, 15, 64)	36,928
flatten (Flatten)	(None, 14400)	0
dense (Dense)	(None, 64)	921,664
dense_1 (Dense)	(None, 2)	130

Total params: 1,033,218 (3.94 MB)

Trainable params: 1,033,218 (3.94 MB)

Non-trainable params: 0 (0.00 B)

```
Epoch 1/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 24s 224ms/step - accuracy: 0.6120 - loss: 0.6368 - val_accuracy: 0.8542 - val_loss: 0.3468
Epoch 2/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 18s 214ms/step - accuracy: 0.9129 - loss: 0.2082 - val_accuracy: 0.7500 - val_loss: 0.9488
Epoch 3/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 20s 237ms/step - accuracy: 0.9484 - loss: 0.1302 - val_accuracy: 0.8622 - val_loss: 0.3643
Epoch 4/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 26s 308ms/step - accuracy: 0.9551 - loss: 0.1215 - val_accuracy: 0.8141 - val_loss: 0.5653
Epoch 5/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 18s 218ms/step - accuracy: 0.9620 - loss: 0.1004 - val_accuracy: 0.8333 - val_loss: 0.5006
Epoch 6/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 19s 221ms/step - accuracy: 0.9665 - loss: 0.0810 - val_accuracy: 0.8526 - val_loss: 0.5643
Epoch 7/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 18s 217ms/step - accuracy: 0.9715 - loss: 0.0800 - val_accuracy: 0.7869 - val_loss: 0.7994
Epoch 8/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 19s 231ms/step - accuracy: 0.9672 - loss: 0.0801 - val_accuracy: 0.8221 - val_loss: 0.7795
Epoch 9/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 41s 492ms/step - accuracy: 0.9820 - loss: 0.0464 - val_accuracy: 0.7917 - val_loss: 1.0009
Epoch 10/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 72s 376ms/step - accuracy: 0.9727 - loss: 0.0799 - val_accuracy: 0.7933 - val_loss: 0.7643
Epoch 11/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 43s 513ms/step - accuracy: 0.9869 - loss: 0.0441 - val_accuracy: 0.8253 - val_loss: 0.9528
Epoch 12/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 34s 400ms/step - accuracy: 0.9878 - loss: 0.0390 - val_accuracy: 0.8269 - val_loss: 0.7699
Epoch 13/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 24s 286ms/step - accuracy: 0.9845 - loss: 0.0390 - val_accuracy: 0.7564 - val_loss: 1.3177
Epoch 14/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 25s 292ms/step - accuracy: 0.9934 - loss: 0.0231 - val_accuracy: 0.8558 - val_loss: 0.6557
Epoch 15/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 24s 284ms/step - accuracy: 0.9914 - loss: 0.0207 - val_accuracy: 0.7564 - val_loss: 1.6536
```

```
84/84 ━━━━━━━━━━━━━━━━━━━━ 24s 284ms/step - accuracy: 0.9914 - loss: 0.0207 - val_accuracy: 0.7564 - val_loss: 1.6536
Epoch 16/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 25s 292ms/step - accuracy: 0.9939 - loss: 0.0179 - val_accuracy: 0.7452 - val_loss: 1.4040
Epoch 17/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 25s 297ms/step - accuracy: 0.9895 - loss: 0.0331 - val_accuracy: 0.8029 - val_loss: 1.0153
Epoch 18/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 25s 292ms/step - accuracy: 0.9966 - loss: 0.0142 - val_accuracy: 0.7788 - val_loss: 1.2101
Epoch 19/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 41s 490ms/step - accuracy: 0.9972 - loss: 0.0095 - val_accuracy: 0.8061 - val_loss: 1.3053
Epoch 20/20
84/84 ━━━━━━━━━━━━━━━━━━━━ 38s 449ms/step - accuracy: 0.9984 - loss: 0.0046 - val_accuracy: 0.7965 - val_loss: 1.4126
```

Tabla Comparativas

	Epoch	Accuracy	Loss	Validation Accuracy	Validation Loss	Process ID	Mean Batch Time (s)	Mean RAM Usage (MB)	Total Training Time (s)
0	1	0.992916	0.019146	0.809295	1.041927	31932	0.277408	2153.655292	125.670445
1	2	0.993661	0.018705	0.794872	1.578601	31932	0.277408	2153.655292	125.670445
2	3	0.996644	0.010943	0.801282	1.264274	31932	0.277408	2153.655292	125.670445
3	4	0.991051	0.020572	0.777244	1.624410	31932	0.277408	2153.655292	125.670445
4	5	0.997017	0.007635	0.785256	1.308199	31932	0.277408	2153.655292	125.670445