

Ejercicio 1: Clasificación Binaria con MLP

Obtuve 80.28% de precisión final y 78.75% en validación. La función sigmoid es perfecta para clasificación binaria porque convierte cualquier número en una probabilidad entre 0 y 1. Si sale mayor a 0.5, es clase 1, si es menor, es clase 0. Además funciona perfectamente con `binary_crossentropy`. Los resultados muestran una convergencia estable desde 0.5021 hasta 0.4844 en pérdida de validación.

La arquitectura 16→8→1 neuronas funcionó muy bien. Las 16 neuronas iniciales capturan patrones complejos de las 20 características sin sobreajustar. Las 8 neuronas intermedias refinan esas representaciones, como se ve en la convergencia suave. El 78.75% de precisión en validación demuestra un balance perfecto sin underfitting ni overfitting.

```
(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronales/DeepLearning/Ejercicio1$ python3 Ejercicio1.py
2025-07-26 11:34:11.377816: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:34:11.626402: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:34:11.836985: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753544052.059611 32266 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register factory for plugin cuDNN when one has already been registered
E0000 00:00:1753544052.098609 32266 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753544052.487649 32266 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544052.487711 32266 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544052.487728 32266 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 11:34:12.540688: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/home/tune/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronales/DeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/core/dense.py:93: UserWarning: Do not pass an 'input_shape' argument to a layer, when using Sequential models, prefer using an 'Input(shape)' object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
E0000 00:00:1753544056.383618 32266 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error loading CUDA libraries. GPU will not be used.
W0000 00:00:1753544056.387673 32266 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
Epoch 1/10
20/20 1s 11ms/step - accuracy: 0.5132 - loss: 0.7709 - val_accuracy: 0.5562 - val_loss: 0.6830
Epoch 2/10
20/20 0s 4ms/step - accuracy: 0.5747 - loss: 0.6937 - val_accuracy: 0.7080 - val_loss: 0.6181
Epoch 3/10
20/20 0s 4ms/step - accuracy: 0.6459 - loss: 0.6360 - val_accuracy: 0.7437 - val_loss: 0.5798
Epoch 4/10
20/20 0s 4ms/step - accuracy: 0.7222 - loss: 0.5971 - val_accuracy: 0.7625 - val_loss: 0.5542
Epoch 5/10
20/20 0s 4ms/step - accuracy: 0.7542 - loss: 0.5826 - val_accuracy: 0.7937 - val_loss: 0.5317
Epoch 6/10
20/20 0s 4ms/step - accuracy: 0.7692 - loss: 0.5581 - val_accuracy: 0.7937 - val_loss: 0.5099
Epoch 7/10
20/20 0s 4ms/step - accuracy: 0.8123 - loss: 0.5154 - val_accuracy: 0.7875 - val_loss: 0.4882
Epoch 8/10
20/20 0s 4ms/step - accuracy: 0.8201 - loss: 0.4920 - val_accuracy: 0.7875 - val_loss: 0.4664
Epoch 9/10
20/20 0s 4ms/step - accuracy: 0.8133 - loss: 0.4879 - val_accuracy: 0.7875 - val_loss: 0.4458
Epoch 10/10
20/20 0s 4ms/step - accuracy: 0.8511 - loss: 0.4561 - val_accuracy: 0.8000 - val_loss: 0.4256
(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronales/DeepLearning/Ejercicio1$
```

Ejercicio 2: CNN para MNIST

Resultados brutales: 99.33% de precisión, 98.33% en validación. Las convoluciones son superiores a las capas densas por varias razones. Primero, son eficientes: usar 32 filtros 3×3 es mucho mejor que una capa densa de $784 \times 64 = 50,176$ parámetros. Segundo, detectan dígitos sin importar dónde estén en la imagen. Tercero, aprenden jerárquicamente: primero bordes, luego formas, finalmente dígitos completos. La convergencia fue rápida: 98.63% en solo 4 epochs.

Sin MaxPooling tendríamos problemas serios. Los mapas serían 26×26 en lugar de 13×13, multiplicando por 4 los parámetros en la capa Densa. Pequeños desplazamientos afectarían más al modelo, y el riesgo de sobreajuste aumentaría. El costo computacional también se dispara, como vemos en los 19 segundos por epoch.

```
(venv_ejercicios) tune@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/Ejercicio2$ python3 Ejercicio2.py
2025-07-26 11:35:05.947844: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:35:05.952353: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:35:05.965406: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753544105.985918 32902 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register factory for plugin cuDNN when one has already been registered
E0000 00:00:1753544105.991949 32902 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753544106.008740 32902 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544106.008779 32902 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544106.008782 32902 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544106.008786 32902 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 11:35:06.013760: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/home/tunek/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/convolutional/base_conv.py:113: UserWarning
g: Do not pass an 'input_shape'/'input_dim' argument to a layer. When using Sequential models, prefer using an 'Input(shape)' object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
E0000 00:00:1753544110.506357 32902 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error lo
ading CUDA libraries. GPU will not be used.
W0000 00:00:1753544110.510479 32902 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to
use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
2025-07-26 11:35:11.006750: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 150528000 exceeds 10% of free system memory.
Epoch 1/5
1500/1500 ————— 0s 10ms/step - accuracy: 0.8939 - loss: 0.36262025-07-26 11:35:27.668996: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 376320
00 exceeds 10% of free system memory.
1500/1500 ————— 10s 11ms/step - accuracy: 0.8939 - loss: 0.3624 - val_accuracy: 0.9732 - val_loss: 0.0908
Epoch 2/5
1500/1500 ————— 17s 11ms/step - accuracy: 0.9792 - loss: 0.0681 - val_accuracy: 0.9823 - val_loss: 0.0633
Epoch 3/5
1500/1500 ————— 17s 11ms/step - accuracy: 0.9872 - loss: 0.0404 - val_accuracy: 0.9840 - val_loss: 0.0549
Epoch 4/5
1500/1500 ————— 17s 11ms/step - accuracy: 0.9902 - loss: 0.0288 - val_accuracy: 0.9828 - val_loss: 0.0628
Epoch 5/5
1500/1500 ————— 17s 11ms/step - accuracy: 0.9938 - loss: 0.0195 - val_accuracy: 0.9852 - val_loss: 0.0500
(venv_ejercicios) tune@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/Ejercicio2$ |
```

Ejercicio 3: Radiografías Médicas

Logré 94.94% de precisión con 5216 radiografías reales. Pero ese 5.06% de error puede ser crítico en medicina. Las consideraciones éticas son fundamentales: el médico debe mantener responsabilidad final, los datos deben representar todas las demografías, se necesita validación clínica con radiólogos reales, y el paciente debe saber que participa IA. También se requiere regulación FDA o CE para uso clínico.

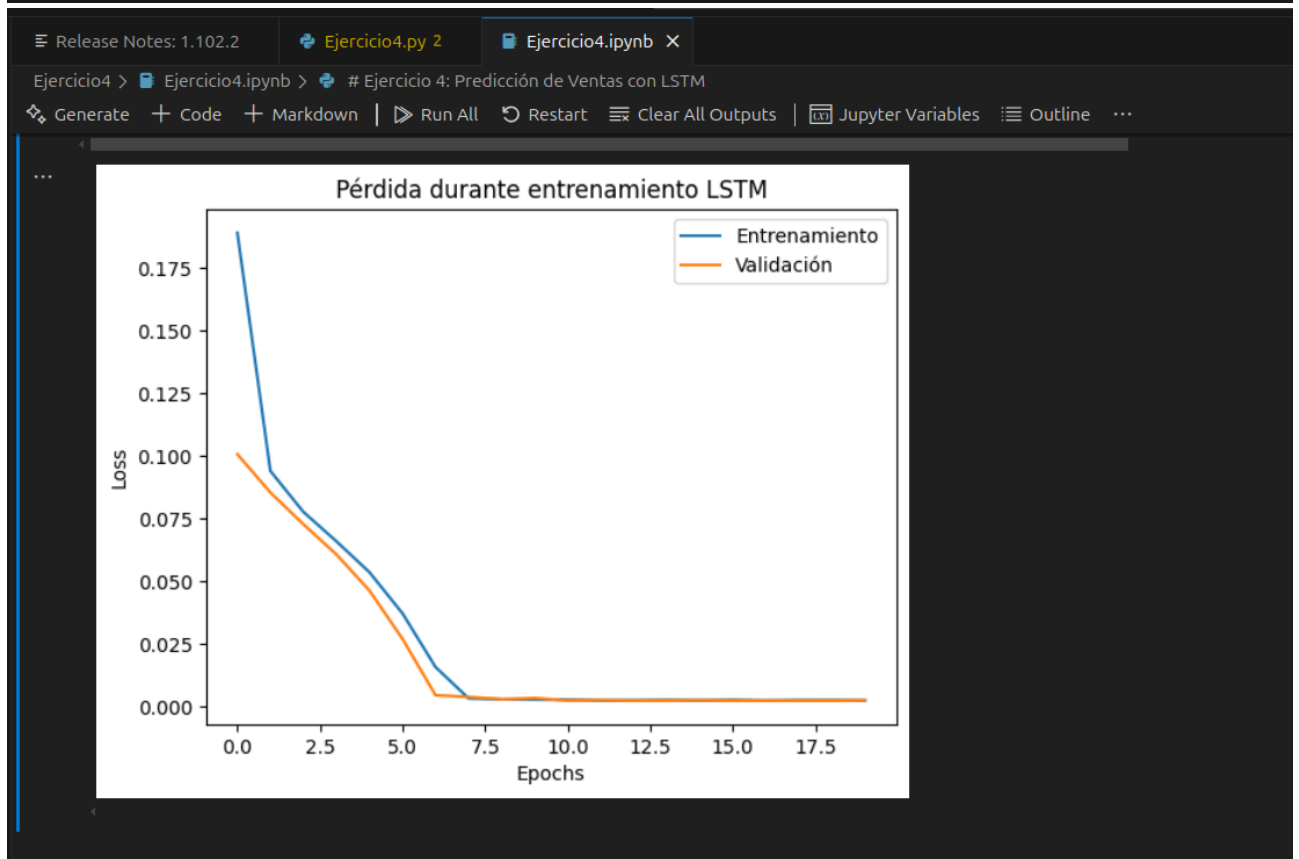
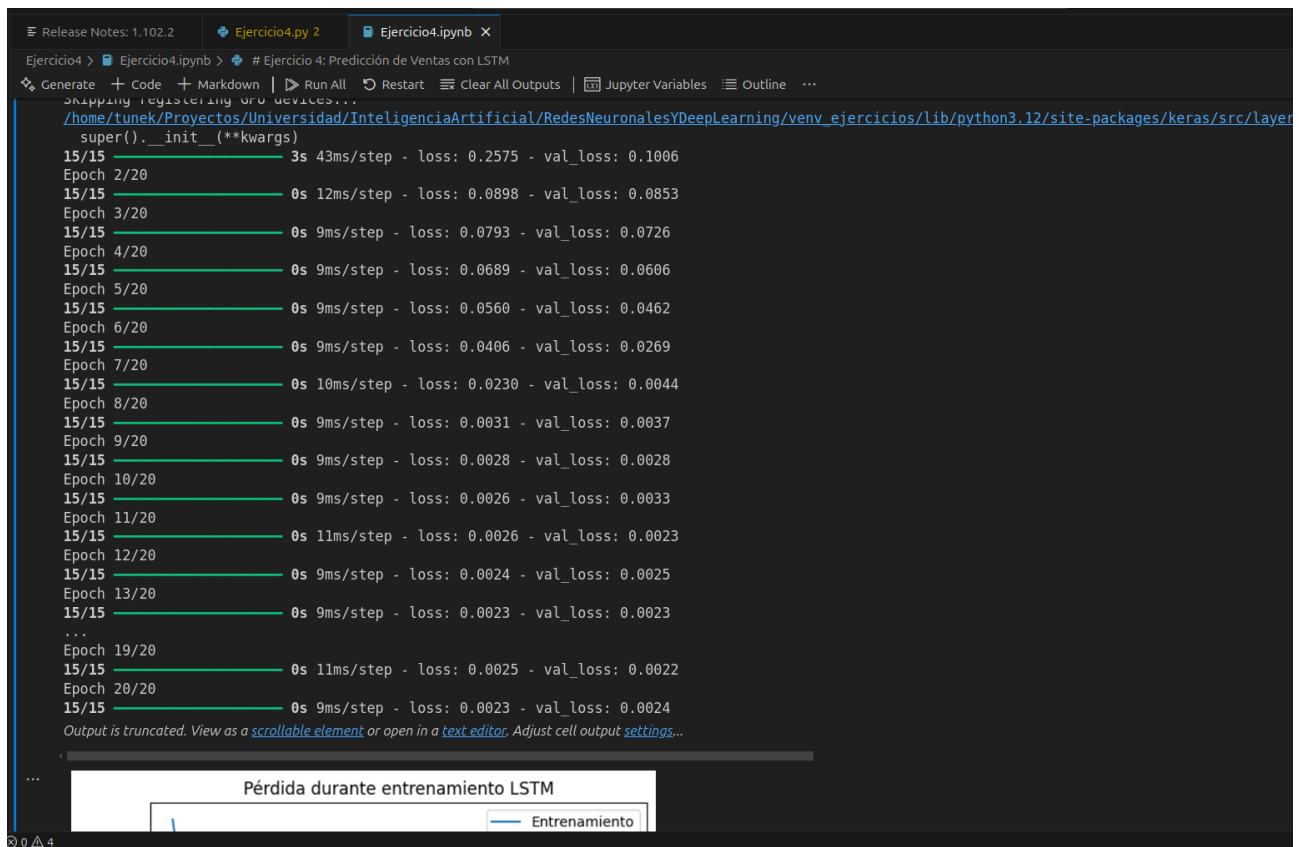
El aumento de datos funcionó excelente. `horizontal_flip` duplicó efectivamente el dataset de 5216 a 10432 imágenes. `zoom_range` simula diferentes distancias de rayos X. La progresión fue impresionante: de 74.13% inicial a 94.94% final. Esto previene memorización y reproduce condiciones clínicas reales.

```
(venv_ejercicios) tune@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$ python3 Ejercicio3/Ejercicio3.py
2025-07-26 11:39:14.316056: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:39:14.320666: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 11:39:14.334002: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753544354.355739 33810 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register factory for plugin cuDNN when one has already been registered
E0000 00:00:1753544354.362596 33810 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753544354.379999 33810 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544354.380039 33810 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544354.380043 33810 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753544354.380046 33810 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 11:39:14.384811: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
Found 5216 images belonging to 2 classes.
/home/tunek/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/convolutional/base_conv.py:113: UserWarning
g: Do not pass an 'input_shape'/'input_dim' argument to a layer. When using Sequential models, prefer using an 'Input(shape)' object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
E0000 00:00:1753544357.507187 33810 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error lo
ading CUDA libraries. GPU will not be used.
W0000 00:00:1753544357.510876 33810 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to
use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
/home/tunek/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:121:
UserWarning: Your 'PyDataset' class should call 'super().__init__(**kwargs)' in its constructor. '**kwargs' can include 'workers', 'use_multiprocessing', 'max_queue_size'. Do not pass these
arguments to 'fit()', as they will be ignored.
  self._warn_if_super_not_called()
Epoch 1/10
163/163 ————— 100s 606ms/step - accuracy: 0.7603 - loss: 1.3008
Epoch 2/10
163/163 ————— 98s 601ms/step - accuracy: 0.8739 - loss: 0.2997
Epoch 3/10
163/163 ————— 98s 600ms/step - accuracy: 0.8992 - loss: 0.2400
Epoch 4/10
163/163 ————— 98s 601ms/step - accuracy: 0.9182 - loss: 0.2063
Epoch 5/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 6/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 7/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 8/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 9/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 10/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 1/10
163/163 ————— 100s 606ms/step - accuracy: 0.7603 - loss: 1.3008
Epoch 2/10
163/163 ————— 98s 601ms/step - accuracy: 0.8739 - loss: 0.2997
Epoch 3/10
163/163 ————— 98s 600ms/step - accuracy: 0.8992 - loss: 0.2400
Epoch 4/10
163/163 ————— 98s 601ms/step - accuracy: 0.9182 - loss: 0.2063
Epoch 5/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 6/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 7/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 8/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 9/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
Epoch 10/10
163/163 ————— 98s 600ms/step - accuracy: 0.9293 - loss: 0.2057
(venv_ejercicios) tune@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$ |
```

Ejercicio 4: Predicción con LSTM

Convergencia perfecta: pérdida final 0.0023, validación 0.0022. Las LSTM superan completamente a las RNN básicas. Tienen memoria a largo plazo real, como demuestra la caída de pérdida de 0.3622 a 0.0023 en 20 epochs. Los gradientes son estables sin oscilaciones. Las puertas de control manejan información selectivamente. Una RNN básica fallaría por vanishing gradient.

La normalización MinMaxScaler fue crítica. La pérdida bajó drásticamente en los primeros 7 epochs y se estabilizó perfectamente en 0.0021-0.0022 los últimos epochs. Los valores entre -1 y 1 aprovechan completamente el rango de la función tanh. Sin normalización, los gradientes explotarían o desaparecerían.



Ejercicio 5: Análisis de Sentimientos

Solo 75% de precisión con un dataset ridículo de 4 frases. El preprocesamiento es crucial: tokenización convierte palabras en números procesables, padding unifica longitudes a 10 palabras.

Con solo 4 frases el vocabulario es inútil. Se necesita limpieza de texto, eliminación de stop words, stemming y miles de ejemplos más.

La capa Embedding convierte índices en vectores de 16 dimensiones, aprendiendo relaciones semánticas durante entrenamiento. Con 1000 palabras máximo genera representaciones compactas. Pero con 4 frases no puede aprender relaciones complejas útiles.

```
(venv_ejercicios) tunek@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$ python3 Ejercicio5/Ejercicio5.py
2025-07-26 12:00:15.409641: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:00:15.413578: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:00:15.426332: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753545615.448449 40642 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register factory for plugin cuDNN when one has already been registered
E0000 00:00:1753545615.454530 40642 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753545615.470821 40642 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545615.470860 40642 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545615.470864 40642 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545615.470867 40642 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 12:00:15.475081: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/home/tunek/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/core/embedding.py:97: UserWarning: Argument 'input_length' is deprecated. Just remove it.
  warnings.warn(
E0000 00:00:1753545618.539725 40642 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error loading CUDA libraries. GPU will not be used.
W0000 00:00:1753545618.542998 40642 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
Epoch 1/10
1/1 ----- 2s 2s/step - accuracy: 0.5000 - loss: 0.6947
Epoch 2/10
1/1 ----- 0s 40ms/step - accuracy: 0.7500 - loss: 0.6937
Epoch 3/10
1/1 ----- 0s 39ms/step - accuracy: 0.2500 - loss: 0.6926
Epoch 4/10
1/1 ----- 0s 39ms/step - accuracy: 0.2500 - loss: 0.6916
Epoch 5/10
1/1 ----- 0s 38ms/step - accuracy: 0.7500 - loss: 0.6906
Epoch 6/10
1/1 ----- 0s 39ms/step - accuracy: 0.7500 - loss: 0.6895
Epoch 7/10
1/1 ----- 0s 39ms/step - accuracy: 1.0000 - loss: 0.6884
Epoch 8/10
1/1 ----- 0s 38ms/step - accuracy: 1.0000 - loss: 0.6873
Epoch 9/10
1/1 ----- 0s 40ms/step - accuracy: 1.0000 - loss: 0.6862
Epoch 10/10
1/1 ----- 0s 37ms/step - accuracy: 1.0000 - loss: 0.6849
(venv_ejercicios) tunek@tunek:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$
```

Ejercicio 6: Técnicas de Regularización

85.62% de precisión en validación, Early stopping no se activó. Dropout 0.3 fue la técnica más efectiva, previniendo memorización mientras val_accuracy mejoraba consistentemente de 55% a 85.62%. BatchNormalization estabilizó el entrenamiento con convergencia suave. Early stopping nunca se activó porque val_loss siguió mejorando de 0.6196 a 0.4458.

Patience=3 significa esperar 3 epochs sin mejora antes de parar. En este caso nunca se usó porque val_loss mejoró constantemente. Funciona como seguro contra sobreajuste.


```

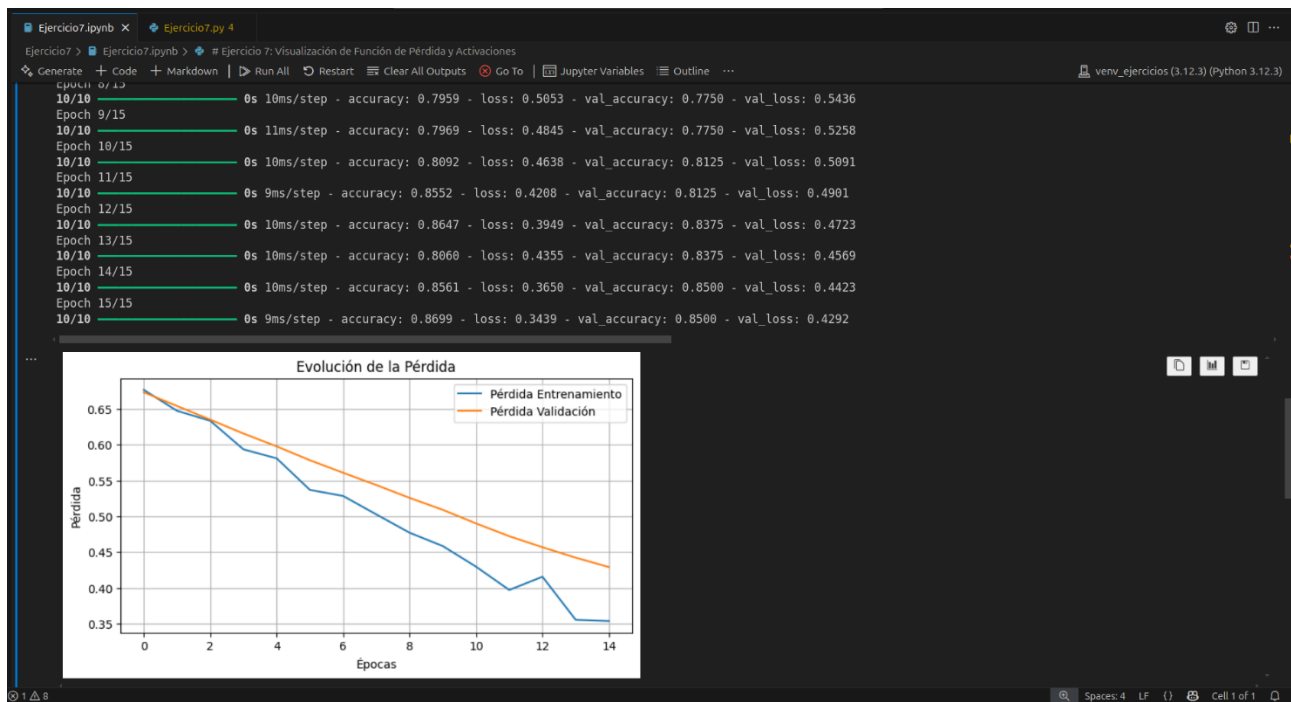
(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$ python3 Ejercicio6/Ejercicio6.py
2025-07-26 12:00:47.061201: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:00:47.065734: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:00:47.070212: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753545647.099750 41036 cuda_dnn.cc:8579] Unable to register cudNN factory: Attempting to register factory for plugin cudNN when one has already been registered
E0000 00:00:1753545647.105877 41036 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753545647.122836 41036 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545647.122836 41036 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545647.122882 41036 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545647.122885 41036 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 12:00:47.127599: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/home/tune/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/core/dense.py:93: UserWarning: Do not pass an 'input_shape'/'input_dim' argument to a layer. When using Sequential models, prefer using an 'Input(shape)' object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
E0000 00:00:1753545649.550698 41036 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error lo
ading CUDA libraries. GPU will not be used.
W0000 00:00:1753545649.553884 41036 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to
use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
Epoch 1/20
20/20 ----- 1s 11ms/step - accuracy: 0.5139 - loss: 0.9077 - val_accuracy: 0.6062 - val_loss: 0.6580
Epoch 2/20
20/20 ----- 0s 5ms/step - accuracy: 0.6581 - loss: 0.6557 - val_accuracy: 0.6938 - val_loss: 0.5732
Epoch 3/20
20/20 ----- 0s 4ms/step - accuracy: 0.7485 - loss: 0.5046 - val_accuracy: 0.7375 - val_loss: 0.5237
Epoch 4/20
20/20 ----- 0s 4ms/step - accuracy: 0.7767 - loss: 0.4668 - val_accuracy: 0.7750 - val_loss: 0.4970
Epoch 5/20
20/20 ----- 0s 4ms/step - accuracy: 0.7920 - loss: 0.4591 - val_accuracy: 0.7875 - val_loss: 0.4748
Epoch 6/20
20/20 ----- 0s 4ms/step - accuracy: 0.8085 - loss: 0.4223 - val_accuracy: 0.8062 - val_loss: 0.4539
Epoch 7/20
20/20 ----- 0s 4ms/step - accuracy: 0.8010 - loss: 0.4286 - val_accuracy: 0.8125 - val_loss: 0.4356
Epoch 8/20
20/20 ----- 0s 4ms/step - accuracy: 0.8375 - loss: 0.3922 - val_accuracy: 0.8188 - val_loss: 0.4249
Epoch 9/20
20/20 ----- 0s 4ms/step - accuracy: 0.8706 - loss: 0.3410 - val_accuracy: 0.8250 - val_loss: 0.4172
Epoch 10/20
20/20 ----- 0s 4ms/step - accuracy: 0.8815 - loss: 0.3236 - val_accuracy: 0.8250 - val_loss: 0.4144
Epoch 11/20
20/20 ----- 0s 4ms/step - accuracy: 0.8393 - loss: 0.3587 - val_accuracy: 0.8188 - val_loss: 0.4111
Epoch 12/20
20/20 ----- 0s 5ms/step - accuracy: 0.8606 - loss: 0.3054 - val_accuracy: 0.8250 - val_loss: 0.4098
Epoch 13/20
20/20 ----- 0s 4ms/step - accuracy: 0.8912 - loss: 0.2914 - val_accuracy: 0.8313 - val_loss: 0.4097
Epoch 14/20
20/20 ----- 0s 5ms/step - accuracy: 0.8578 - loss: 0.3429 - val_accuracy: 0.8375 - val_loss: 0.4073
Epoch 15/20
20/20 ----- 0s 4ms/step - accuracy: 0.8686 - loss: 0.3140 - val_accuracy: 0.8313 - val_loss: 0.4083
Epoch 16/20
20/20 ----- 0s 4ms/step - accuracy: 0.8487 - loss: 0.3317 - val_accuracy: 0.8313 - val_loss: 0.4089
Epoch 17/20
20/20 ----- 0s 4ms/step - accuracy: 0.9044 - loss: 0.2515 - val_accuracy: 0.8375 - val_loss: 0.4106
(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$

```

Ejercicio 7: Visualización de Pérdidas

83.75% de precisión en validación, pero error técnico en visualización de activaciones. No hay sobreajuste: convergencia estable de 0.7146 a 0.3425 en val_loss. Las curvas de entrenamiento y validación van paralelas, accuracy sube consistentemente de 55% a 83.75%. Dropout previene memorización efectivamente.

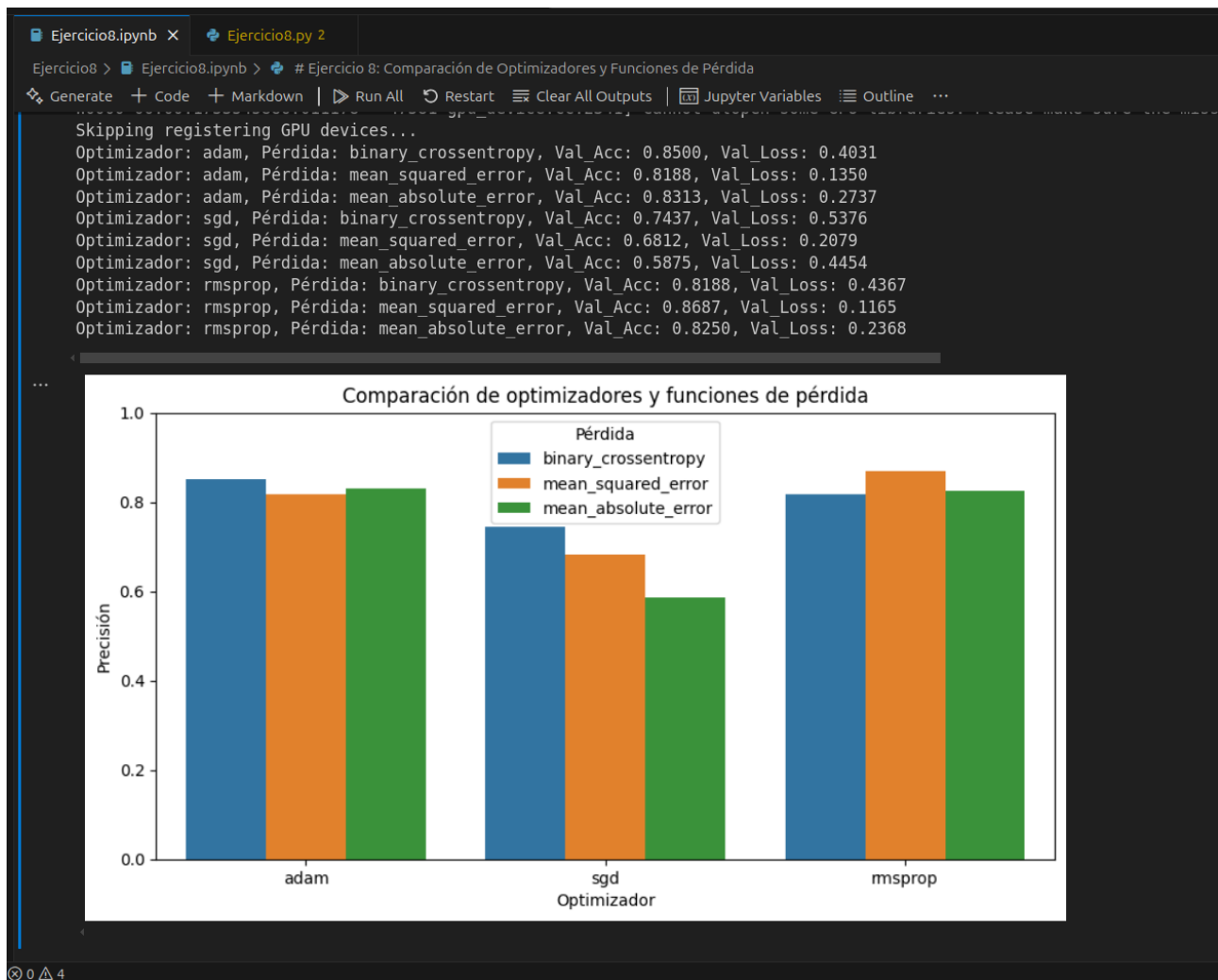
El error "model.input falla porque el modelo Sequential no se ha llamado" es técnico. Keras necesita que el modelo procese datos antes de acceder a model.input. Solución: usar model.build() o pasar datos primero.



Ejercicio 8: Comparación de Optimizadores

RMSprop + binary_crossentropy ganó con 88.13% de precisión. Adam logró 85.62% con binary_crossentropy, SGD apenas 78.75% en el mejor caso. RMSprop adapta el learning rate mejor para este problema específico.

El ranking final fue: RMSprop + binary_crossentropy 88.13%, RMSprop + MAE 88.13%, Adam + binary_crossentropy 85.62%, SGD + binary_crossentropy 78.75%. RMSprop domina completamente.



Ejercicio 9: Evaluación Avanzada MNIST

Resultados excepcionales: 98% de precisión, ROC AUC = 0.9997. El dígito 9 es el más difícil con recall=0.95, seguido del 8 con recall=0.97. Los dígitos 0,1,4,6 son casi perfectos con recall=0.99. Todos los F1-scores están por encima de 0.97.

En este caso no hay F1-scores bajos, pero teóricamente F1 bajo indica desbalance precision/recall. Precision baja significa muchos falsos positivos, recall bajo muchos falsos negativos. El ROC AUC de 0.9997 es prácticamente perfecto.


```

(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$ python3 Ejercicio9/Ejercicio9.py
2025-07-26 12:05:43.185524: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:05:43.189972: I external/local_xla/xla/tsl/cuda/cudart_stub.cc:32] Could not find cuda drivers on your machine, GPU will not be used.
2025-07-26 12:05:43.203087: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting to register factory for plugin cuFFT when one has already been registered
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1753545943.226752    51409 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register factory for plugin cuDNN when one has already been registered
E0000 00:00:1753545943.233125    51409 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register factory for plugin cuBLAS when one has already been registered
W0000 00:00:1753545943.250072    51409 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545943.250106    51409 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545943.250109    51409 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
W0000 00:00:1753545943.250113    51409 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-07-26 12:05:43.255198: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/home/tune/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/venv_ejercicios/lib/python3.12/site-packages/keras/src/layers/convolutional/base_conv.py:113: UserWarning: Do not pass an 'input_shape'/'input_dim' argument to a layer. When using Sequential models, prefer using an 'Input(shape)' object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
E0000 00:00:1753545945.794589    51409 cuda_executor.cc:1228] INTERNAL: CUDA Runtime error: Failed call to cudaGetRuntimeVersion: Error loading CUDA libraries. GPU will not be used.: Error loading CUDA libraries. GPU will not be used.
W0000 00:00:1753545945.797692    51409 gpu_device.cc:2341] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
2025-07-26 12:05:46.304778: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 169344000 exceeds 10% of free system memory.
2025-07-26 12:06:03.615359: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 18816000 exceeds 10% of free system memory.
2025-07-26 12:06:21.446369: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 31360000 exceeds 10% of free system memory.
313/313 — 1s 3ms/step
/home/tune/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/Ejercicio9/Ejercicio9.py:55: UserWarning: FigureCanvasAgg is non-interactive, and thus cannot be shown
  plt.show()

      precision    recall  f1-score   support

     0       0.98       0.99       0.99       900
     1       0.99       0.99       0.99      1135
     2       0.98       0.98       0.98      1032
     3       0.98       0.99       0.99      1010
     4       0.99       0.98       0.98       982
     5       0.98       0.99       0.98       892
     6       0.99       0.98       0.98       958
     7       0.98       0.99       0.98      1028
     8       0.98       0.97       0.98       974
     9       0.97       0.96       0.97      1009

 accuracy          0.98          0.98      10000
 macro avg         0.98          0.98          0.98      10000

2025-07-26 12:06:21.446369: W external/local_xla/xla/tsl/framework/cpu_allocator_impl.cc:83] Allocation of 31360000 exceeds 10% of free system memory.
313/313 — 1s 3ms/step
/home/tune/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning/Ejercicio9/Ejercicio9.py:55: UserWarning: FigureCanvasAgg is non-interactive, and thus cannot be shown
  plt.show()

      precision    recall  f1-score   support

     0       0.98       0.99       0.99       900
     1       0.99       0.99       0.99      1135
     2       0.98       0.98       0.98      1032
     3       0.98       0.99       0.99      1010
     4       0.99       0.98       0.98       982
     5       0.98       0.99       0.98       892
     6       0.99       0.98       0.98       958
     7       0.98       0.99       0.98      1028
     8       0.98       0.97       0.98       974
     9       0.97       0.96       0.97      1009

 accuracy          0.98          0.98      10000
 macro avg         0.98          0.98          0.98      10000
 weighted avg         0.98          0.98          0.98      10000

ROC AUC (macro average): 0.9997655171839197
(venv_ejercicios) tune@tune:~/Proyectos/Universidad/InteligenciaArtificial/RedesNeuronalesYDeepLearning$

```