

6g-ai: Manual de uso

Redes neuronales convolucionales para la clasificación de imágenes



23 DE FEBRERO DE 2026
UNIVERSIDAD DE EXTREMADURA
Pedro Antonio Salguero López

1. Introducción

Esta parte del servicio permite entrenar y probar redes neuronales convolucionales (CNN) para clasificación de imágenes. El servicio está diseñado para ser flexible, permitiendo al usuario elegir una red preentrenada y personalizar su arquitectura agregando hasta tres capas adicionales. Además, se pueden ajustar parámetros de entrenamiento como número de épocas, proporción de entrenamiento y test, y aplicar early stopping para evitar sobreajuste.

2. Requisitos del Dataset

Para que la aplicación funcione correctamente, el dataset debe cumplir los siguientes requisitos:

1. **Formato de imágenes permitido:** JPG, JPEG o PNG.
2. **Estructura de carpetas:**
 - La carpeta raíz debe contener subcarpetas, cada una representando una clase.
 - Cada subcarpeta debe contener únicamente imágenes pertenecientes a esa clase.

Ejemplo de estructura correcta:

```
dataset/
├── Gatos/
│   ├── gato1.jpg
│   ├── gato2.png
│   └── ...
├── Perros/
│   ├── perro1.jpg
│   ├── perro2.jpeg
│   └── ...
└── Pájaros/
    ├── pajaro1.jpg
    ├── pajaro2.png
    └── ...
```

3. Interfaz de Usuario

La aplicación consta de varias secciones:

3.1 Selección de Red Preentrenada

- El usuario puede elegir entre varias redes preentrenadas (por ejemplo, DenseNet121).
- Estas redes ya cuentan con pesos previamente entrenados en grandes datasets como ImageNet, lo que acelera el proceso de entrenamiento.

3.2 Personalización de la Red

- El usuario puede agregar hasta **tres capas adicionales** al final de la red preentrenada.
- Tipos de capas permitidas:
 - **Dense (Fully Connected)**: para clasificación final.
 - **Dropout**: para regularización.
 - **BatchNormalization**: para estabilizar y acelerar el entrenamiento.

Nota: El orden y la configuración de estas capas puede afectar el desempeño de la red.

3.3 Configuración de Entrenamiento

- **Número de épocas:** Define cuántas veces la red recorrerá todo el dataset de entrenamiento.
- **Porcentaje de entrenamiento y test:** Permite definir la proporción de imágenes utilizadas para entrenamiento y validación. Ejemplo: 80% entrenamiento, 20% test.
- **Early stopping:** Puede activarse para detener el entrenamiento si la pérdida de validación no mejora después de un número determinado de épocas, evitando sobreajuste.

4. Proceso de Entrenamiento

1. Seleccionar la red preentrenada deseada.
2. Añadir hasta tres capas adicionales según se requiera.
3. Ajustar parámetros de entrenamiento: épocas, proporción de entrenamiento/test y early stopping.
4. Subir el dataset en la estructura correcta.
5. Iniciar el entrenamiento.

5. Resultados

Una vez finalizado el entrenamiento, la aplicación proporciona diferentes formas de analizar el desempeño del modelo:

5.1 Evolución de Precisión y Pérdida

- Se muestran **gráficas de la precisión (accuracy)** y la **pérdida (loss)** tanto del entrenamiento como del test a lo largo de las épocas.
- Esto permite visualizar:
 - Si la red está aprendiendo correctamente.
 - Posibles problemas de **sobreajuste (overfitting)** o **subajuste (underfitting)**.
- Las curvas típicas incluyen:
 - **Precisión de entrenamiento vs precisión de validación**
 - **Pérdida de entrenamiento vs pérdida de validación**

5.2 Matriz de Confusión

- La matriz de confusión se genera utilizando los datos del **test**.
- Permite ver con detalle:
 - Cuántos casos fueron correctamente clasificados.
 - Cuántos casos fueron confundidos entre clases.
- Esto ayuda a identificar **clases problemáticas** que necesitan más imágenes o ajustes en la red.

5.3 Predicciones de Cada Caso

- Se listan las **predicciones individuales** para cada imagen del test, mostrando:
 - Imagen de prueba
 - Clase real
 - Clase predicha
 - Probabilidad asociada a la predicción
- Esto facilita un análisis más detallado de errores específicos del modelo.

5.4 Descarga de la Red Entrenada

- Una vez satisfecho con el desempeño, es posible **descargar la red entrenada** en un archivo `.h5`.
- Este archivo contiene:
 - La arquitectura de la red
 - Los pesos entrenados
 - La configuración de entrenamiento
- Permite **reutilizar la red** en otros proyectos o aplicaciones sin necesidad de volver a entrenar desde cero.

6. Evaluación y Prueba

- Una vez entrenada la red, se puede probar con imágenes nuevas no vistas por el modelo.
- La aplicación permite cargar imágenes individuales o un pequeño conjunto para evaluar el desempeño de la red entrenada.
- Los resultados incluyen:
 - Clase predicha
 - Probabilidad de predicción
 - Comparación con la etiqueta real (si aplica)

7. Buenas Prácticas

- Asegúrate de que cada clase tenga un número suficiente de imágenes para entrenar la red.
- Evita clases con solo unas pocas imágenes, ya que pueden generar un modelo sesgado.
- Normaliza las imágenes antes de subirlas (mismo tamaño y orientación).
- Prueba con diferentes combinaciones de capas adicionales para mejorar el desempeño.

8. Resumen del Flujo de Uso

1. Preparar dataset con estructura de carpetas por clase.
2. Seleccionar red preentrenada.
3. Añadir hasta tres capas personalizadas(opcional).
4. Configurar parámetros de entrenamiento (épocas, proporción train/test, early stopping).
5. Subir dataset a la aplicación.
6. Iniciar entrenamiento y monitorizar progreso.
7. Probar red entrenada con nuevas imágenes.
8. Analizar resultados y revisar configuración si es necesario.

9. Interfaz de Usuario (Versión Visual)

9.1 Pantalla Inicial

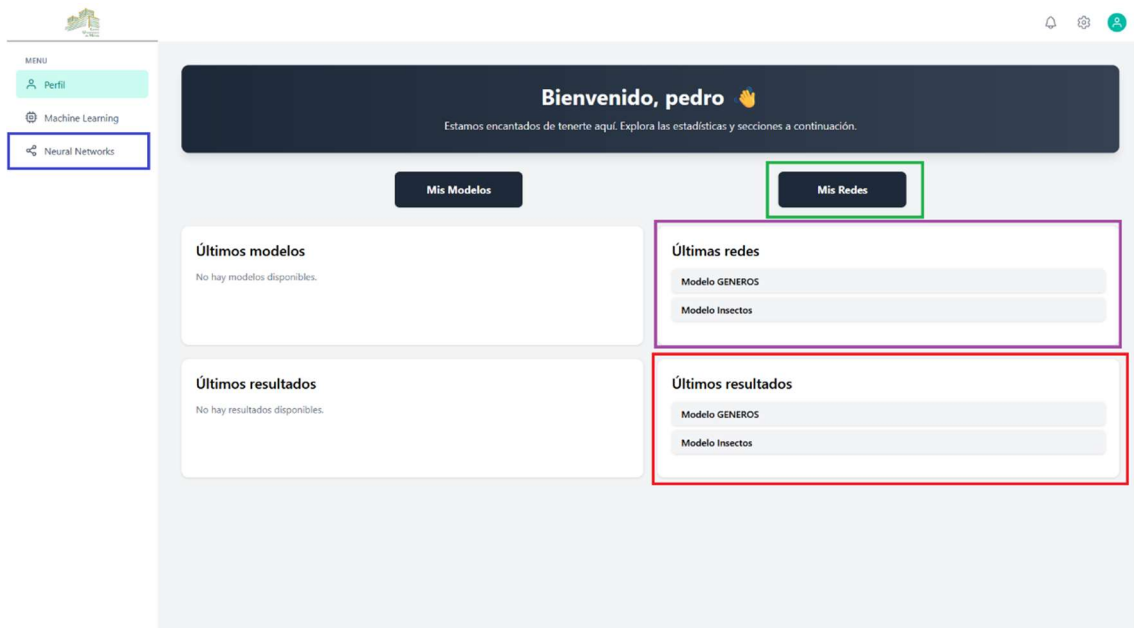


Imagen 1. Pantalla inicial

En la pantalla principal se muestran cuatro secciones destacadas, identificadas con recuadros de distintos colores, cada una de las cuales permite acceder a diferentes funcionalidades del sistema:

9.1.1. Opción en el sidebar (recuadro azul)

Ubicada en el menú lateral, esta opción permite acceder a una pantalla específica del sistema.

Al seleccionarla, el usuario es redirigido a la pantalla correspondiente.

9.1.2. Acceso a la lista de redes (recuadro verde)

Este apartado dirige al usuario a la pantalla donde se muestra la **lista completa de redes** disponibles en el sistema.

Al hacer clic, se accede a dicha sección (ver **Imagen 2: Listado de redes creadas**).

9.1.3. Redes creadas (recuadro violeta)

En esta sección se muestra un listado de las **redes creadas**.

Cada red dispone de un hipervínculo que permite acceder a su información detallada (ver **Imagen 4: Detalle de una red creada**).

9.1.4. Resultados (recuadro rojo)

Aquí se presenta un listado de los **resultados generados**. Cada resultado incluye un hipervínculo que redirige la pantalla de resultados. (ver **Imagen 5: Listado resultados**).

9.2 Lista de Redes Existentes

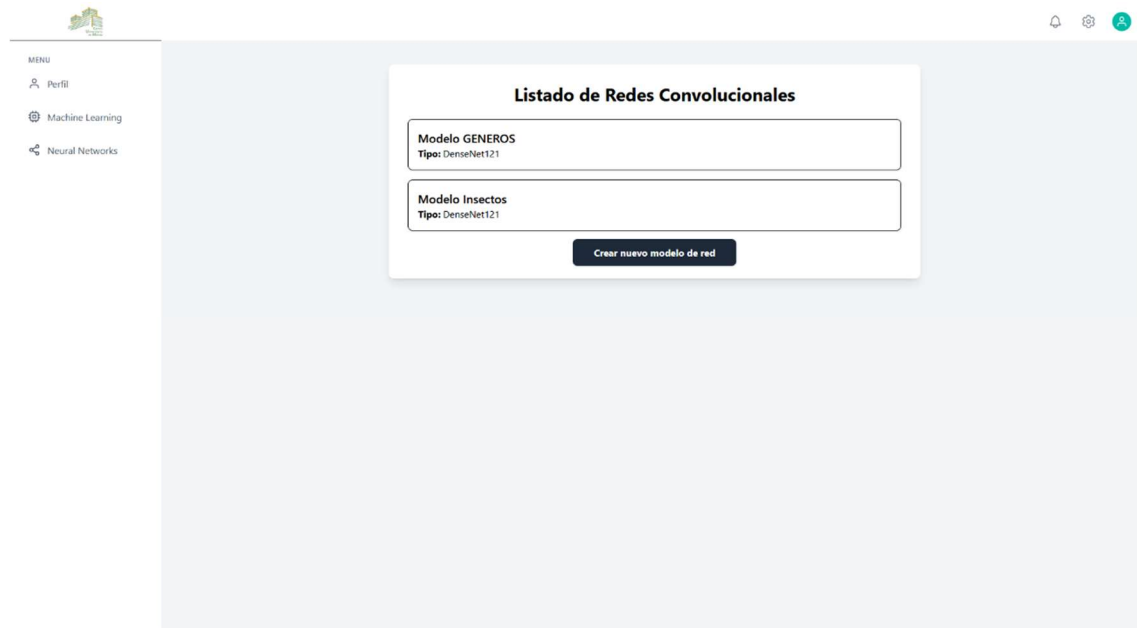
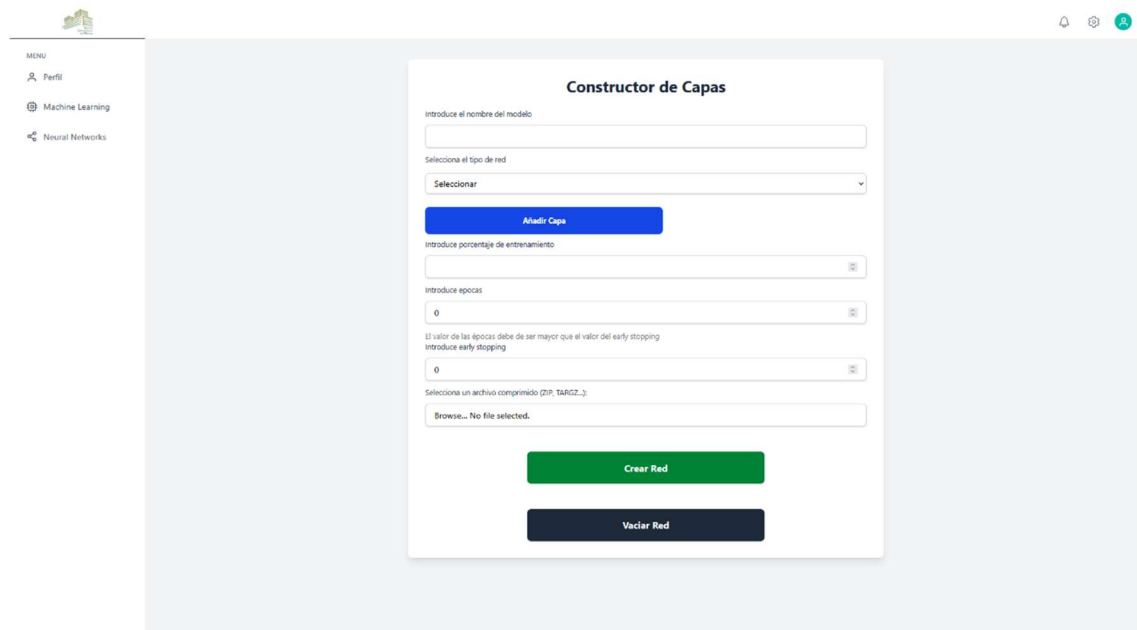


Imagen 2. Listado redes creadas

La **lista de redes** es la pantalla en la que se muestran todas las redes creadas en el sistema hasta el momento. Cada elemento del listado representa una red y permite acceder a su información correspondiente.

Además, esta pantalla incluye un botón de acción que dirige al usuario a la pantalla de **creación de una nueva red**, desde donde es posible registrar y configurar una red adicional.

9.3 Creación de una Nueva Red



The image shows a web interface for creating a neural network. On the left is a sidebar menu with 'MENU', 'Perfil', 'Machine Learning', and 'Neural Networks'. The main content area is titled 'Constructor de Capas' and contains a form with the following fields and buttons:

- Text input: 'Introduce el nombre del modelo'
- Dropdown menu: 'Selecciona el tipo de red' with a 'Seleccionar' button
- Blue button: 'Añadir Capa'
- Text input: 'Introduce porcentaje de entrenamiento'
- Text input: 'Introduce épocas' (value: 0)
- Text input: 'Introduce early stopping' (value: 0) with a note: 'El valor de las épocas debe de ser mayor que el valor del early stopping'
- File upload: 'Selecciona un archivo comprimido (ZIP, TAR.GZ...)' with a 'Browse... No file selected.' button
- Green button: 'Crear Red'
- Dark blue button: 'Vaciar Red'

Imagen 3. Formulario para crear red

La **creación de nueva red** es la pantalla destinada al registro de una red en el sistema.

En ella, el usuario debe completar un formulario con las estipulaciones y parámetros previamente definidos. Una vez rellenados los campos requeridos y confirmada la acción, la nueva red queda registrada y pasa a formar parte de la lista de redes disponibles.

9.4 Vista de Red Creada

Detalles de red

Nombre del modelo
Modelo Insectos

Tipo de red
DenseNet121

Tipo de capa
Dense Layer
Unidades
128

Tipo de capa
Dense Layer
Unidades
64

Tipo de capa
Dropout
Porcentaje
0.5

Porcentaje de entrenamiento
0.8

Épocas
100

Early stopping
15

Editar Eliminar Modelo Entrenar

Imagen 4. Detalles de una red creada

La **Vista de red creada** es la pantalla en la que se muestra la información detallada de una red previamente registrada. En ella se presentan los parámetros configurados y el estado actual de la red.

Además, esta vista incluye las siguientes acciones:

- **Editar:** Permite modificar los parámetros configurables de la red.
- **Entrenar:** Inicia el proceso de entrenamiento con la configuración actualmente guardada.
- **Eliminar:** Permite borrar la red del sistema.

Es importante tener en cuenta las siguientes consideraciones:

- **No se podrá eliminar una red mientras esté en proceso de entrenamiento.**
- Si se realiza una **edición durante el entrenamiento**, los parámetros modificados **no afectarán al entrenamiento en curso**. Los cambios se aplicarán únicamente en ejecuciones o entrenamientos posteriores.

9.8 Resultados

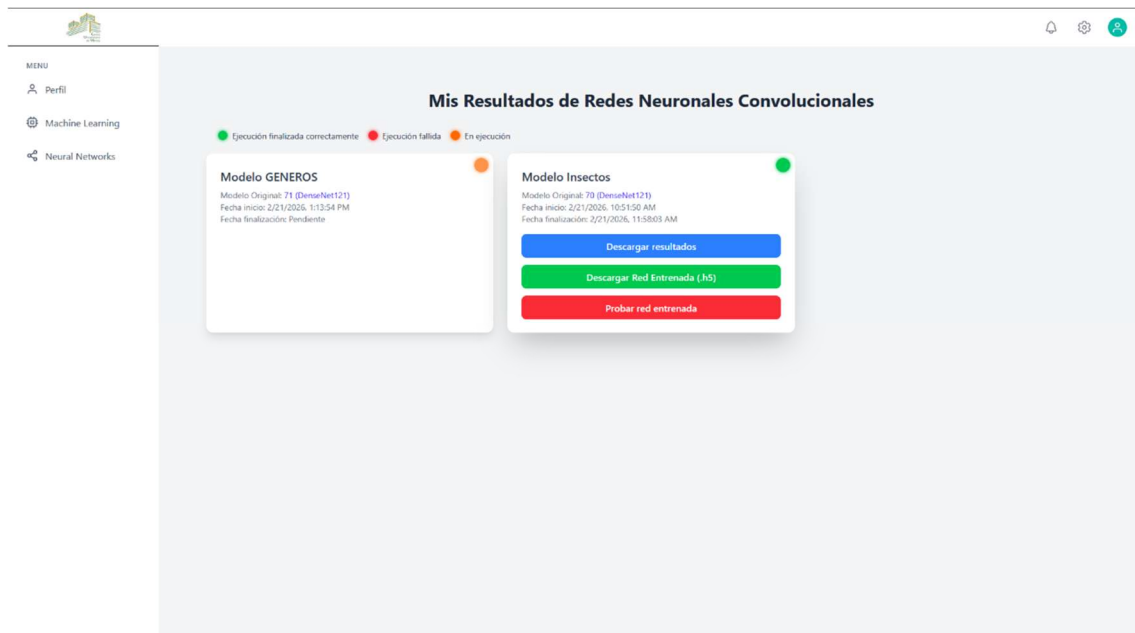


Imagen 5. Listado resultados

La pantalla de **Resultados** muestra el listado de ejecuciones asociadas a las redes. En ella se pueden visualizar tanto resultados ya finalizados como ejecuciones que aún se encuentran en curso.

Cada resultado incluye un indicador visual ubicado en la esquina superior derecha, cuyo color refleja el estado de la ejecución según la siguiente leyenda:

- **Verde:** Ejecución completada con éxito.
- **Rojo:** Ejecución finalizada con error.
- **Naranja:** Ejecución en curso.

De este modo, el usuario puede identificar de forma rápida y visual el estado de cada ejecución sin necesidad de acceder al detalle individual de cada resultado.

Importante: Si una red ya ha generado resultados y se vuelve a ejecutar, los resultados anteriores **se sobrescriben**. Para poder comparar ejecuciones o conservar información histórica, se recomienda **descargar los resultados antes de ejecutar nuevamente**.

10. Resultados del Entrenamiento

En esta sección se presentan los resultados obtenidos tras el entrenamiento de la red neuronal. A continuación, se describen los elementos mostrados:

10.1 Precisión de la red

Se muestra el valor final de precisión (accuracy) obtenido por el modelo tras el entrenamiento.

Este valor indica el porcentaje de aciertos sobre el total de muestras evaluadas.

- **Precisión en entrenamiento:** rendimiento del modelo con los datos usados para entrenar.
- **Precisión en validación/test:** rendimiento del modelo con datos no vistos previamente.

10.2 Evolución de la precisión (entrenamiento y test)

Se presenta una gráfica que muestra la evolución de la precisión a lo largo de las épocas de entrenamiento.

- Eje X: número de épocas.
- Eje Y: valor de precisión.
- Curva de entrenamiento.
- Curva de validación/test.

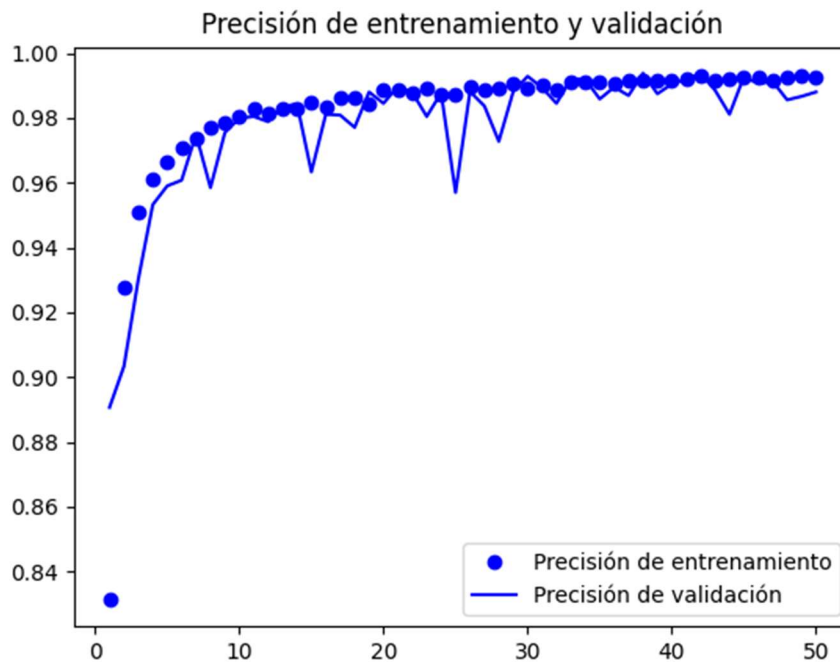


Imagen 6. Evolución de la precisión en entrenamiento y test

Esta gráfica permite analizar:

- Si el modelo mejora progresivamente.
- Si existe sobreajuste (cuando la precisión de entrenamiento aumenta, pero la de test disminuye o se estanca).

10.3 Evolución de la pérdida (entrenamiento y test)

Se muestra una gráfica con la evolución de la función de pérdida (loss) durante el entrenamiento.

- Eje X: número de épocas.
- Eje Y: valor de la pérdida.
- Curva de entrenamiento.
- Curva de validación/test.

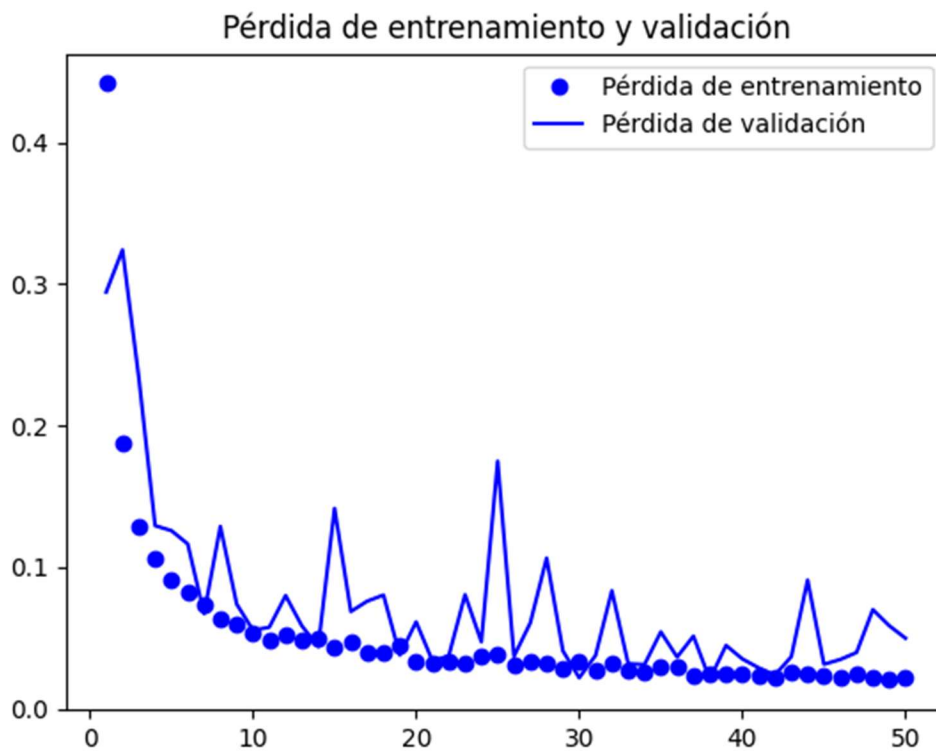


Imagen 7. Evolución de la pérdida para entrenamiento y test

La pérdida indica el error del modelo. Valores más bajos representan mejor ajuste.

10.4 Matriz de confusión

Se incluye la matriz de confusión generada a partir de los datos de test.

La matriz permite visualizar:

- Verdaderos positivos.
- Verdaderos negativos.
- Falsos positivos.
- Falsos negativos.

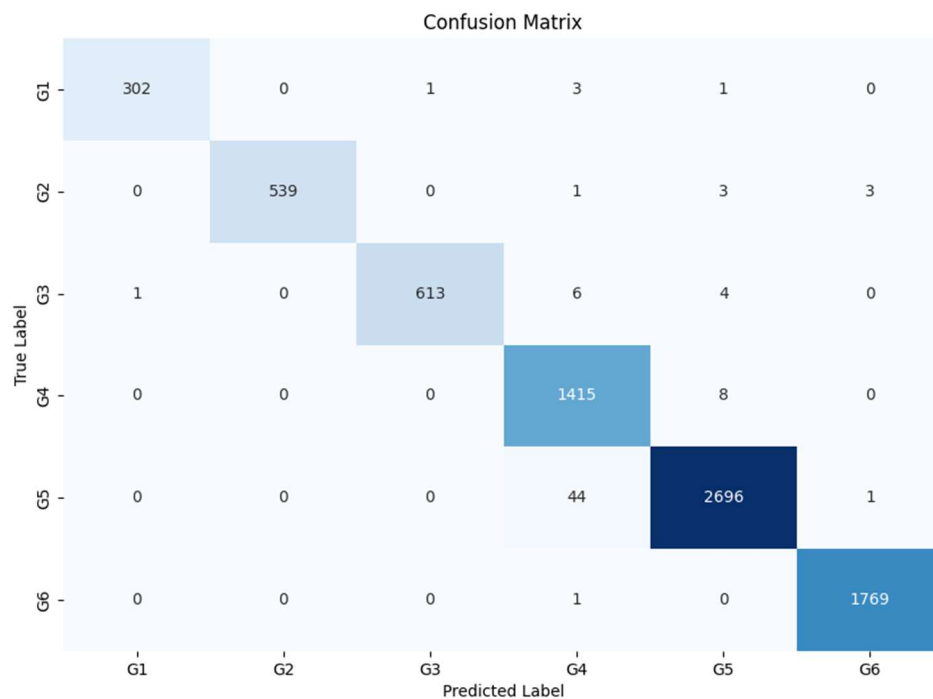


Imagen 8. Matriz de confusión

Es útil para analizar en qué clases el modelo comete más errores.

10.5 Predicciones sobre el conjunto de test

Se muestra el resultado de las predicciones realizadas por el modelo sobre cada muestra del conjunto de test.

Para cada caso se indica:

- Valor esperado (etiqueta real).
- Valor predicho por la red.
- Indicación de acierto o error.

Esto permite realizar un análisis detallado del comportamiento del modelo en casos individuales.

10.6 Modelo entrenado (.h5)

Se genera un archivo con el modelo entrenado en formato .h5.

Este archivo:

- Contiene la arquitectura del modelo.
- Incluye los pesos aprendidos durante el entrenamiento.
- Puede cargarse posteriormente sin necesidad de reentrenar la red.