



**UNIVERSIDAD  
DE ANTIOQUIA**

Facultad de Ingeniería

## **ENTREGA 2**

## **PROYECTO**

Modelo de clasificación para la detección de pacientes con neumonía mediante análisis de imágenes de Rayos X de tórax

## **ASIGNATURA**

Fundamentos de Deep Learning

## **PROGRAMA ACADÉMICO**

Doctorado en Ingeniería Electrónica y Computación

## **NOMBRE DE ESTUDIANTE**

Andrés Felipe Ramírez-Barrera

[andres.ramirez34@udea.edu.co](mailto:andres.ramirez34@udea.edu.co)

CC. 71.267.247

**Universidad de Antioquia  
Facultad de Ingeniería  
2025-2**

## **I. Resumen y Objetivo**

Este documento describe la estructura estándar de un cuaderno (Notebook) de aprendizaje profundo diseñado para la detección de neumonía. El proyecto se centra en la clasificación binaria a partir de imágenes de radiografía de tórax (X-Ray) obtenidas del conjunto de datos de Chest X-Ray Pneumonia de Kaggle. La ejecución se realizó utilizando 2 GPUs Tesla T4. La estructura garantiza claridad, reproducibilidad y destaca las técnicas específicas utilizadas.

**Objetivo Central:** Entrenar y evaluar Redes Neuronales Convolucionales (CNNs), utilizando Aprendizaje por Transferencia, para la clasificación binaria (Normal vs. Neumonía) en radiografías de tórax.

## **II. Estructura del Cuaderno Sección por Sección**

### **Sección 1: Configuración, Adquisición de Datos y Entorno**

**Propósito:** Establecer el entorno de ejecución, definir los parámetros globales y cargar los conjuntos de datos.

**Importación de Librerías:** Importar los paquetes necesarios (TensorFlow/Keras, PyTorch, NumPy, Pandas, Scikit-learn, Matplotlib).

**Variables de Configuración:** Definir de manera centralizada todos los hiperparámetros, rutas de archivos y configuraciones del modelo:

IMAGE\_SIZE (ej. 224x224 o 299x299, basado en el modelo pre-entrenado).

BATCH\_SIZE (ej. 32 o 64).

EPOCHS (ej. 10 a 50).

LEARNING\_RATE (ej. 0.001 o 0.0001).

MODEL\_BACKBONE (ej. VGG-16, ResNet-50 o InceptionV3).

DATA\_PATHS (directorios raíz para los datos de Radiografía de Tórax de Kaggle).

**Carga y División de Datos:** Cargar y verificar el conjunto de datos de Radiografía de Tórax:

**Conjunto de Datos:** Chest X-Ray Pneumonia (2 clases: Normal y Neumonía).

**División:** 4173 imágenes para entrenamiento, 1043 para validación y 624 para pruebas.

**Cálculo de Pesos de Clase (Class Weights):** Calcular y definir los pesos de clase para compensar el desequilibrio en el conjunto de datos de entrenamiento. Esta matriz de pesos se usa en la función de pérdida.

## **Sección 2: Preprocesamiento y Aumento de Datos**

**Propósito:** Preparar la imagen para la CNN, aplicando las transformaciones necesarias para el modelo base.

**Pipeline de Preprocesamiento:**

**Preprocesamiento específico de la arquitectura:** Utilizar la función de preprocesamiento propia de la arquitectura elegida (ej. `preprocess_input` de ResNet) para garantizar que los datos cumplan con los requisitos del modelo pre-entrenado.

**Redimensionamiento:** Redimensionar todas las imágenes al `IMAGE_SIZE` definido.

**Normalización:** Reescalar los valores de los píxeles (ej. al rango  $[0, 1]$  o la normalización requerida por el modelo base).

**Aumento de Datos (Augmentation):** Definir y aplicar transformaciones aleatorias a los datos de entrenamiento para aumentar la diversidad y evitar el sobreajuste.

**División del Conjunto de Datos:** Confirmación de la división: Entrenamiento, Validación y Prueba.

## **Sección 3: Definición y Compilación del Modelo**

**Propósito:** Definir la arquitectura, incorporar técnicas de regularización y compilar el modelo.

**Carga del Modelo Pre-entrenado (Extractor de Características):** Cargar el modelo base elegido, pre-entrenado en ImageNet, y congelar sus pesos.

**Cabecal de Clasificación Personalizado (Classification Head):** Adjuntar nuevas capas entrenables:

**Estructura:** Agrupación Global Promedio -> Capa Densa (con Regularización L2) -> Batch Normalization -> Capa de Salida.

**Capa de Salida:** 2 salidas con activación Softmax (equivalente a clasificación binaria).

**Compilación del Modelo:** Compilar el modelo definiendo:

**Optimizador:** (ej. Adam).

**Función de Pérdida:** Categorical Cross-Entropy Loss (pérdida de entropía cruzada categorica), ya que solo hay dos clases. (de nuevo, equivalente a la versión binaria)

**Métricas:** loss y accuracy.

#### **Sección 4: Ejecuciones de Entrenamiento y Evaluación**

**Objetivo:** Entrenar el modelo con callbacks de control y evaluar su rendimiento.

**Definición de Callbacks:** Establecer mecanismos de control para la estabilidad y eficiencia del entrenamiento:

**Checkpoint:** Guardar los mejores pesos del modelo (basado en la pérdida de validación).

**Early Stopping:** Detener el entrenamiento si la pérdida de validación no mejora después de un número predefinido de épocas (paciencia).

##### **Ejecución 4.1: Entrenamiento Principal (Iteraciones):**

Entrenar el modelo para el número máximo de épocas definido, aplicando los pesos de clase (Class Weights).

La ejecución se detendrá automáticamente gracias al Early Stopping o al finalizar todas las épocas.

##### **Evaluación (Conjunto de Prueba):**

Cargar la mejor versión del modelo (gracias al Checkpoint) y generar predicciones en el conjunto de Prueba reservado (624 imágenes).

Calcular la métrica accuracy y el valor final de la loss.

Analizar la Matriz de Confusión para comprender los errores de clasificación (Falsos Positivos/Negativos).

### **III. Resultados y Análisis Comparativo de las Iteraciones**

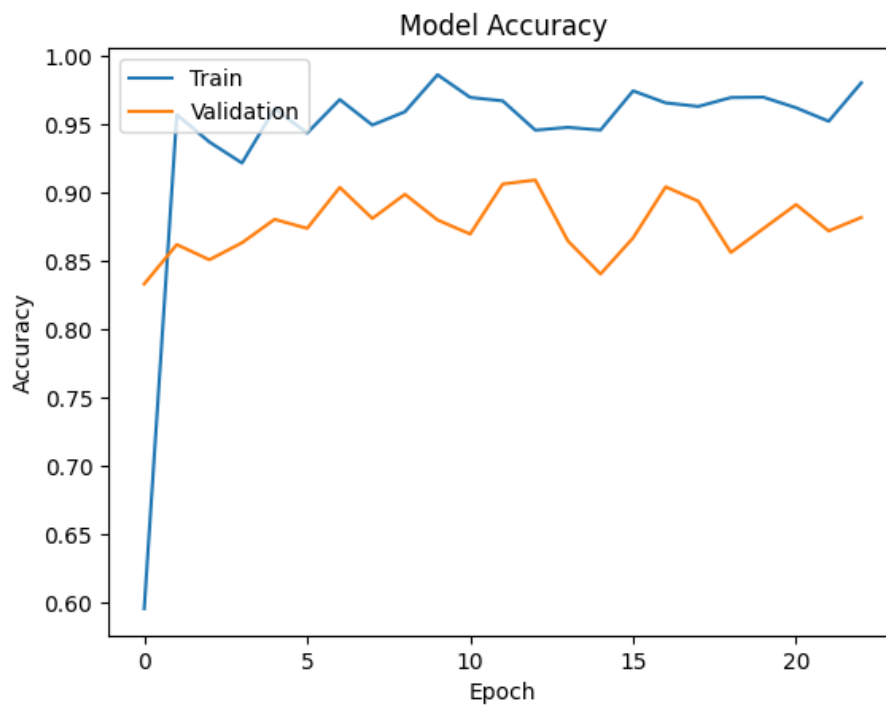
Esta sección recopila los resultados de las diferentes ejecuciones del modelo para permitir una comparación directa de la efectividad de las arquitecturas y los hiperparámetros probados.

**Tabla de Resumen de Experimentos (Ejecuciones):**

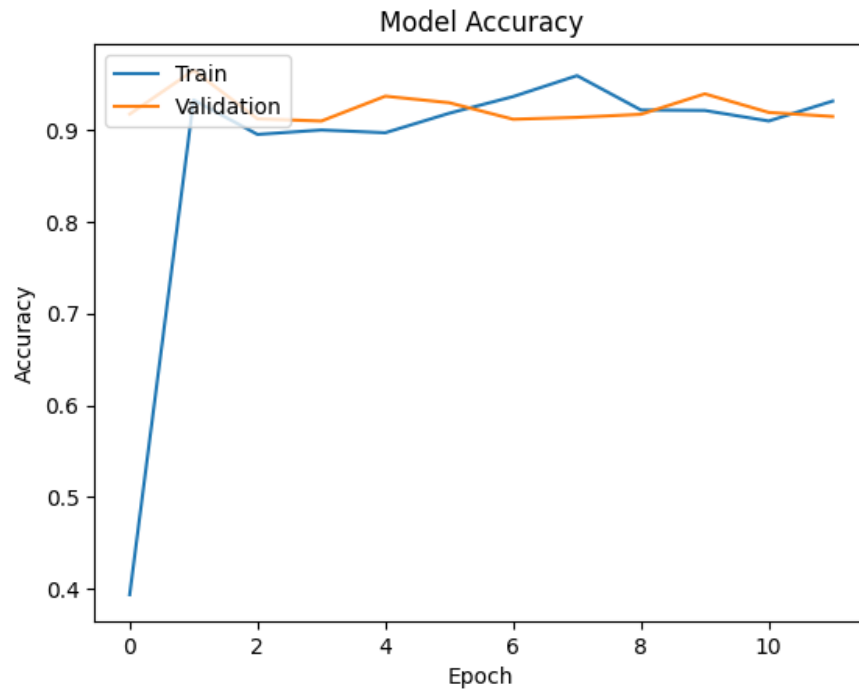
Modelo	Validation Accuracy	Test Accuracy	Test Loss
InceptionV3	94.14%	79.81%	0.5028
VGG19	91.04%	84.01%	0.5791
EfficientNetB4	96.5%	78.17%	0.5403
ResNet50	98.08%	90.77%	0.2251
EfficientNetB4 - L2	94.49%	84.96%	1.2863
ResNet50 - L2	95.04%	92.34%	0.3553

De esta corrida de experimentos se observó sobre entrenamiento (observar imagenes más abajo) en modelos como EfficientNetB4 y ResNet50, por lo que se implementaron mecanismos de regularización y se corrieron nuevamente. ResNet50 con regularización resultó como la configuración con mejor rendimiento al ser el que mejor generaliza los datos.

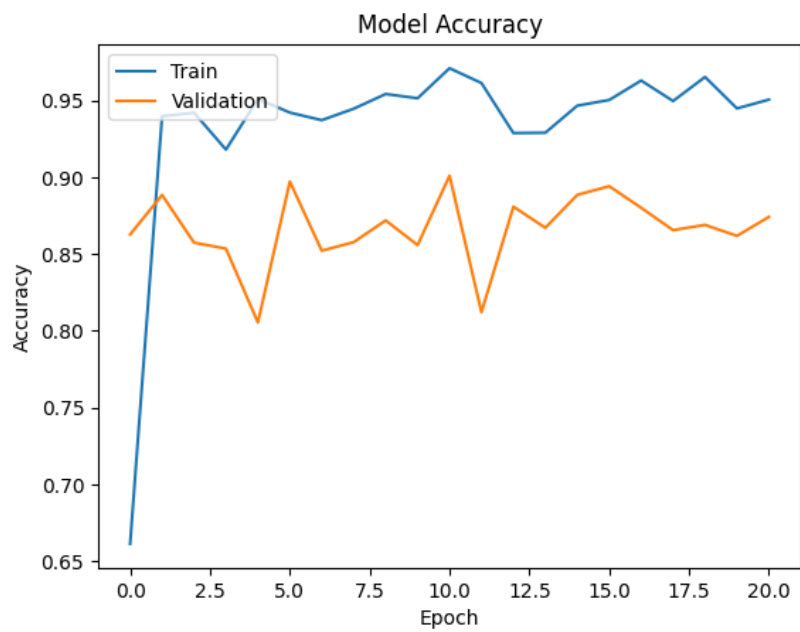
**ResNet50**



**EfficientNetB4**



ResNet50 + L2



### Análisis el Rendimiento del modelo seleccionado:

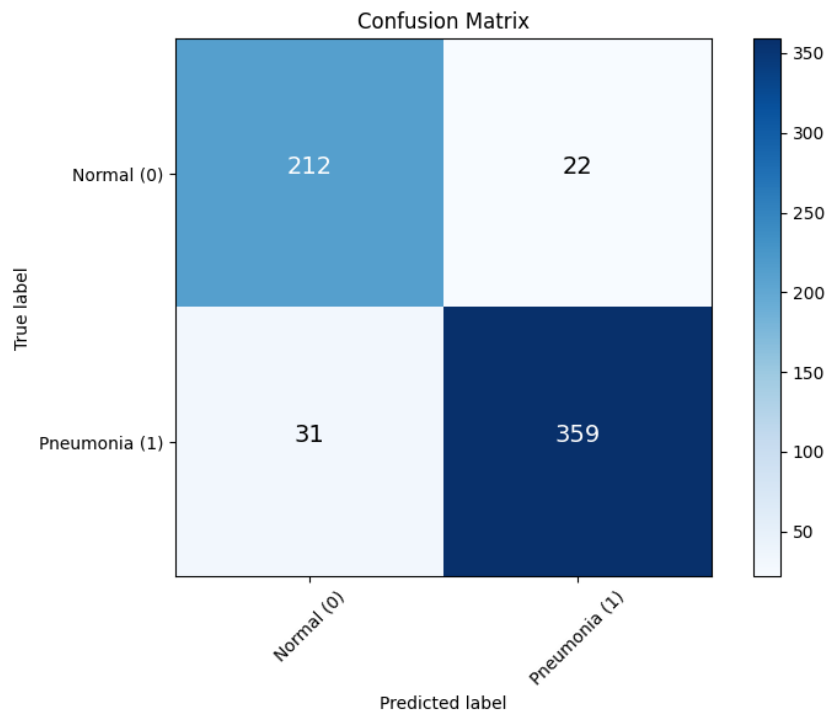
Accuracy: 91.51%

Sensitivity (Recall): 0.9205

Specificity: 0.9060

F1 Score: 0.9313

### Confusión Matrix



A partir de la matriz de confusión presentada, se puede observar que:

212 Verdaderos Negativos (Normal correctamente clasificado)

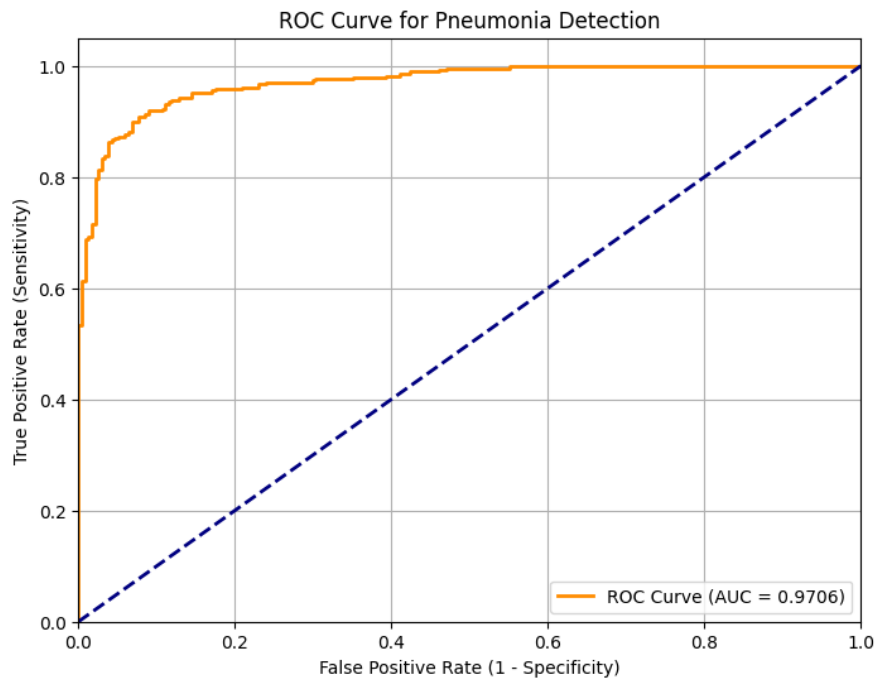
359 Verdaderos Positivos (Neumonía correctamente clasificada)

22 Falsos Positivos

31 Falsos Negativos

y considerando las métricas calculadas (Accuracy: 0.9151, Sensitivity: 0.9205, Specificity: 0.9060, F1-score: 0.9313), se logra evidenciar que el modelo presenta un rendimiento sólido y equilibrado, logrando identificar correctamente la mayoría de los casos tanto de neumonía como normales. La matriz de confusión refleja una proporción alta de aciertos (571 clasificaciones correctas de 624 muestras), lo cual respalda la estabilidad del modelo en escenarios clínicos simulados.

**AUC (Area Under the Curve): 0.9706**



El AUC de 0.9706 confirma una excelente capacidad discriminativa, evidenciando que el modelo separa de manera fiable ambas clases. Este valor destaca particularmente que, incluso bajo diferentes umbrales de decisión, el sistema mantendría un alto desempeño.

## Conclusión

El modelo basado en ResNet50 con regularización L2 logró un desempeño sólido en la detección de neumonía en rayos X. A partir de la matriz de confusión y las métricas obtenidas (Accuracy: 0.9151, Sensitivity: 0.9205, Specificity: 0.9060, F1-score: 0.9313, AUC: 0.9706), se concluye que el sistema presenta alta capacidad discriminativa, identifica correctamente la mayoría de los casos positivos y mantiene un balance adecuado entre la detección de neumonía y la reducción de falsos positivos.

Los errores observados (22 falsos positivos y 31 falsos negativos) se distribuyen de manera esperada en escenarios clínicos reales, especialmente en casos de patrones radiológicos sutiles o atípicos. En general, el modelo demuestra buena generalización y estabilidad frente a un conjunto de prueba independiente.