

# Documentación práctica 1 INAR

## 1. Resumen del Modelo Original CNN

El modelo original contaba con una arquitectura CNN simple diseñada para clasificar imágenes de perros y gatos. Utilizaba capas convolucionales y densas en una configuración básica.

Tamaño de imagen: El modelo trabajaba con imágenes de (256, 256) píxeles.

Batch Size: El tamaño era de 125.

Capas Originales:

- 4 capas convolucionales, con un número de filtros creciente (32, 64, 128, y 256).
- Después de cada capa convolucional había una capa de MaxPooling2D con tamaño (2, 2) para reducir la dimensionalidad.
- Una capa de Dropout para reducir el sobreajuste.
- Una capa Flatten para aplanar la salida de las capas convolucionales antes de pasar a las capas densas.
- Una capa densa de 128 neuronas seguida de una capa de salida de 2 neuronas.

## 2. Cambios Realizados

En mi versión de la CNN, he realizado varios ajustes con el objetivo de mejorar su precisión. Los cambios realizados son los siguientes:

- Reducción del tamaño de la imagen:
  - Cambié el tamaño de las imágenes de entrada de (256, 256) a (128, 128). La razón de este ajuste fue reducir el peso del modelo, haciendo el entrenamiento más rápido y sin perder demasiada información visual para la clasificación.

- **Aumento del Número de Capas Convolucionales:**
  - Añadí una capa de Conv2D con 256 filtros y otra con 512 filtros. Cada una de estas capas tienen después una de MaxPooling.
  - Este aumento de la profundidad permite al modelo capturar características más complejas en etapas más profundas, para diferenciar entre imágenes con mayor precisión.
- **Ajuste de la capa densa final:**
  - En el modelo original, la capa densa final tenía 512 neuronas, la cual podía llevar al sobreajuste.
  - Reduje esta capa a 64 neuronas, disminuyendo el riesgo de sobreajuste.
- **Dropout:**
  - Añadí Dropout (0.5) después de la última capa convolucional y en la capa densa. Esto reduce el sobreajuste al evitar que el modelo dependa demasiado de ciertas neuronas.

### 3. Pruebas Realizadas

	A	B	C	D	E	F	G	H	I
1	Intento	Tamaño de Imagen	Capas Convolucionales	Capas de MaxPooling	Capas Dropout	Capas Flatten	Capas Densas	Épocas	Accuracy
2		1 (128, 128)	4	4	2	1	2	12	0,647
3		2 (128, 128)	5	5	2	1	2	12	0,689
4		3 (192, 192)	5	5	2	1	2	10	0,482

En el intento 1, usé el modelo base con una configuración que incluía un tamaño de imagen de (128, 128), cuatro capas convolucionales, cuatro capas de MaxPooling, dos capas de Dropout y dos capas densas en un entrenamiento de 12 épocas. Este modelo alcanzó un accuracy de 0.647.

En el intento 2, añadí una capa convolucional y una capa de MaxPooling para aumentar la profundidad del modelo. Mantuve el mismo tamaño de imagen y número de épocas. Estos cambios permitieron que el modelo capturara más detalles visuales, mejorando el accuracy a 0.689.

En el intento 3, aumenté el tamaño de imagen a (192, 192) para captar más detalles en cada imagen. Sin embargo, reduje las épocas a 10 para reducir el tiempo de entrenamiento, ya que este era muy extenso. Este cambio no fue del todo efectivo, ya que el accuracy cayó a 0.482, probablemente porque el modelo necesitaba más tiempo de entrenamiento.

En conclusión, el segundo intento consiguió el mejor balance entre profundidad y tiempo de entrenamiento.

## **4. Exploración de errores y mejoras**

Para revisar los errores del modelo, he creado un fichero llamado predictions.csv, que contiene las predicciones generadas para cada imagen de prueba, el valor 1 corresponde a perros y 0 a gatos. Para este análisis, me enfoqué en las imágenes que van del 1000 hasta 1009.

Después de revisar los resultados, el modelo falló en clasificar estas imágenes:

- Imagen 1009: Interpretada como gato incorrectamente.
- Imagen 1004: Interpretada como gato incorrectamente.
- Imagen 1003: Interpretada como perro incorrectamente.
- Imagen 1002: Interpretada como gato incorrectamente.

### **Análisis de los errores:**

1. Imagen 1009: Aquí aparecen dos perros, uno de ellos en primer plano. La imagen tiene una iluminación alta y varias sombras, lo que podría haber confundido al modelo, ya que podría relacionar sombras y zonas oscuras con características de un gato.
2. Imagen 1004: Es un perro grande acostado. La pose relajada y el ángulo en el que fue captada la imagen, podrían haber confundido al modelo a la hora de detectar los rasgos de un perro.
3. Imagen 1003: Esta es una foto de un gato al aire libre, en el que hay césped de fondo. Aunque es claramente un gato, la textura del pelaje y el entorno podrían haber hecho que el modelo no reconociera las características de gato.
4. Imagen 1002: Un perro pequeño jugando en el suelo con un juguete en la boca. Cuando los perros están en poses poco comunes o interactuando con objetos, el modelo parece tener problemas para identificar correctamente sus rasgos faciales.

## **Posibles causas de los errores:**

Estos fallos indican que el modelo tiene problemas en situaciones como:

- Imágenes con varios elementos o sujetos.
- Perros en poses poco comunes o interactuando con objetos.
- Condiciones de iluminación que generan sombras y contraste fuertes.

## **Posibles mejoras del modelo:**

Para mejorar en estos casos, podría ser útil aplicar Data Augmentation en el entrenamiento, agregando más variedad en poses, rotaciones, y ajustes de iluminación en las imágenes. Además, incluir más fotos en el dataset con características similares a las de estas imágenes podría ayudar al modelo a reconocer mejor estos escenarios.

## **1. Resumen del Modelo Original VGG16**

El modelo original es una red pre entrenada VGG16 diseñada para clasificación en 1000 categorías, que cuenta con capas convolucionales y densas.

- Tamaño de imagen: Este modelo utiliza imágenes de tamaño (224, 224) píxeles.
- Batch Size: El tamaño por lotes es de 125.
- Capas:
  - Capas convolucionales: 5 bloques de convolución, cada uno con varias capas que suman hasta 13 capas convolucionales, aumentando progresivamente los filtros
  - Capas de MaxPooling: Cada bloque de convolución termina con una capa de MaxPooling para reducir las dimensiones.
  - Capas densas: Después de aplanar las características, cuenta con dos capas densas de 4096 neuronas y una capa de salida de 1000 neuronas.

## 2. Cambios Realizados en el Modelo VGG16

He adaptado el modelo VGG16 original para optimizarlo a la tarea de clasificación de perros y gatos. Para ello, he realizado los siguientes ajustes:

- **Remplazo de la Capa de Salida:** En el modelo original, la VGG16 estaba configurada para clasificar 1000 categorías. Reemplacé la última capa densa con una capa de dos neuronas, configurada para distinguir solo entre perros y gatos, usando una activación softmax.
- **Congelación de Capas Iniciales:** En la versión final, congelé todas las capas de la VGG16 excepto la última capa de salida, permitiendo que el modelo conserve las características de las capas pre entrenadas en ImageNet. Esto reduce el tiempo de entrenamiento y mejora la estabilidad del modelo.
- **Capas Adicionales:** Agregué capas densas adicionales después de la capa de aplanado para mejorar la capacidad del modelo de captar patrones más detallados en las imágenes de perros y gatos. Añadí las siguientes capas:
  - Una capa densa de 256 neuronas con activación relu y Dropout (0.5) para reducir el sobreajuste.
  - Una capa adicional de 128 neuronas seguida de otro Dropout (0.5), lo que ayuda a mejorar la generalización del modelo en el conjunto de validación.

## 3. Exploración de errores y mejoras

En este apartado, seguiré el mismo procedimiento que en la CNN para revisar la precisión del modelo VGG16 en un conjunto de 10 imágenes de prueba. Identificaré los errores de clasificación y analizaré las posibles causas de cada uno. Por último, una vez analizadas las imágenes, propondré una serie de mejoras para mi modelo VGG16.

### Análisis de Errores:

1. **Imagen 1007:** Esta imagen muestra a un gato de espaldas con la cabeza girada, mirando hacia un lado. En la foto no se aprecian bien los rasgos faciales, como los ojos o la nariz, que suelen ser clave para la detección. Además, las sombras en el fondo podrían haber ayudado a que el modelo interpretara incorrectamente las características, confundiendo al gato con un perro.
2. **Imagen 1004:** Aquí tenemos a un perro tumbado en el suelo y captado desde un ángulo superior. La postura y el ambiente pueden dar una impresión más ambigua para el modelo, ya que no se resaltan características específicas de un perro. Como el perro ocupa un gran espacio en la imagen sin un enfoque claro

en su cara, el modelo podría haber tenido problemas para capturar las características típicas de un perro.

### **Posibles Causas del Error:**

- **Falta de Rasgos Faciales Visibles:** En ambas imágenes, las caras de los animales no están claramente visibles, lo cual parece ser una limitación para el modelo. La VGG16 podría estar entrenada para detectar características faciales que no se encuentran en estas imágenes.
- **Iluminación y Sombras:** La imagen 1007 tiene sombras que podrían haber afectado la identificación, mientras que el ángulo de la imagen 1004 crea un efecto que distrae al modelo de los detalles relevantes.

### **Posibles Mejoras para el Modelo:**

- **Data Augmentation:** Agregar más imágenes de gatos y perros en poses y ángulos raros durante el entrenamiento podría ayudar al modelo a adaptarse a este tipo de situaciones.
- **Entrenamiento con distinta iluminación:** Introducir un dataset de datos con diferentes ambientes de iluminación podría ayudar al modelo a reconocer mejor a los animales independientemente de las sombras.