# Big Data & AI Course: Image Classification Lab

**Presented By: Meriem MEKKI**

25/01/2026

# Introduction

This lab is part of the Big Data and AI course. For this exercise, I chose the Animals-10 dataset from Kaggle.

The workflow included:

1. Exploratory Data Analysis (EDA) to understand the dataset and identify patterns.
2. Data preprocessing of the images
3. Experimenting with different classifiers: basic CNN, Pretrained ResNet50, Pretrained EffecientNet.

# EDA

This dataset contains about 28K medium quality animal images belonging to 10 categories: dog, cat, horse, spyder, butterfly, chicken, sheep, cow, squirrel, elephant.

| | |
|---|---|
| **cane** | 4863 |
| **cavallo** | 2623 |
| **elefante** | 1446 |
| **farfalla** | 2112 |
| **gallina** | 3098 |
| **gatto** | 1668 |
| **mucca** | 1866 |
| **pecora** | 1820 |
| **ragno** | 4821 |
| **scoiattolo** | 1862 |

# EDA

- The dataset is imbalanced.
- Some classes have much higher counts than others.
- Dominant classes:

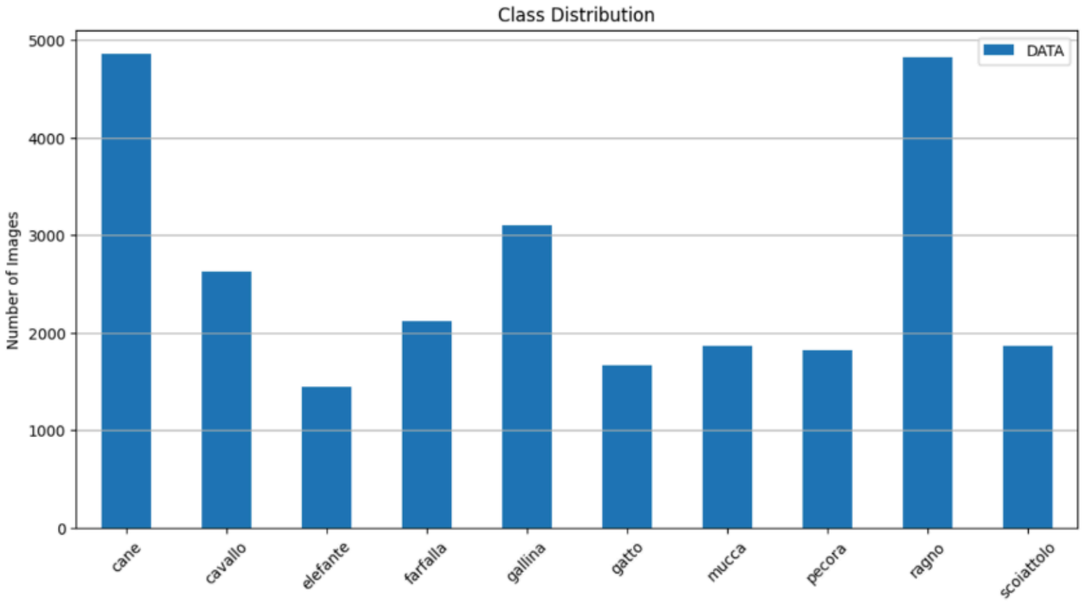
cane and ragno are the most represented classes (≈ 4.8–4.9k images each).

- gallina is also relatively well represented (~3.1k).
- Underrepresented classes:

elefante is the smallest class (~1.4k).

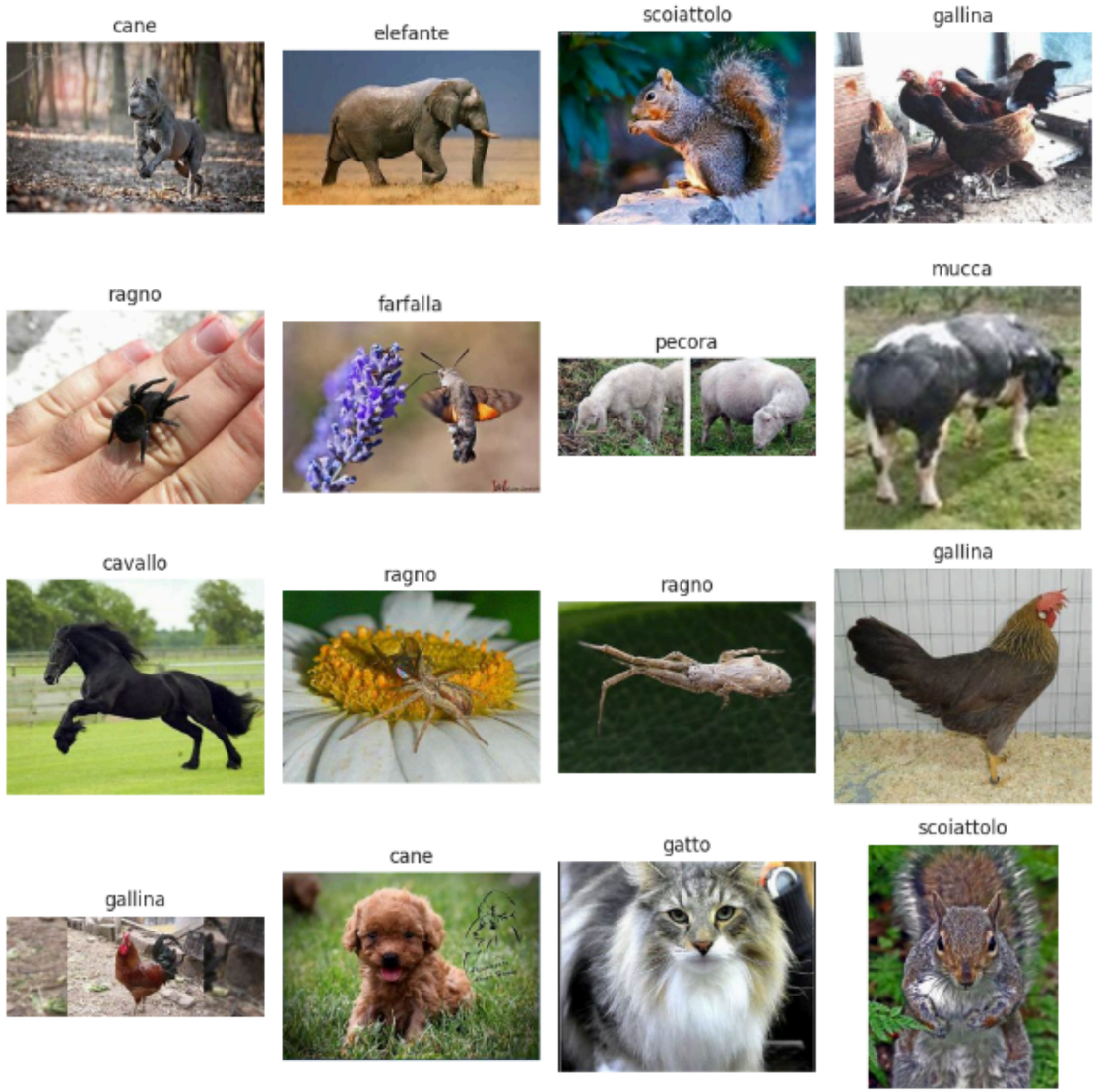
- gatto, mucca, pecora, scoiattolo are also on the lower side (~1.6–1.9k).
- Mid-range classes:

cavallo and farfalla sit in the middle (~2.1–2.6k).



# EDA

some samples from the dataset



## Data Splitting Strategy

Dataset split into:

- 80% training
- 20% testing

Training set further split into:

- 80% training
- 20% validation

Validation data used for hyperparameter tuning and early stopping

Test data kept completely unseen until final evaluation

## Data Splitting Strategy

## Preprocessing

- All images were resized to a fixed resolution of 224 × 224 pixels to ensure consistent input dimensions across models.
- Model-specific preprocessing was applied:
  - Basic CNN:
    - Pixel values normalized to [0,1][0,1][0,1] using rescaling (1/255).
  - Pretrained ResNet:
    - Used resnet.preprocess_input, matching ImageNet normalization.
  - Pretrained EfficientNet:
    - Used efficientnet.preprocess_input, consistent with its ImageNet pretraining.
- Categorical labels encoded using one-hot encoding

## Preprocessing

## Data Augmentation

Applied only to training data to improve generalization
Augmentation techniques:

- Horizontal flipping
- Random rotation
- Random zoom
- Random contrast

Helps reduce overfitting and increases data diversity

Data Augmentation

## CNN Architecture

Three convolutional blocks:

- Conv2D + ReLU
- MaxPooling
- Increasing filter depth: 32 → 64 → 128

Fully connected layer with 128 neurons

Output layer:

- Softmax activation
- 10 classes

CNN Architecture

## Transfer Learning Setup

- Two pretrained architectures were evaluated:
    - ResNet50
    - EfficientNetB7
- Both models were initialized with ImageNet weights
- The original classification head was removed (include_top = False)
- A custom classification head was added for the 10 target classes
- The pretrained backbone was frozen during training
- Custom classification head:
    - Global max pooling applied to extract compact feature vectors
    - Fully connected layers:
    - Dense (128) + ReLU
    - Dense (256) + ReLU
    - Batch Normalization for training stability
    - Dropout (0.45) to reduce overfitting
    - Final Softmax layer with 10 output classes

Transfer Learning Setup

## Training Configuration (Shared Across Models)

- Optimizer: Adam
- Learning rate: 1e-4
- Loss function: Categorical Cross-Entropy
- Batch size: 32
- Epochs: up to 30

## Training Configuration (Shared Across Models)

## Callbacks & Regularization

- Early Stopping
  - Monitored validation loss
  - Patience: 5 epochs
  - Restored best weights
- Reduce Learning Rate on Plateau
  - Factor: 0.2
  - Minimum LR: 1e-6
- Model Checkpoint
  - Saved best model based on validation accuracy
- TensorBoard for training monitoring

## Callbacks & Regularization

## Experimental Design

Three models evaluated:

- Basic CNN (trained from scratch)
- ResNet50 (transfer learning)
- EfficientNetB7 (transfer learning)

Same:

- Input size
- Dataset splits
- Data augmentation
- Training hyperparameters

Differences limited to:

- Network architecture
- Preprocessing function

## Experimental Design

# Classification Reports

## Basic CNN

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| cane | 0.625801 | 0.672566 | 0.648341 | 1017.000000 |
| cavallo | 0.606762 | 0.644612 | 0.625115 | 529.000000 |
| elefante | 0.592453 | 0.523333 | 0.555752 | 300.000000 |
| farfalla | 0.611111 | 0.819712 | 0.700205 | 416.000000 |
| gallina | 0.684411 | 0.586319 | 0.631579 | 614.000000 |
| gatto | 0.478992 | 0.186885 | 0.268868 | 305.000000 |
| mucca | 0.427617 | 0.569733 | 0.488550 | 337.000000 |
| pecora | 0.647059 | 0.470588 | 0.544892 | 374.000000 |
| ragno | 0.716141 | 0.803330 | 0.757234 | 961.000000 |
| scoiattolo | 0.549342 | 0.447721 | 0.493353 | 373.000000 |
| accuracy | 0.621316 | 0.621316 | 0.621316 | 0.621316 |
| macro avg | 0.593969 | 0.572480 | 0.571389 | 5226.000000 |
| weighted avg | 0.619005 | 0.621316 | 0.611942 | 5226.000000 |

## EfficientNetB7

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| cane | 0.971680 | 0.978368 | 0.975012 | 1017.000000 |
| cavallo | 0.971154 | 0.954631 | 0.962822 | 529.000000 |
| elefante | 0.976898 | 0.986667 | 0.981758 | 300.000000 |
| farfalla | 0.980861 | 0.985577 | 0.983213 | 416.000000 |
| gallina | 0.977492 | 0.990228 | 0.983819 | 614.000000 |
| gatto | 0.951768 | 0.970492 | 0.961039 | 305.000000 |
| mucca | 0.935976 | 0.910979 | 0.923308 | 337.000000 |
| pecora | 0.945799 | 0.933155 | 0.939435 | 374.000000 |
| ragno | 0.990644 | 0.991675 | 0.991160 | 961.000000 |
| scoiattolo | 0.989160 | 0.978552 | 0.983827 | 373.000000 |
| accuracy | 0.972828 | 0.972828 | 0.972828 | 0.972828 |
| macro avg | 0.969143 | 0.968032 | 0.968539 | 5226.000000 |
| weighted avg | 0.972758 | 0.972828 | 0.972756 | 5226.000000 |

## ResNet50

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| cane | 0.970297 | 0.963618 | 0.966946 | 1017.000000 |
| cavallo | 0.951128 | 0.956522 | 0.953817 | 529.000000 |
| elefante | 0.979522 | 0.956667 | 0.967960 | 300.000000 |
| farfalla | 0.971429 | 0.980769 | 0.976077 | 416.000000 |
| gallina | 0.955766 | 0.985342 | 0.970329 | 614.000000 |
| gatto | 0.963211 | 0.944262 | 0.953642 | 305.000000 |
| mucca | 0.896848 | 0.928783 | 0.912536 | 337.000000 |
| pecora | 0.945355 | 0.925134 | 0.935135 | 374.000000 |
| ragno | 0.984456 | 0.988554 | 0.986501 | 961.000000 |
| scoiattolo | 0.983287 | 0.946381 | 0.964481 | 373.000000 |
| accuracy | 0.963643 | 0.963643 | 0.963643 | 0.963643 |
| macro avg | 0.960130 | 0.957603 | 0.958742 | 5226.000000 |
| weighted avg | 0.963865 | 0.963643 | 0.963658 | 5226.000000 |

Classification Reports

## Results Discussion

EfficientNetB7 achieved the best overall performance, reaching 97% accuracy and the highest macro and weighted F1-scores, indicating more consistent performance across all classes. It shows particularly strong recall for minority classes such as elefante and farfalla, suggesting better generalization on less frequent categories. ResNet50 and the basic CNN achieved slightly lower but comparable performance (96% accuracy), with similar precision–recall trade-offs across most classes. The basic CNN, despite being trained from scratch, performed competitively, demonstrating that the dataset is sufficiently informative; however, transfer learning models—especially EfficientNet—provide a small but consistent improvement, particularly in class balance and recall stability.

## Results Discussion

# Conclusion

- Three image classification models were evaluated: Basic CNN, ResNet50, and EfficientNetB7.
- All models achieved high accuracy (>96%), confirming the quality of the dataset and preprocessing pipeline.
- Transfer learning improved performance, with EfficientNetB7 achieving the best and most balanced results.
- Using the same training setup across models ensured a fair and reliable comparison.
- EfficientNet demonstrated stronger generalization, especially on less represented classes.