Introduction to AI and ML

Dr. Lorenzo Papa

Internal Research Fellow in AI4EO, ESA ESRIN Frascati, Rome, Italy

Email: lorenzo.papa@esa.int

A brief introduction of me

Internal Research Fellow, ESA Φ-lab, ESRIN, IT

PhD in Computer Science Engineering, Sapienza University, IT

Master degree in AI and Robotics, Sapienza University, IT

- **Visiting Research**, The University of Sydney, NSW, Australia
- **Visiting Research,** ESA Φ-lab, ESRIN, IT

Links & Contacts:

- Email: Lorenzo.Papa@esa.int
- Google Scholar: Lorenzo Papa - Google Scholar
- LinkedIn: Lorenzo Papa | LinkedIn

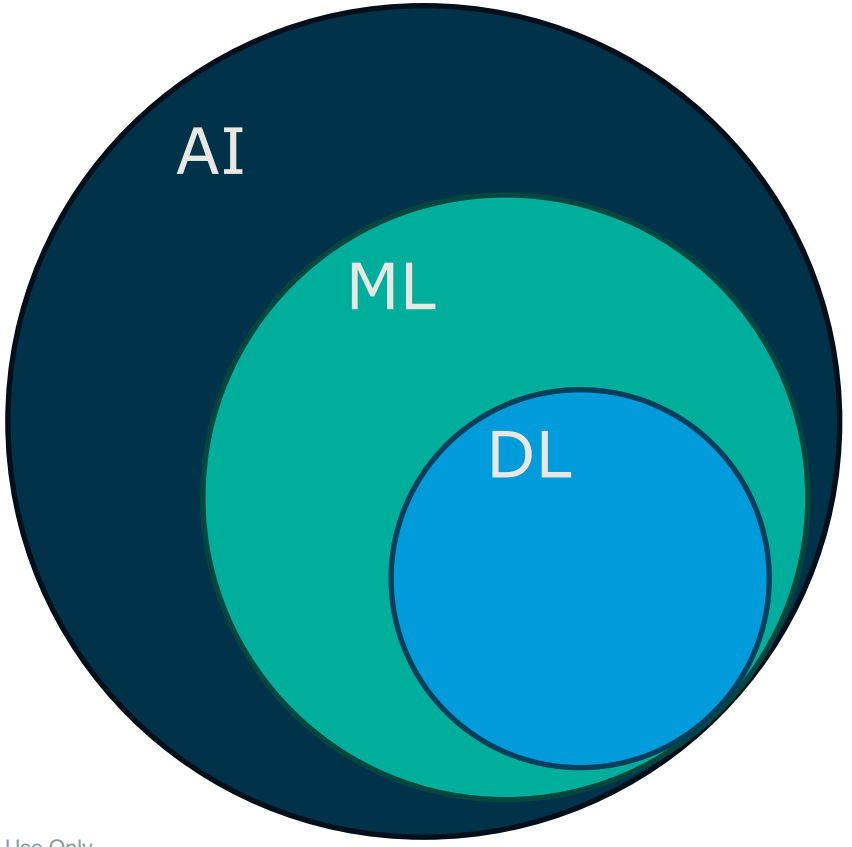
Outline

- Introduction to AI, ML and DL
- ESA's Earth Observation Missions and ESA Φ-lab
- Data handling: from ML to DL
- A DL training pipeline

https://github.com/lorenzopapa5/BiDs_training_course_2025/tree/main

Overview of Learning systems

"Machine learning is the subfield of computer science that gives computers the ability to learn without being explicitly programmed" — *Arthur Samuel, 1959*



Artificial Intelligence

- AI refers to computer systems designed to execute tasks that typically demand human intellect, including interpreting visual data, recognizing speech, and making decisions.
- It is a scientific discipline focused on replicating and explaining intelligent actions through computational methods.
- AI comes into play when machines simulate "cognitive" abilities commonly linked to human thought processes, such as acquiring knowledge and solving complex problems.

From Machine Learning to Deep Learning

- Learning is the process of **acquiring knowledge** or skills through experience, study, or instruction. In humans and animals, it involves adapting behaviour based on past outcomes. In **machines**, it means improving performance on a task based on data.
- Machine Learning is a subset of learning where **machines learn from data** to make predictions or decisions **without being explicitly programmed**. It uses algorithms that identify patterns and improve over time.
- Deep Learning is a specialized branch of ML that uses **artificial neural networks with many layers** to learn complex patterns from large datasets. It powers modern AI applications like image recognition, natural language processing, and autonomous systems.

From Deep Learning to … Future?

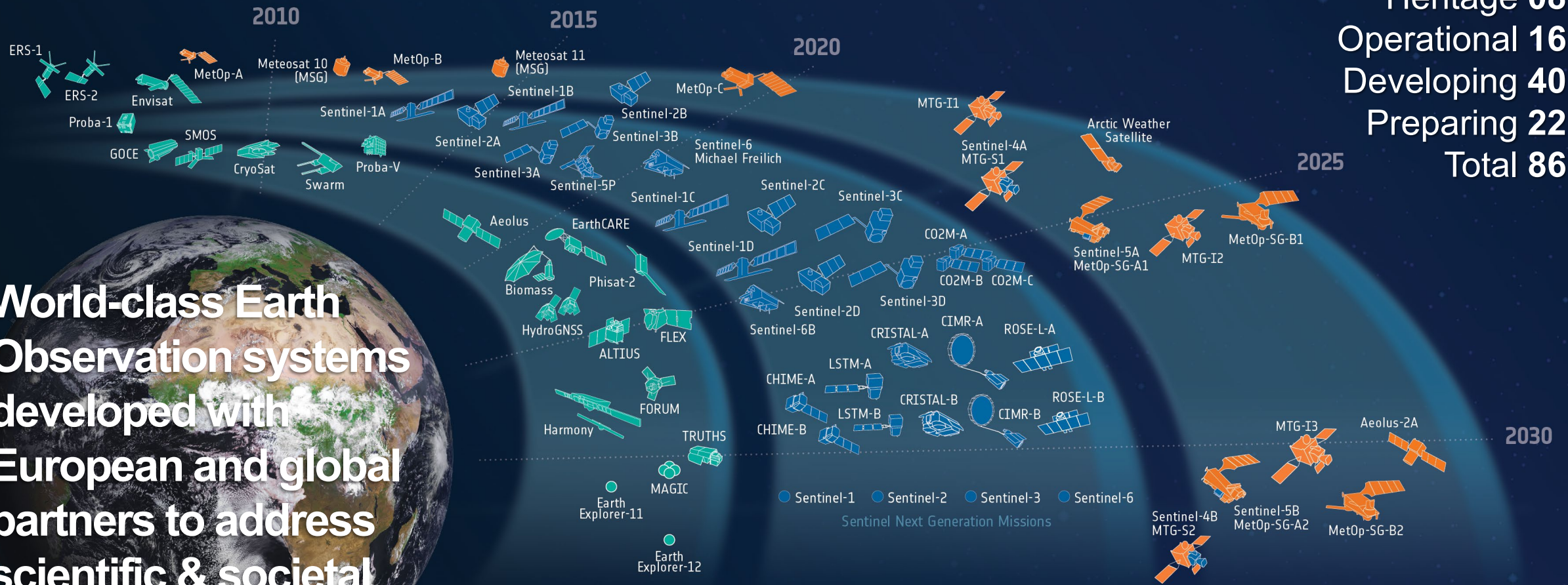
Yes, we've all heard these buzzwords …

… but first, why we need all of this?

ESA's Earth Observation Missions

Satellites

Heritage **08**
Operational **16**
Developing **40**
Preparing **22**
Total **86**

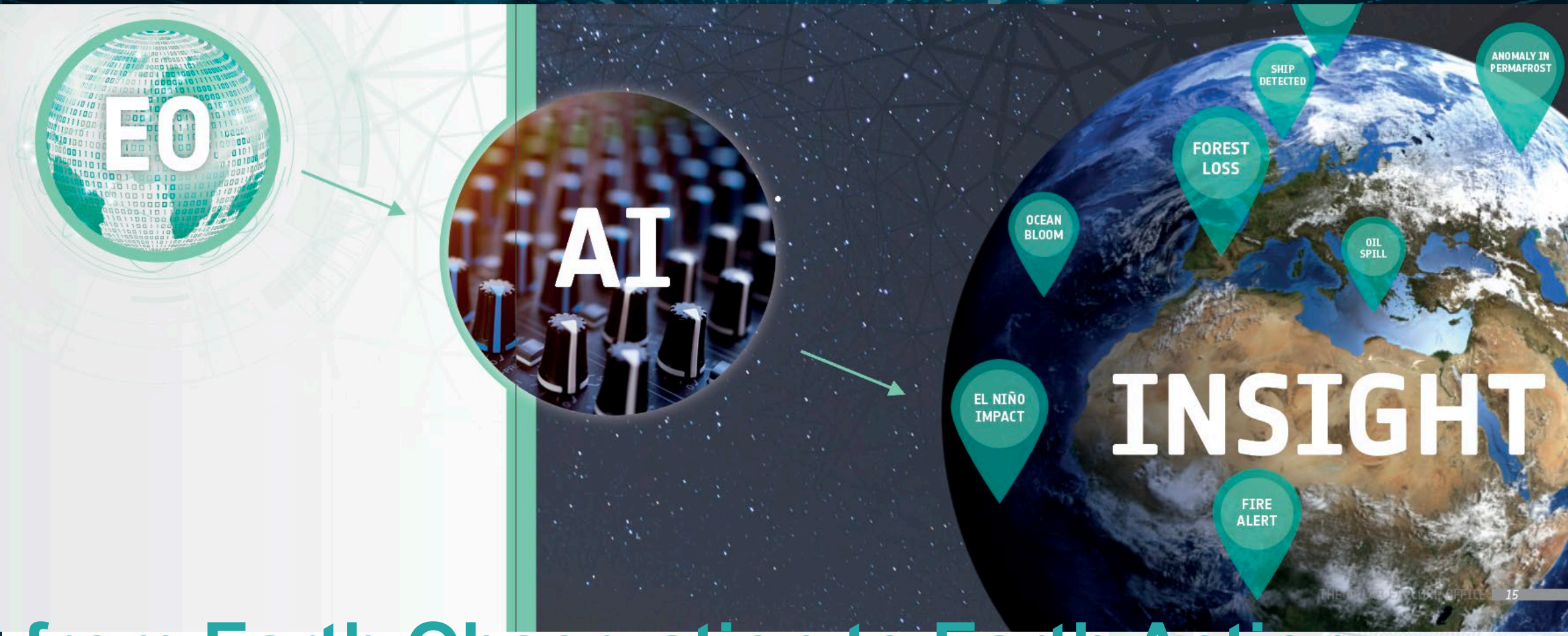


World-class Earth Observation systems developed with European and global partners to address scientific & societal challenges

Science esa | Copernicus European Union | Meteorology EUMETSAT

The ESA Φ-lab – Why ?

esa Φ-lab



from Earth Observation to Earth Action

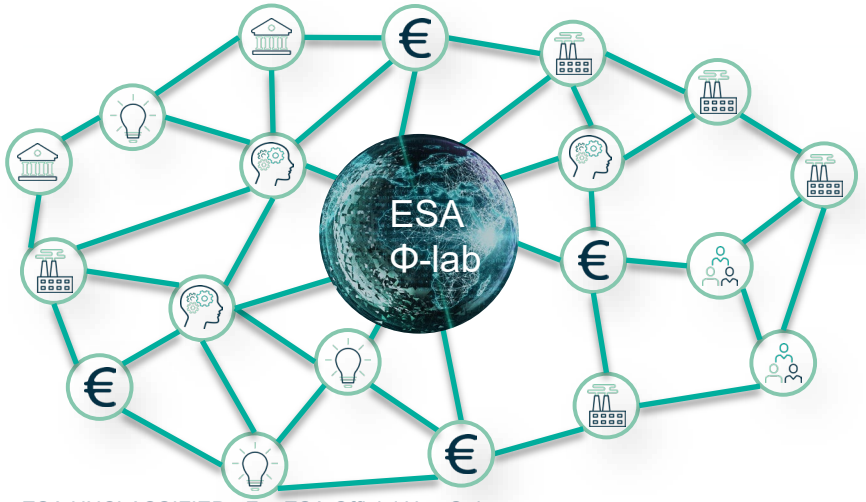
From data to actionable information

The ESA Φ-lab – How?

Φ-lab aims to become "the reference" for the transformational innovation and a key influencer (by reputation and authority) in the Earth Observation ecosystem

Catalyst

- **Attract EO academic and industrial researchers to generate transformative ideas**
- **Exploit fail fast ethos, rapidly prototyping concepts**
- **An informal but rigorous, multi-disciplinary, collaborative environment**
- **Act as facilitator to foster competitiveness growth and entrepreneurial initiatives**
- **Implement investment actions from ESA MSs or in the investors industry**

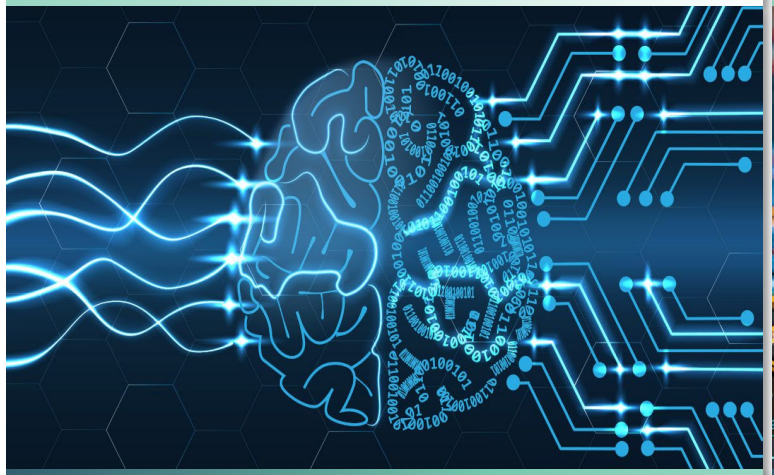


Bridge

- **Be the bridge between the European start-ups, academic and industrial researchers, New Space operators, Investors, ICT players, EO world leaders, and ESA**
- **Act as hub stimulating, connecting, and developing a growing ecosystem of talents and capabilities across Europe**

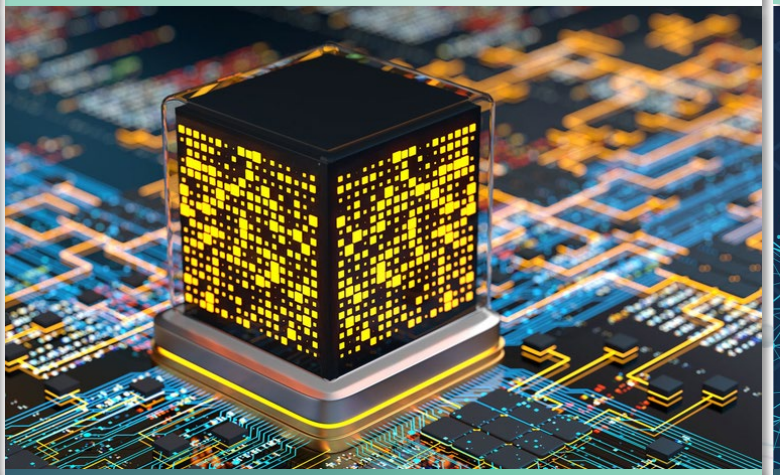
Innovation Technologies axis and Applications

AXIS I
Augmented Intelligence



Foundation Models
Digital Assistant and Twins
Generative AI
Decision Intelligence, Agentic AI
Explainability (xAI)
Physics-Informed ML
AI4EO for Climate, Health, and Human

AXIS II
Innovative Computing Paradigms



On-Board AI
Quantum Computing
Hybrid HPC Computing
Neuromorphic
Biocomputing and others

AXIS III
Innovative Computing Paradigms

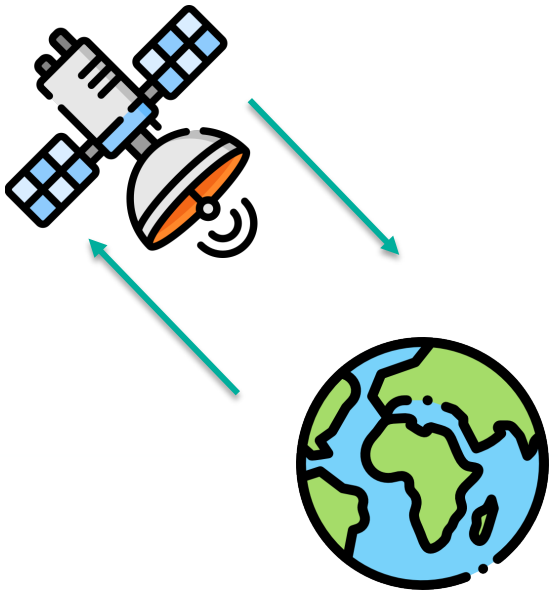
Cognitive Space
VR/AR Immersive Visualisation
Web 3.0
IOT
Distributed Ledgers/Blockchain

Data handling: from manual feature extraction (ML) to automatic processing (DL)

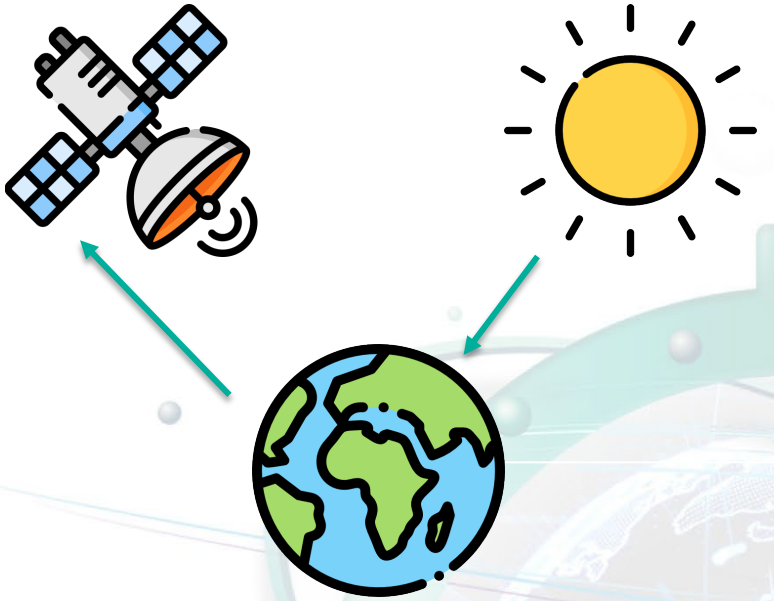
Remote Sensing Data

Remote Sensing Data

Active Sensors



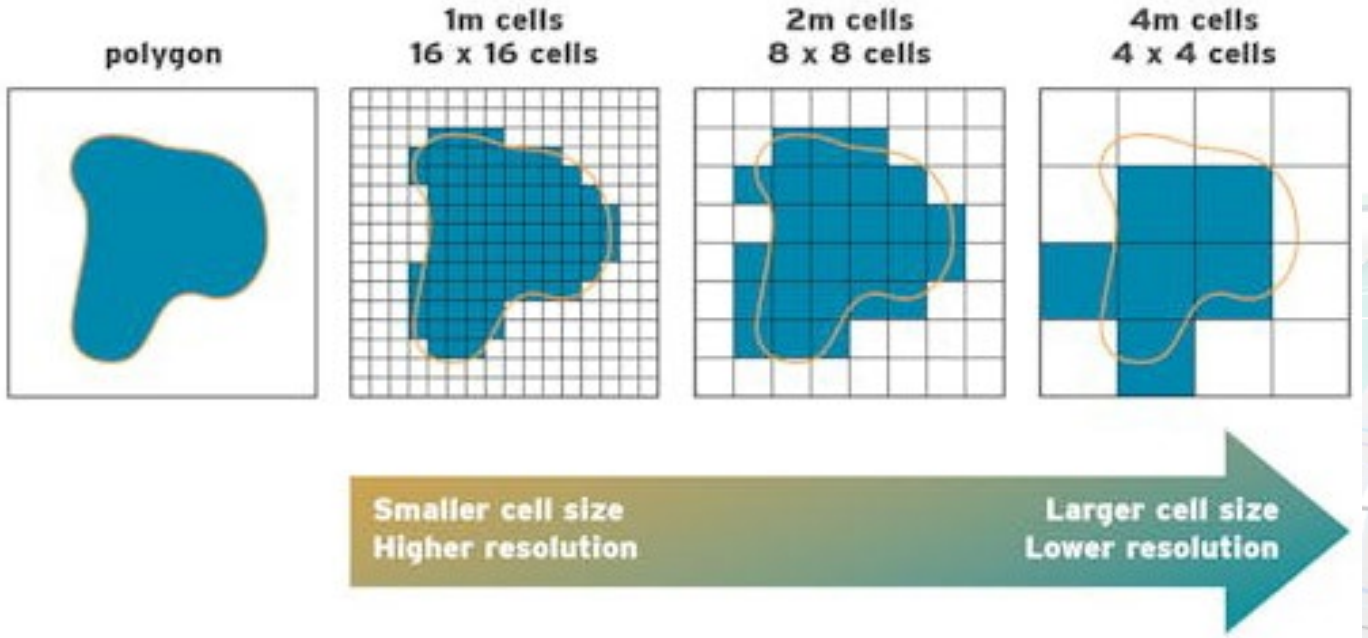
Passive Sensors



So, based on these settings, which are the sensor-level attributes that affect the quality and type of features that can be extracted from satellite images?

Remote Sensing Data

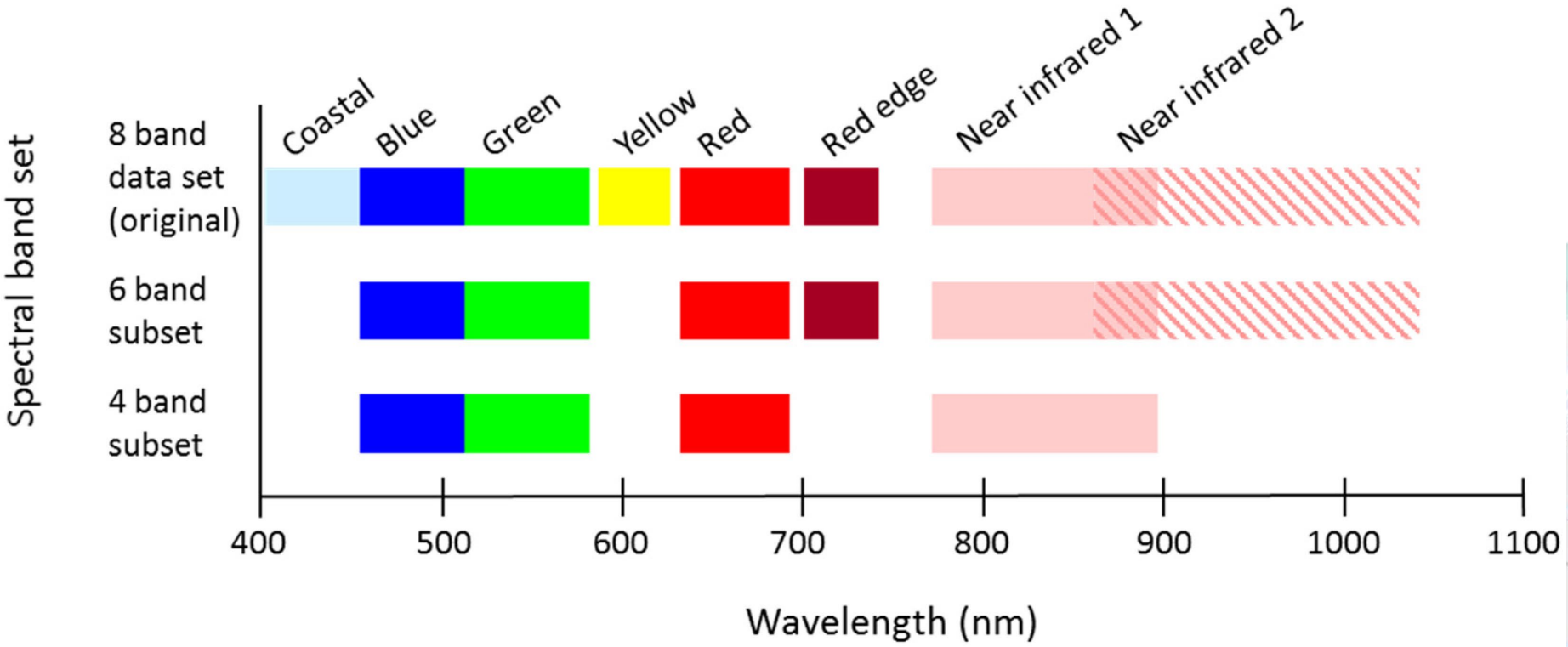
Spatial Resolution*: it represents the smallest possible feature that can be detected. The spatial resolution quantifies the capability to separate two close targets. The pixel size is often considered as spatial resolution. It depends on the design of the sensor, while the pixel size depends on the digital sampling of the signal.



** Optical/Multispectral sensors. For SAR sensors, this is primarily determined by the acquisition mode*

Remote Sensing Data

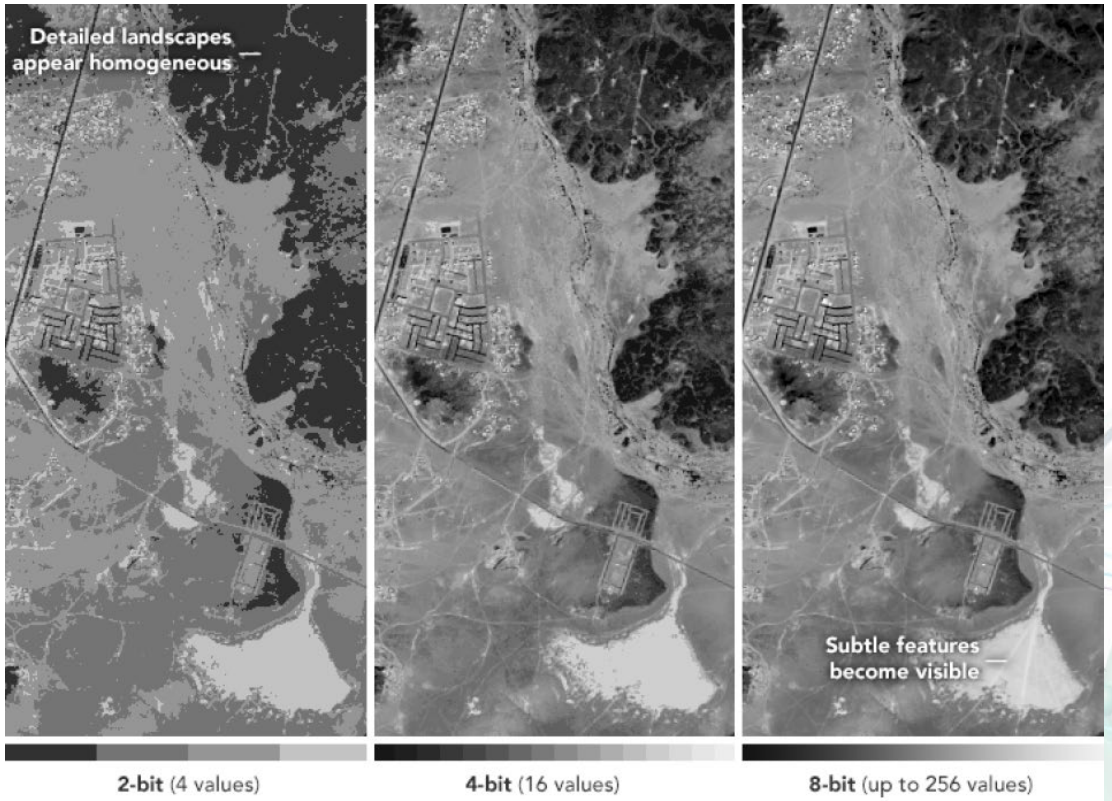
Spectral Resolution*: it represents the wavelength of the different frequency bands recorded. Spectral resolution describes the ability of a sensor to define fine wavelength intervals.



** Optical/Multipsectral sesnors. For SAR sensors, this corresponds to radar frequency and signal polarization*

Remote Sensing Data

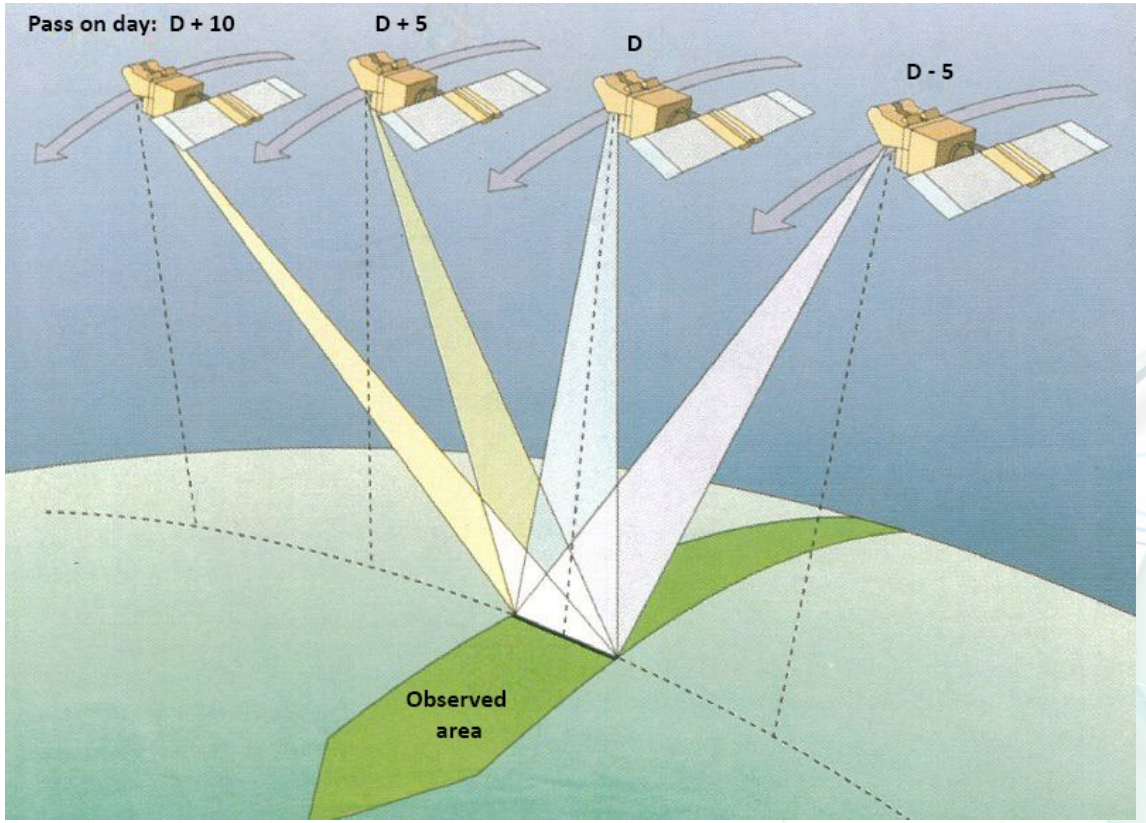
Radiometric Resolution*: it refers to the number of different intensities of radiation the sensor can distinguish. The greater the radiometric resolution, the more accurate the sensed image will be.



** Measured in bits for optical sensors (light intensity), and typically in decibels for SAR (signal power)*

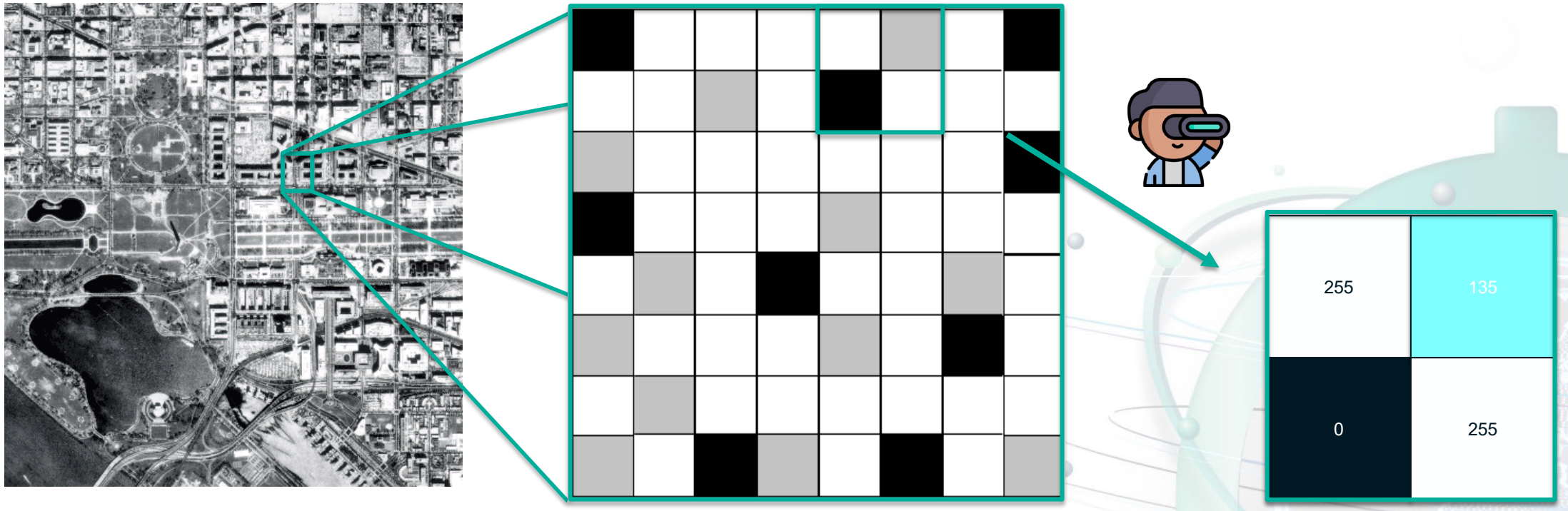
Remote Sensing Data

Temporal Resolution: is the frequency of flyovers by the satellite. This resolution can become relevant in time series studies or those requiring an averaged or mosaic image. (revisit time)



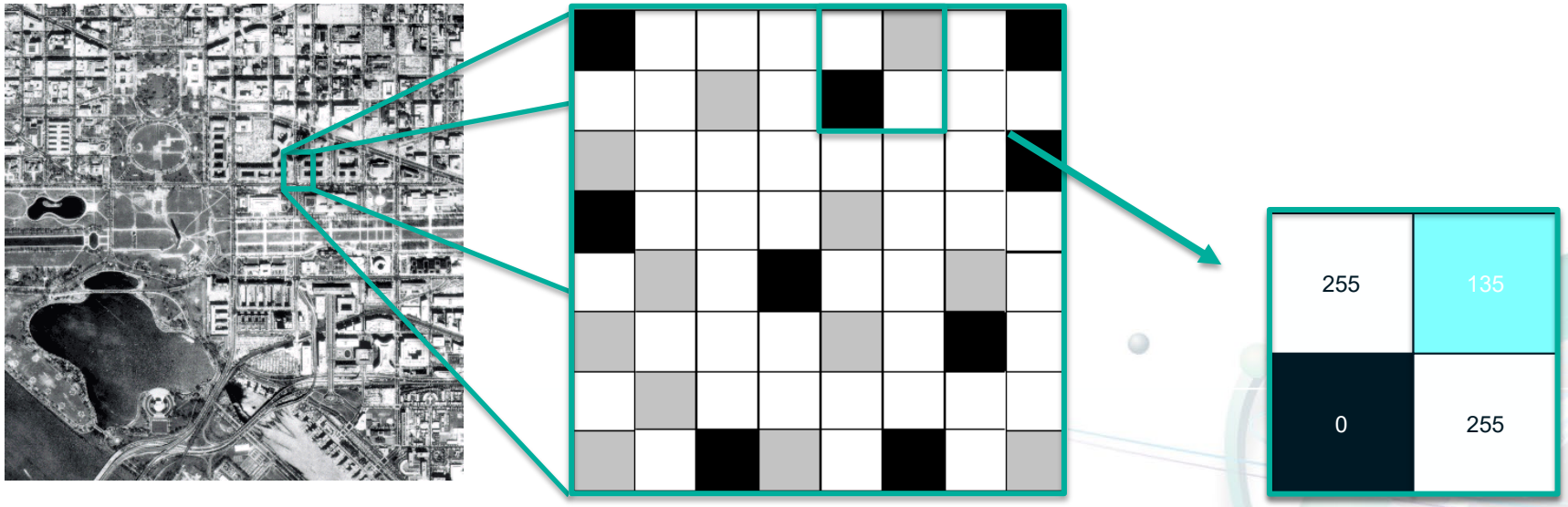
Introduction

An image, is represented as a matrix of numerical values



Introduction

An image, is represented as a matrix of numerical values



… it seems easy, no? …

How to handle Images in Python

OpenCV

```python
# Using OpenCV
import cv2

# General image path
image_path = 'path/to/your/image.jpg'

# Load image using OpenCV
image_bgr = cv2.imread(image_path)

# Convert BGR to RGB
image_rgb_cv2 = cv2.cvtColor(image_bgr, cv2.COLOR_BGR2RGB)

# Display image using OpenCV
cv2.imshow('Image in RGB (OpenCV)', image_rgb_cv2)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

matplotlib

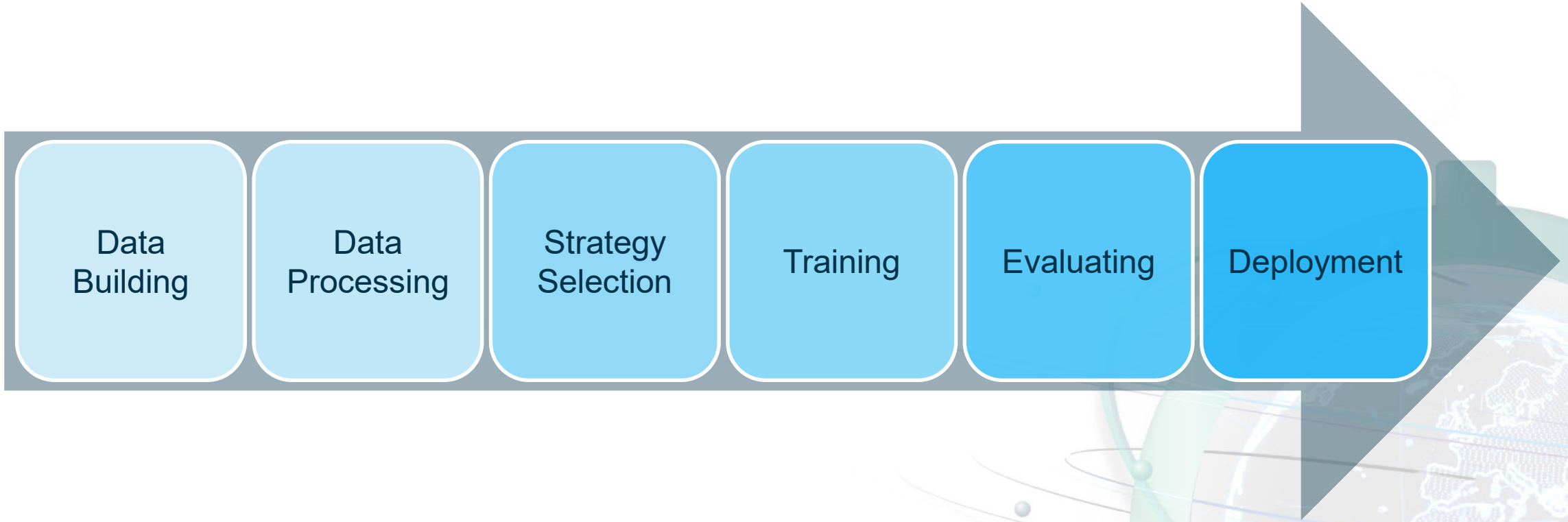
pillow

```python
# Using Pillow and Matplotlib
from PIL import Image
import matplotlib.pyplot as plt

# Load image using Pillow
image_pil = Image.open(image_path)

# Convert to RGB (in case it's not)
image_rgb_pil = image_pil.convert('RGB')

# Display using Matplotlib
plt.imshow(image_rgb_pil)
plt.title('Image in RGB (Pillow + Matplotlib)')
plt.axis('off')
plt.show()
```

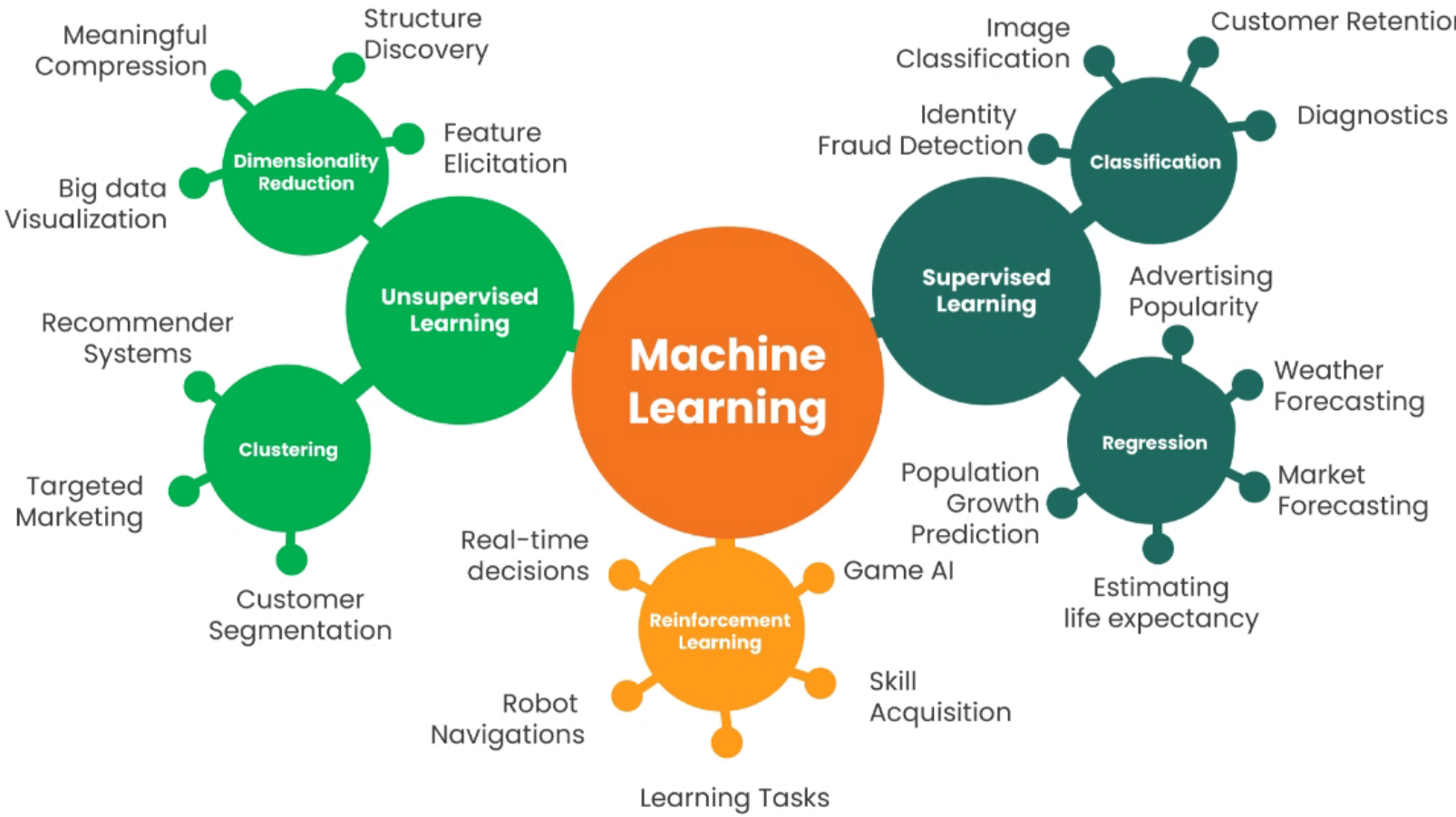
From Images to Feature Extraction with Filters

… GoTo Image_Filtering.ipynb

Machine Learning



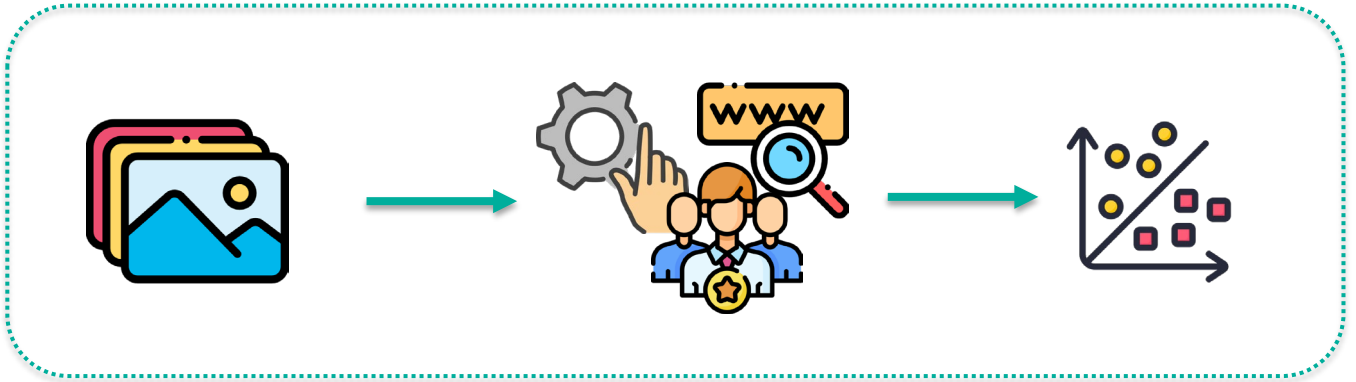
Machine Learning



*Source: **Axtria Inc.***

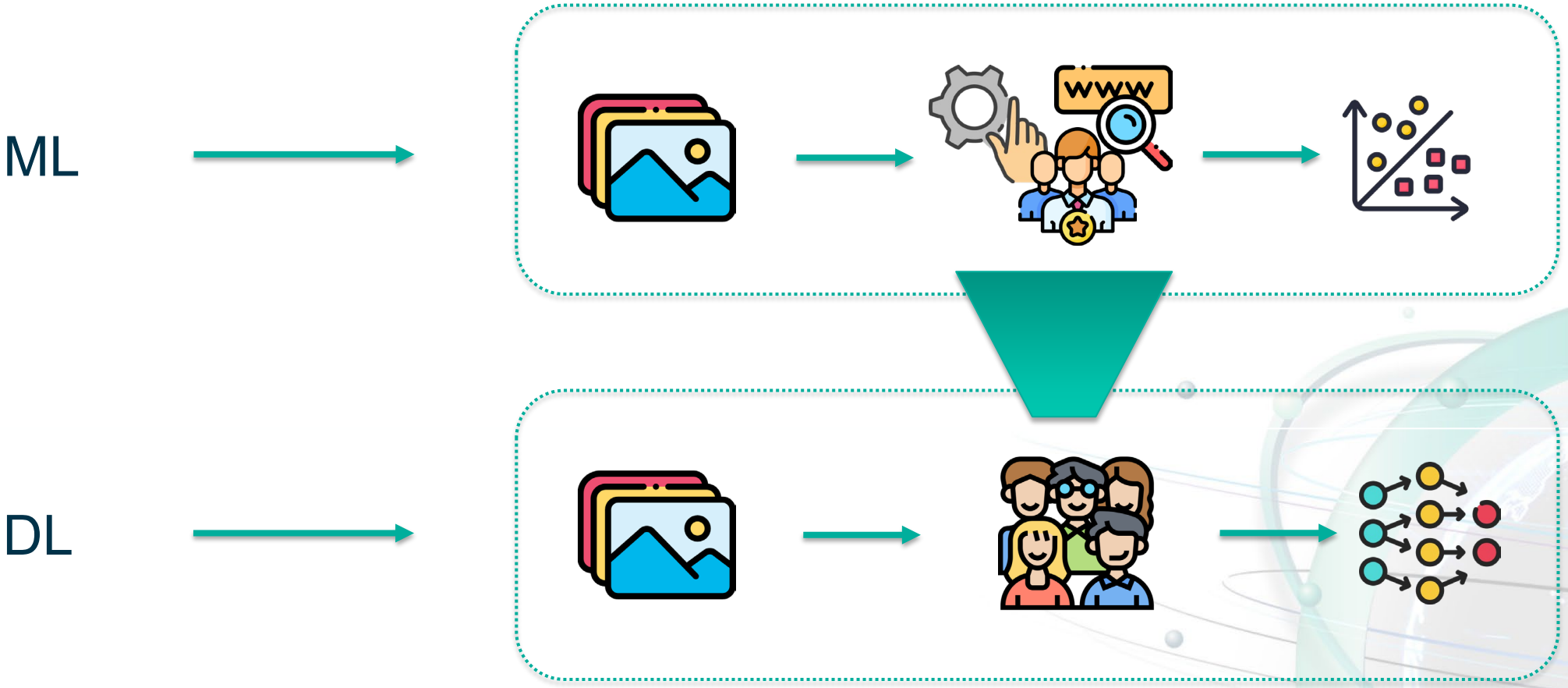
From Filters to Deep Learning

ML



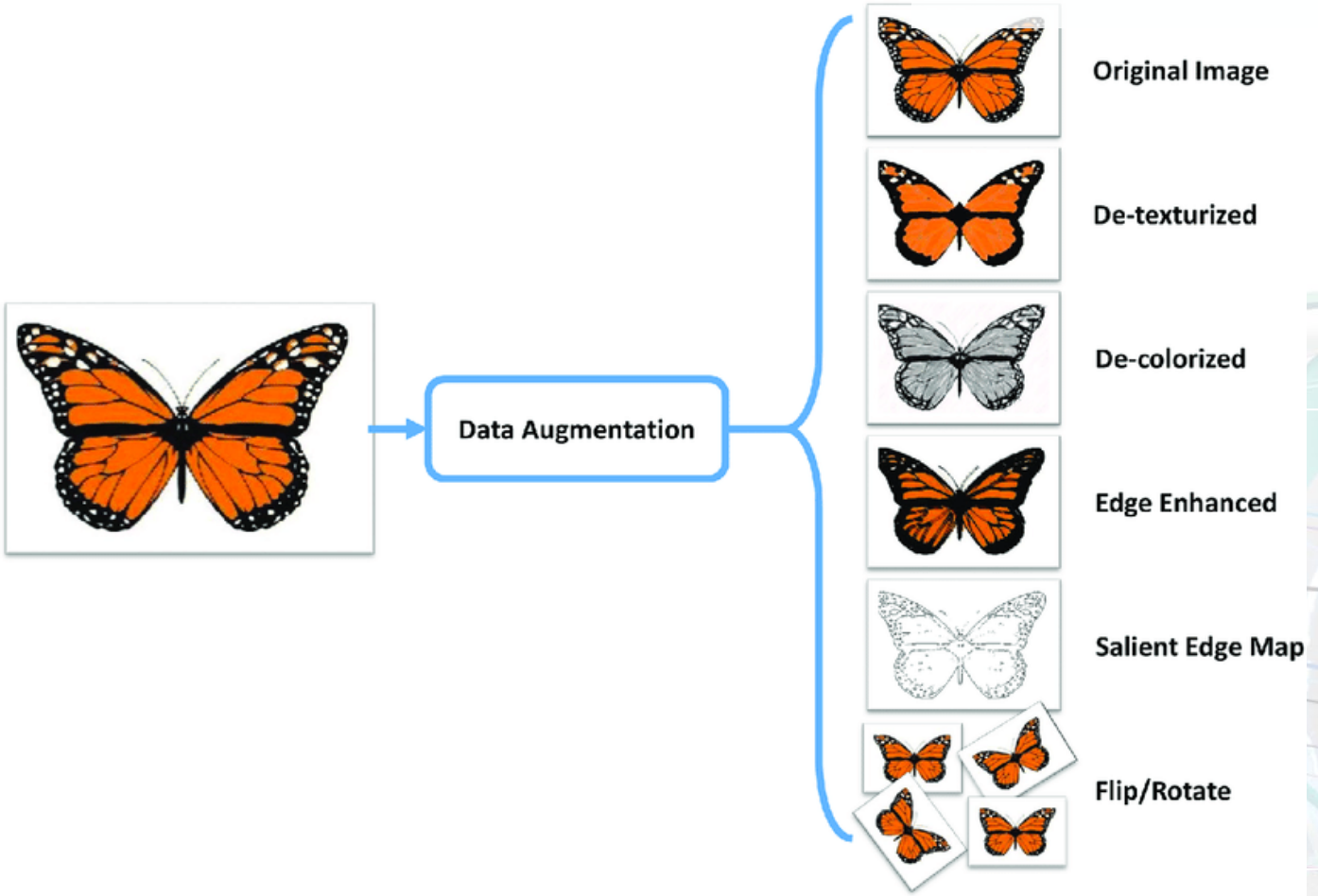
DL

From Filters to Deep Learning



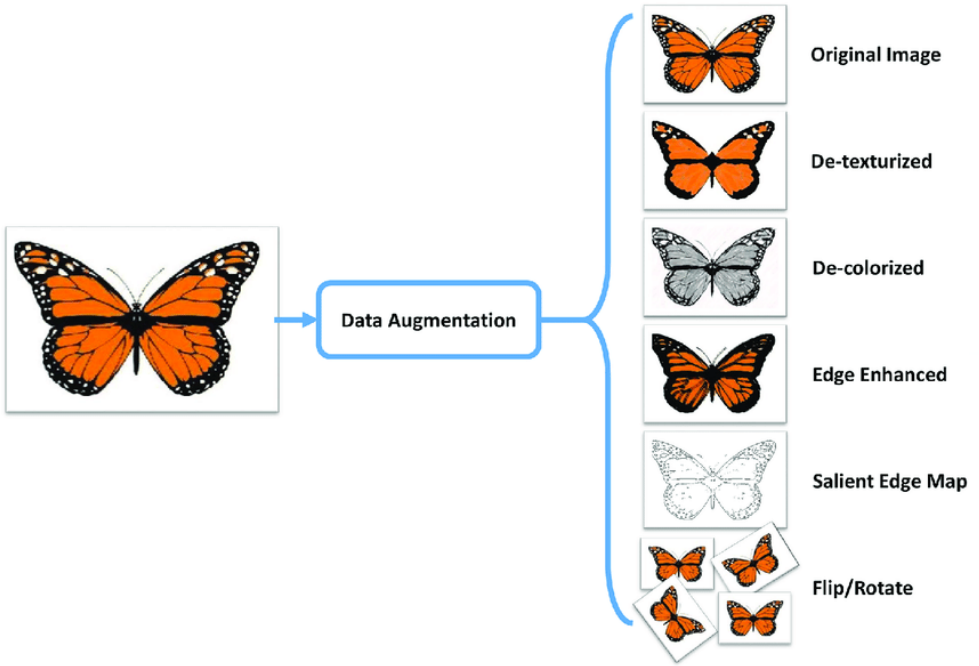
Data Augmentation

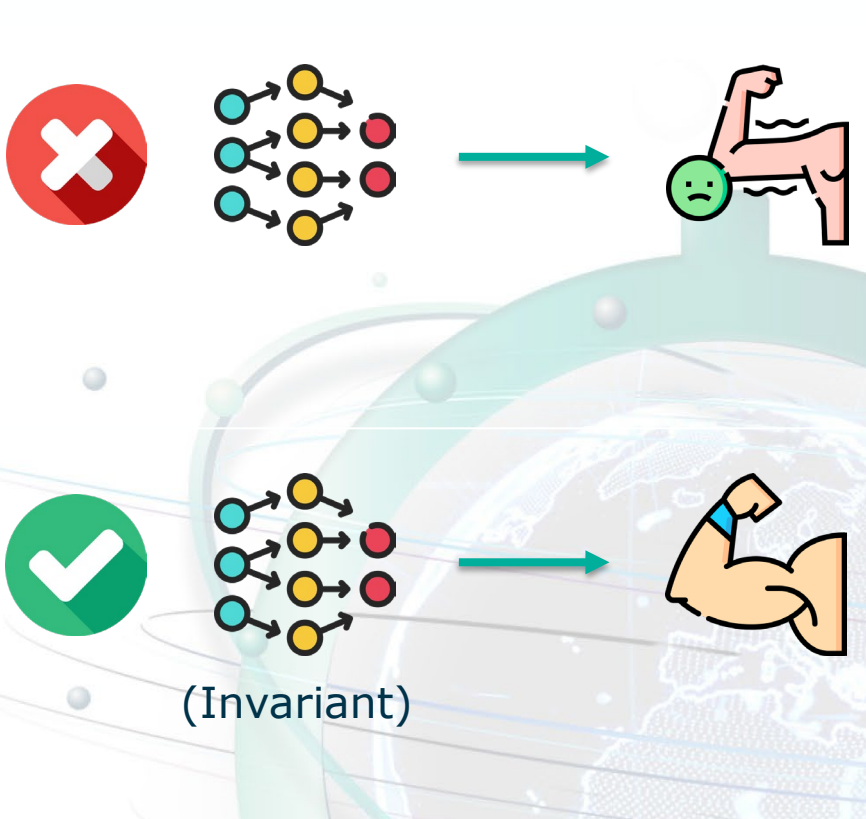
However, image processing is still fundamental in DL …



Data Augmentation

However, image processing is still fundamental in DL …

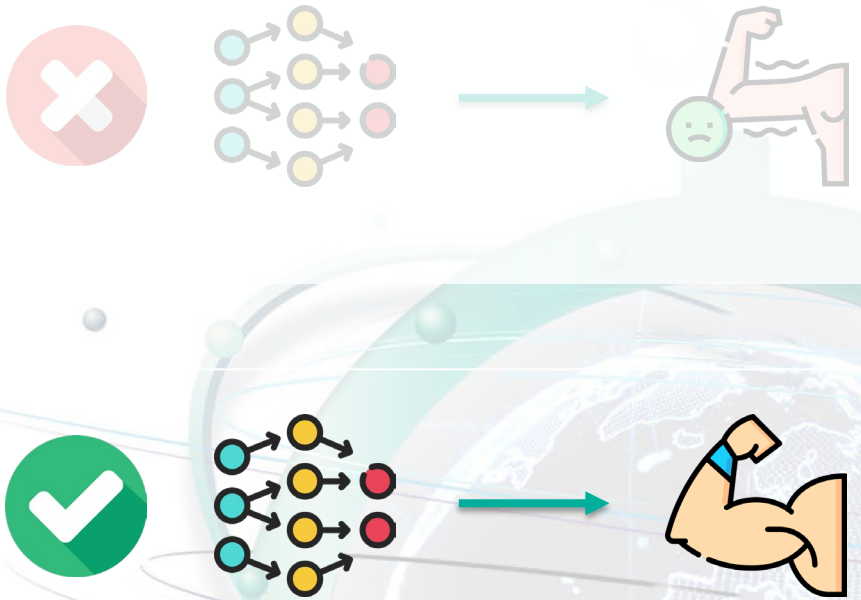




Data Augmentation

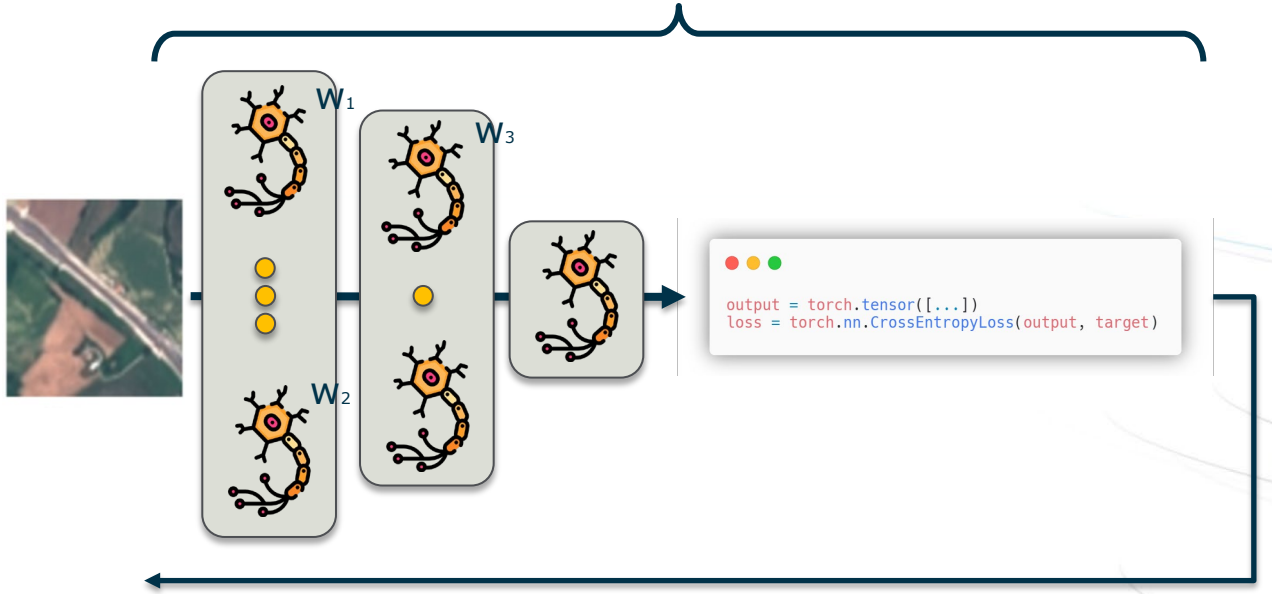
However, image processing is still fundamental in DL …

```python
import torch
from torchvision import transforms
from PIL import Image

image_path = "path/to/your/image.jpg"
image = Image.open(image_path).convert("RGB")

# Define the data augmentation pipeline
transform = transforms.Compose([
    transforms.RandomHorizontalFlip(p=0.5),
    transforms.RandomRotation(degrees=15),
    transforms.ColorJitter(brightness=0.2, contrast=0.2, saturation=0.2, hue=0.1),
    transforms.RandomResizedCrop(size=(224, 224), scale=(0.8, 1.0)),
    transforms.ToTensor()
])

# Apply the transformations
augmented_image = transform(image)
```



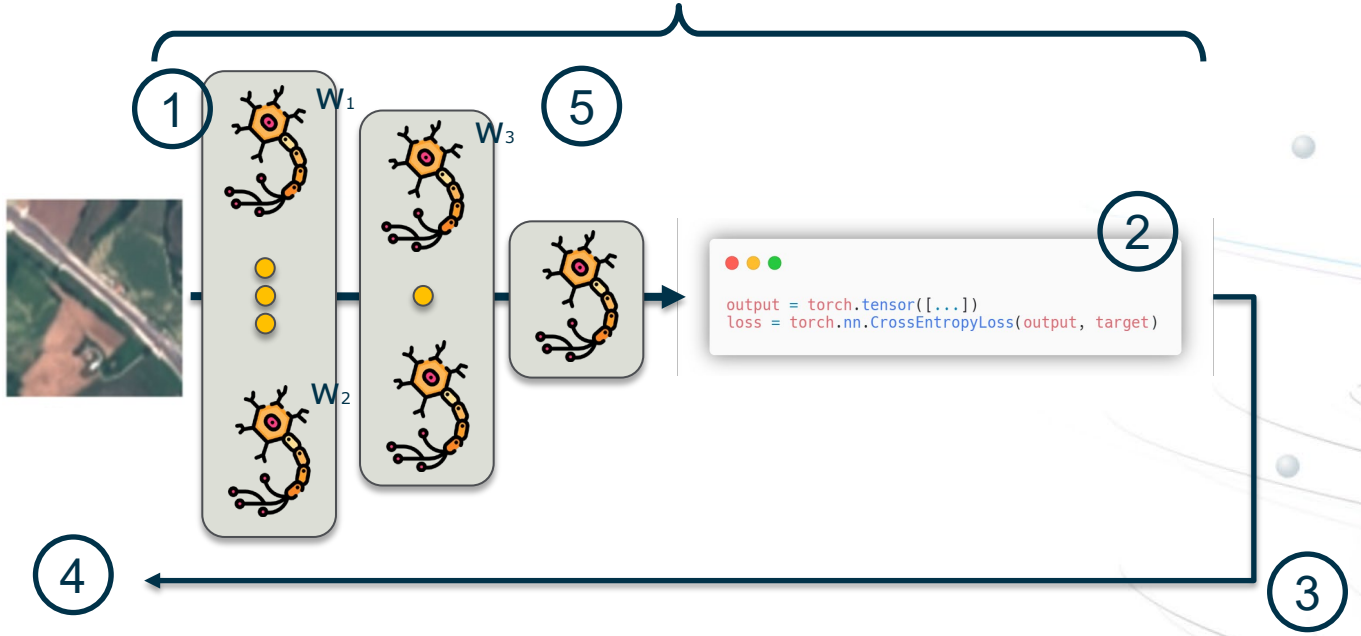
From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment



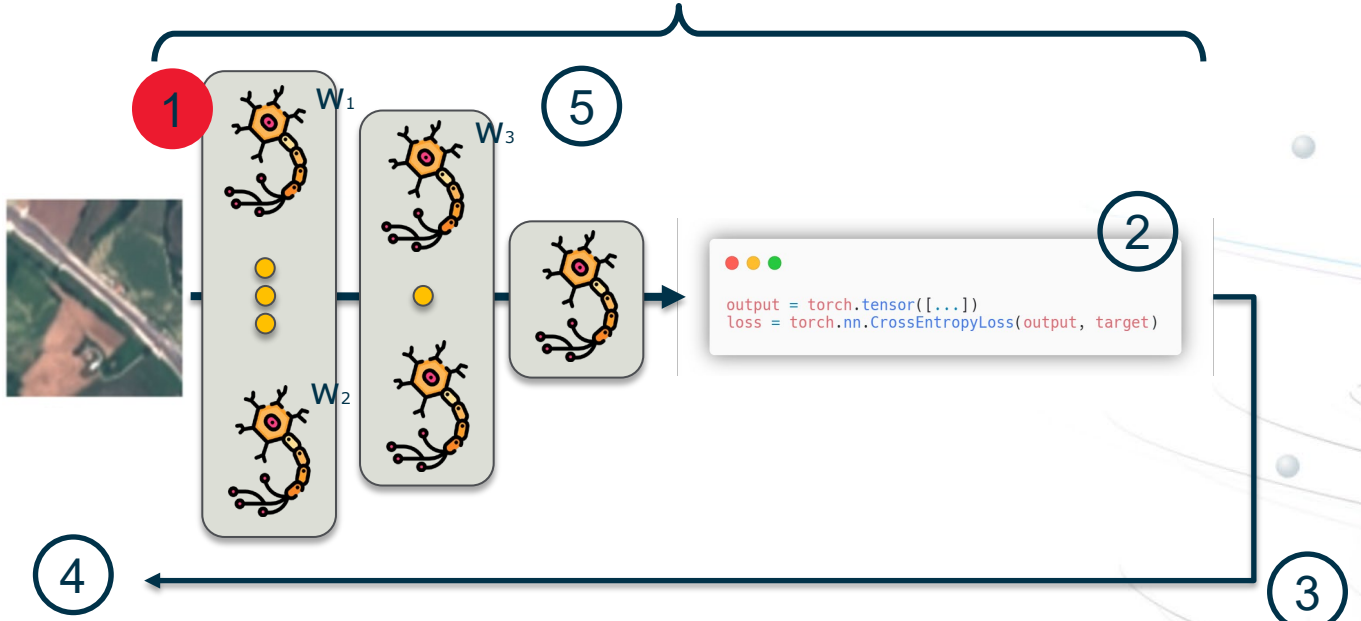
From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment

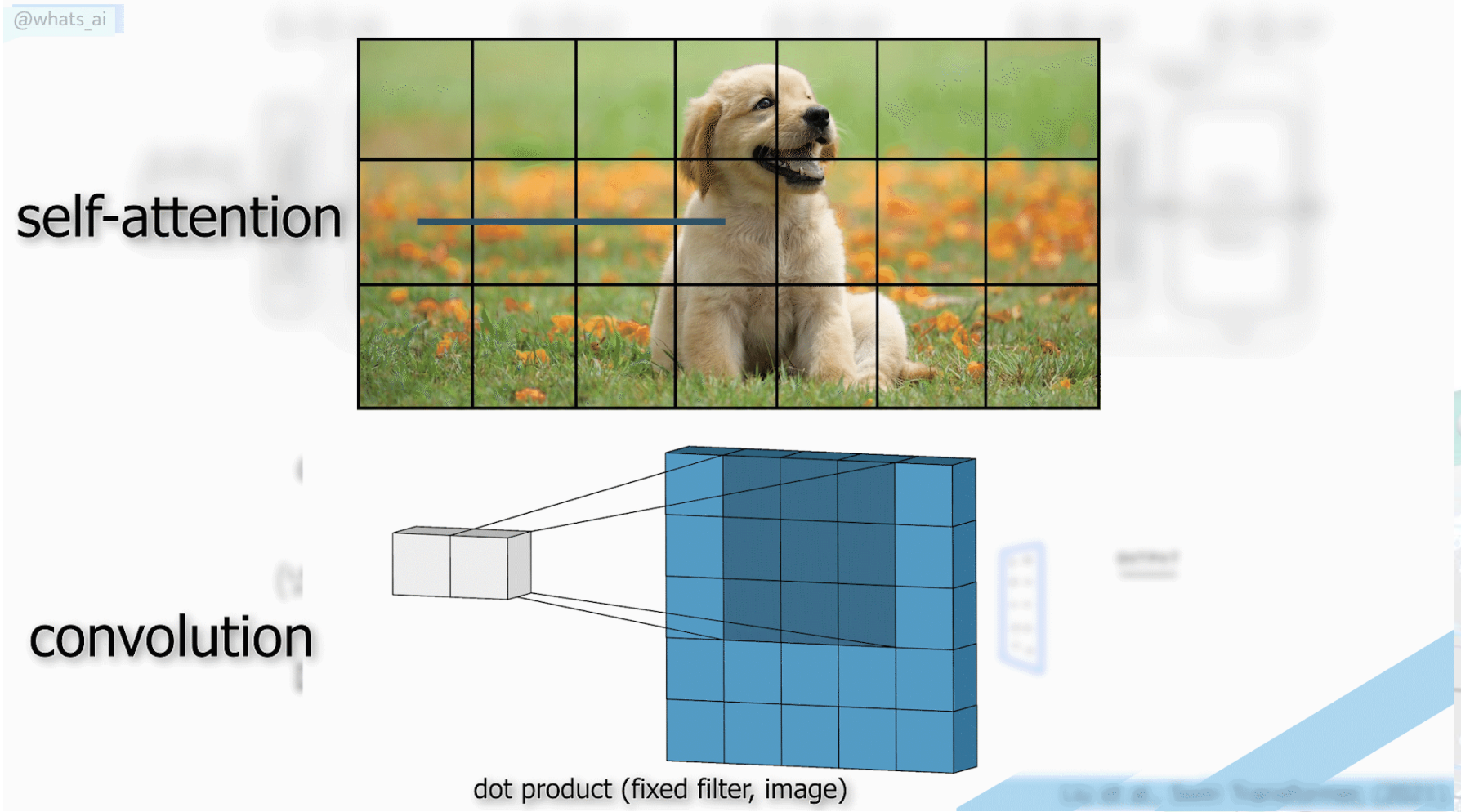


From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment

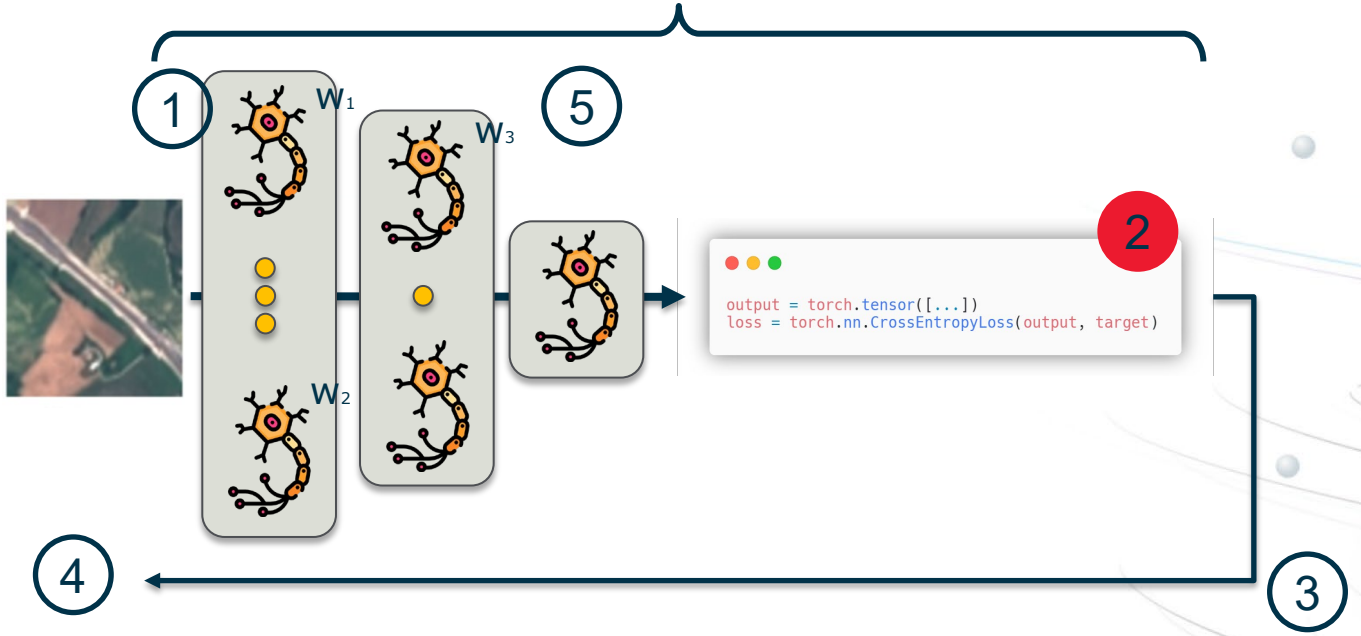


1 Layers & Neural Networks



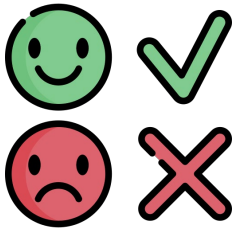
From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment



2 Forward Pass & Loss Function

How good or bad (accurate) is the prediction?

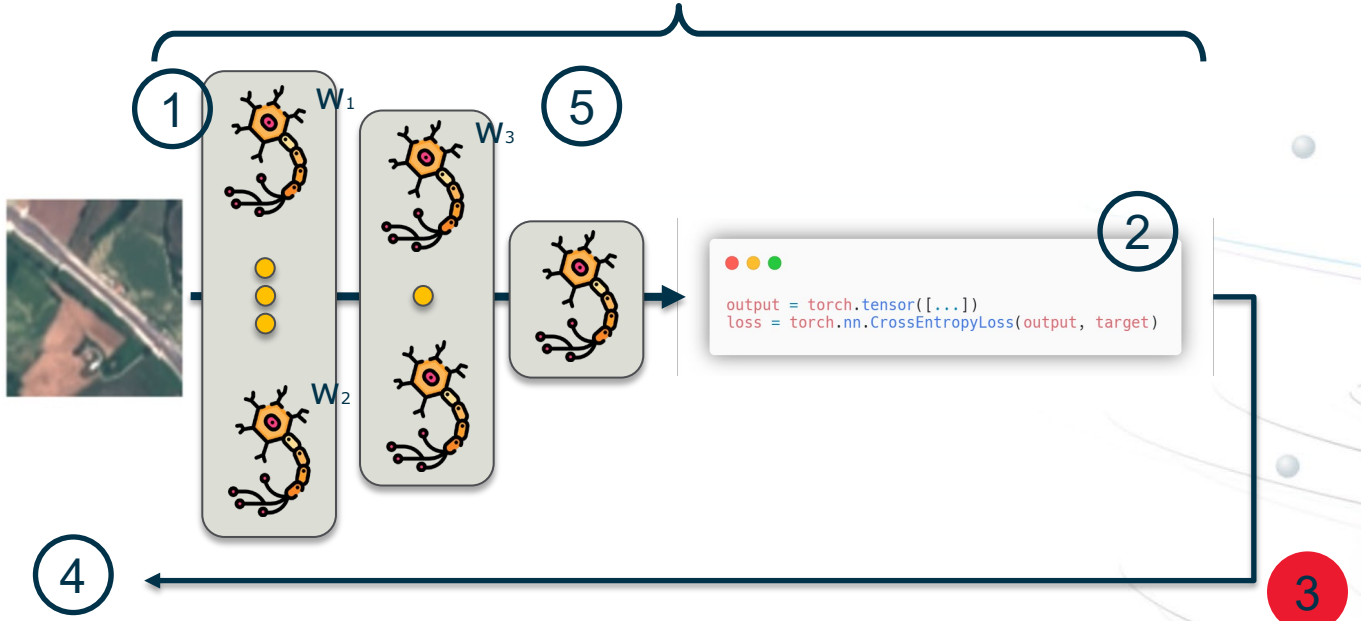


```
predicted_class = torch.argmax(output)
```

```
loss = torch.nn.CrossEntropyLoss(output, target)
```

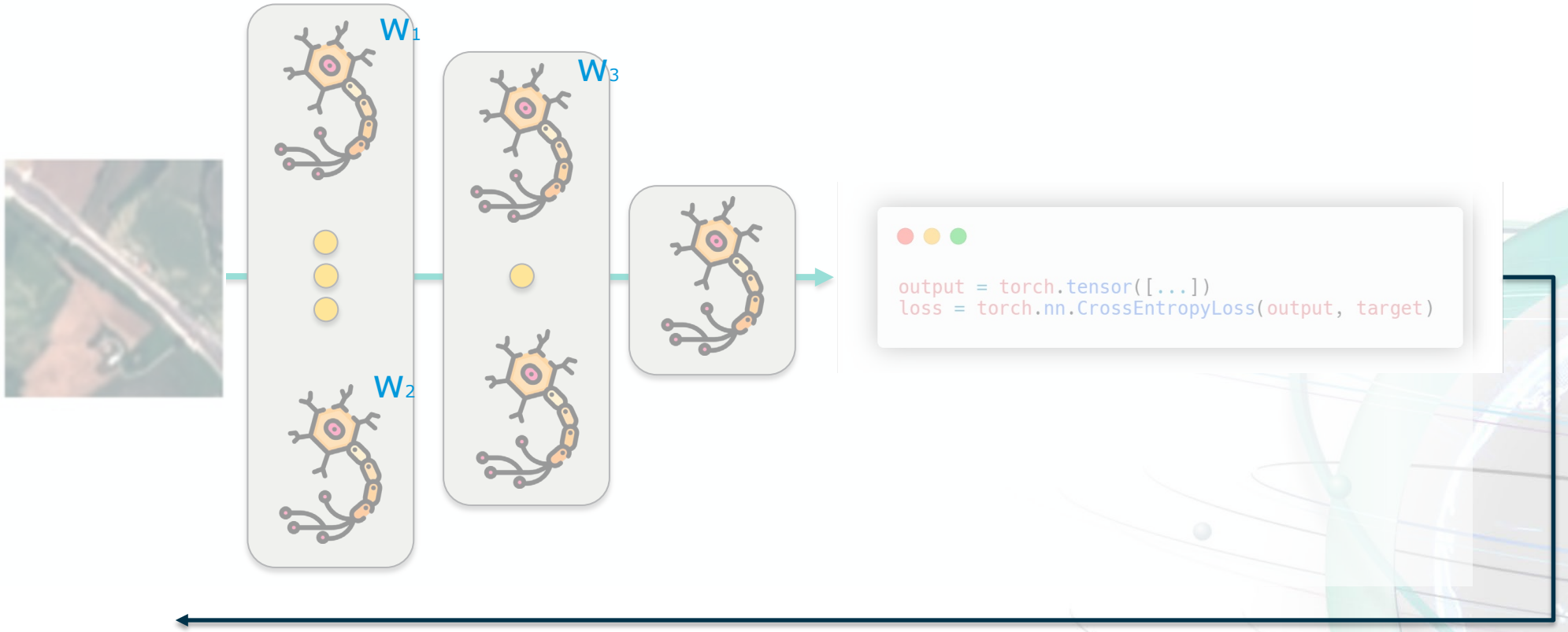
From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment

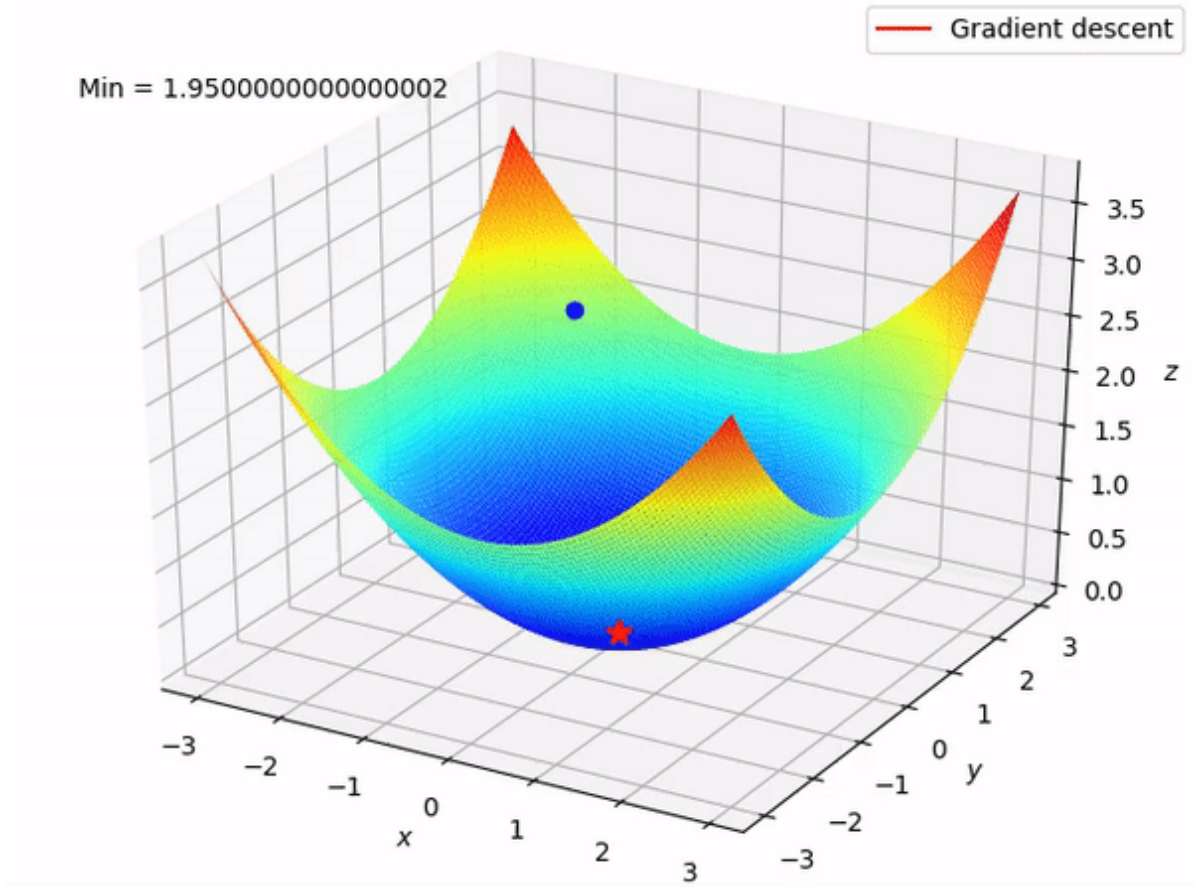


3 Backpropagation Algorithm & Optimizer

Once the model makes a prediction and we've calculated the loss, we need to tell the model *how to improve*.



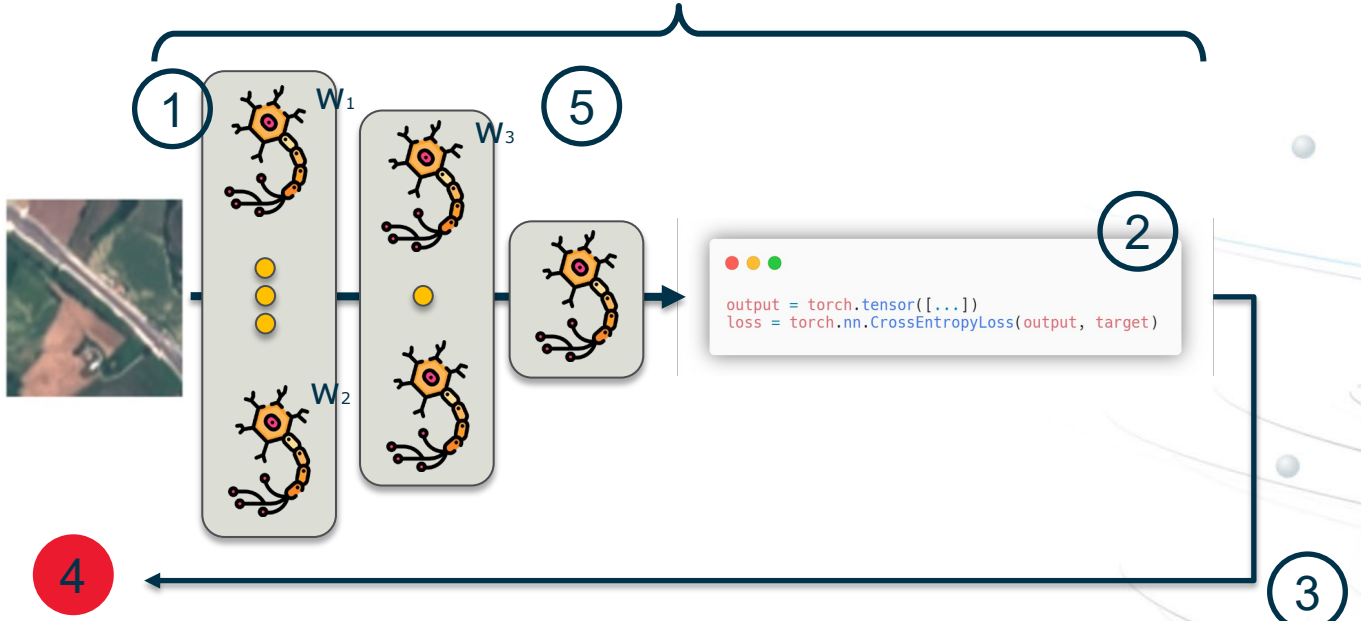
3 Backpropagation Algorithm & Optimizer



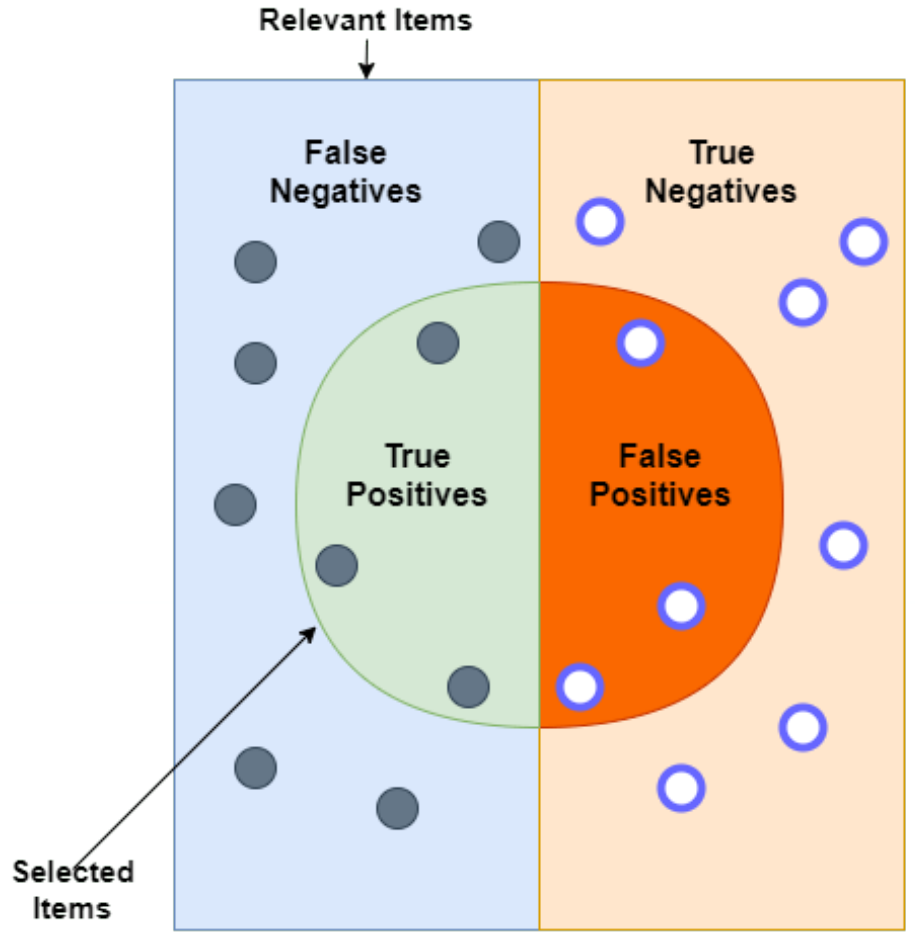

```python
import torch

# Simple model: 1 linear layer
model = torch.nn.Linear(2, 1)

# Loss function
loss_fn = torch.nn.MSELoss()

# Optimizer: you can switch SGD to Adam if needed
optimizer = torch.optim.SGD(model.parameters(),
lr=0.01)
# Input and target (1 sample, 2 features → 1 target)
x = torch.tensor([[1.0, 2.0]])
y = torch.tensor([[1.0]])

# Forward pass
output = model(x)
loss = loss_fn(output, y)

# Backward pass and weight update
optimizer.zero_grad()  # clear previous gradients
loss.backward()        # compute new gradients
optimizer.step()       # update model weights
```

From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment



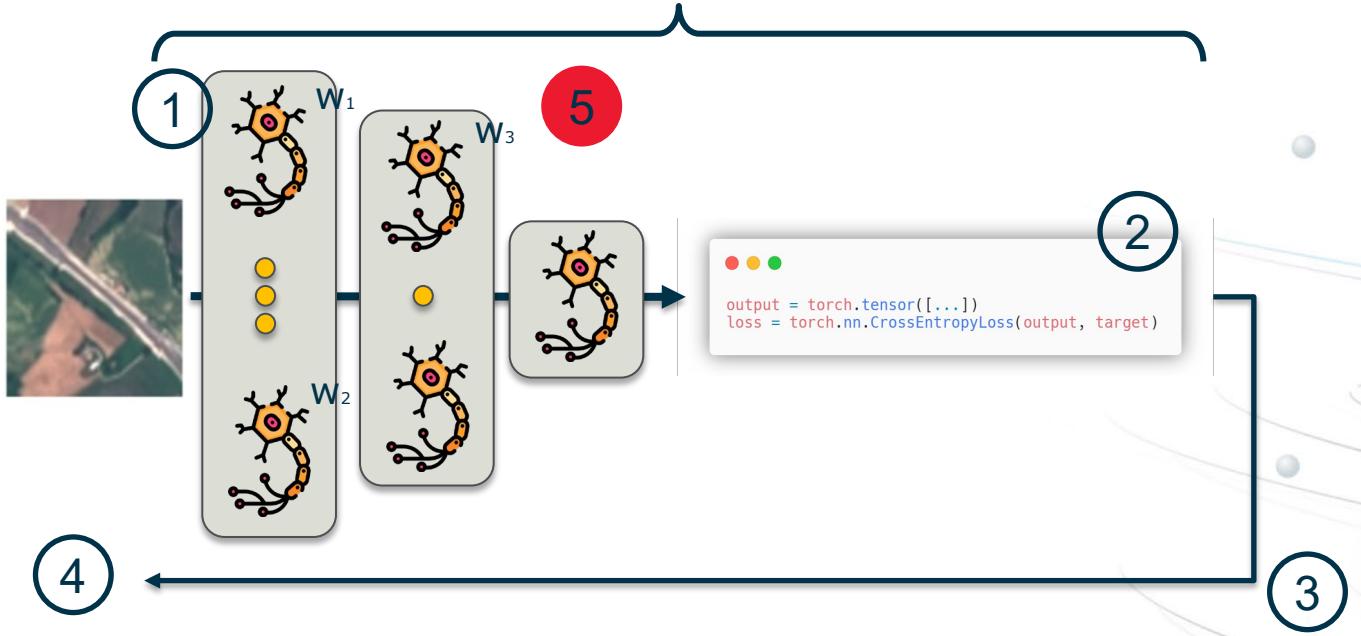
4 Evaluation Metrics



Accuracy	Predictions/ Classifications	$\frac{\text{Correct}}{\text{Correct + Incorrect}}$
Precision	Predictions/ Classifications	$\frac{\text{True Positive}}{\text{True Positive + False Positive}}$
Recall	Predictions/ Classifications	$\frac{\text{True Positive}}{\text{True Positive + False Negative}}$
F1	Predictions/ Classifications	$\frac{\text{2 * True Positive}}{\text{True Positive + 0.5 (False Positive + False Negative)}}$
IoU	Object Detections/ Segmentations	$\frac{\text{Pixel Overlap}}{\text{Pixel Union}}$

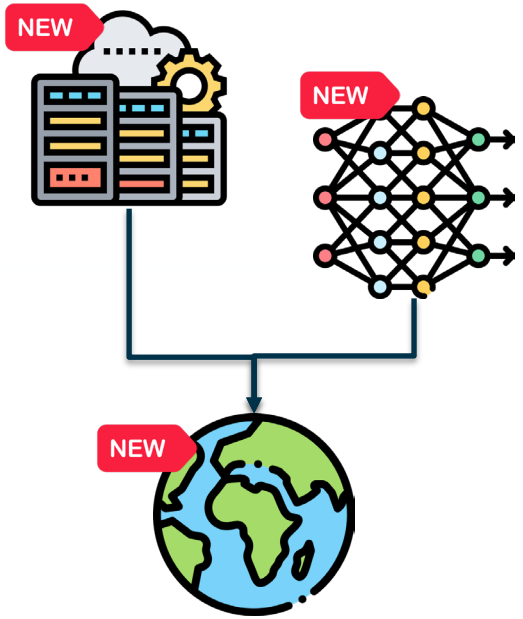
From ML to DL (pipeline)

Data Building | Data Processing | Strategy Selection | Training | Evaluating | Deployment

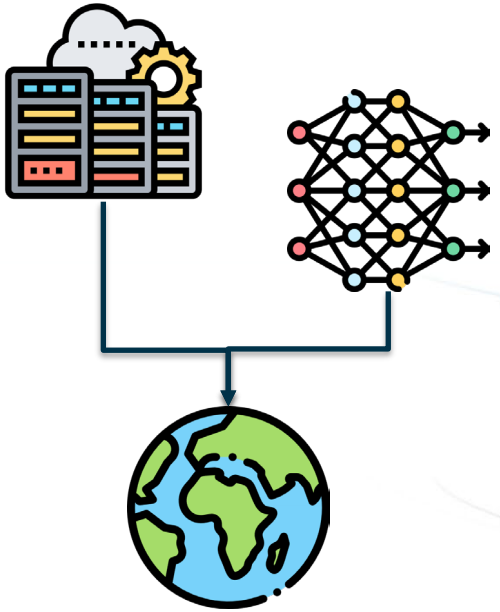


5 Pretrained & Finetuning Models

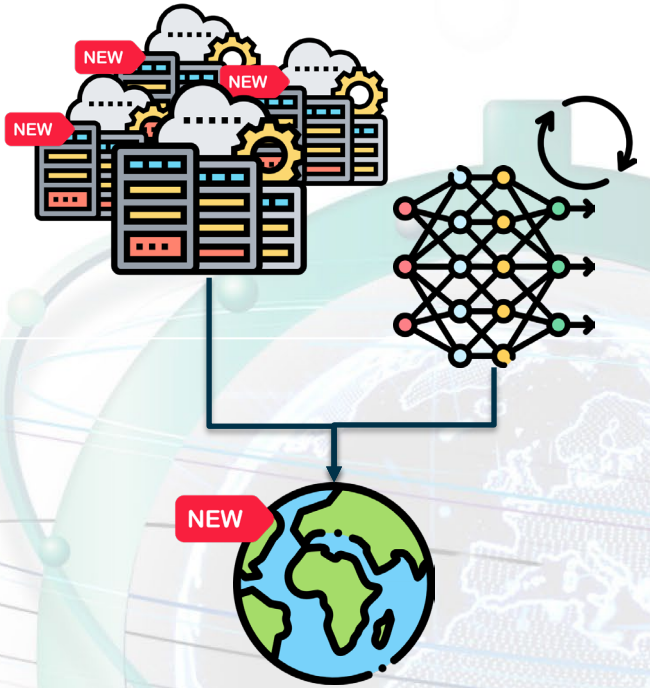
Training from scratch



Pretrained Model



Finetuning



5 Pretrained & Finetuning Models

```python
import torch
import torch.nn as nn
import torch.optim as optim
from torchvision import models

# Load a pre-trained ResNet18 model
model = models.resnet18(pretrained=True)

for param in model.parameters():
    param.requires_grad = False # ...? Let's have a look!!

# Replace the final fully connected layer to match our new task (e.g., 10 classes)
model.fc = nn.Linear(model.fc.in_features, 10)

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
model = model.to(device)

loss_fn = nn.CrossEntropyLoss()
optimizer = optim.SGD(model.fc.parameters(), lr=0.001, momentum=0.9)

model.train()
# Training cycle ....
```

… GoTo EuroSATClassifier.ipynb

Collaboration opportunities at Φ-lab

Shared mutual interest

Join the Φ-lab to explore disruptive ideas

as a Visiting Researcher (industry, academia),
Visiting Professor, Research Fellow, PhD, YGT, etc.

Funded

1. Φ-lab's Invitation To Tender on ESA-STARS
   - Foundation Models, Generative AI, QC4EO, Edge computing, Web 3.0, etc..
2. InCubed : partnership development of commercial products or services
3. Open Space Innovation Platform : co-funded research or researchers
4. EO Science4Society : no SOW, 100/200K, 6/18 months
5. ESA Technology Programmes like GSTP and TDE

Working with us it is easy

- Visiting Professor
- Visiting Researchers (Industrial and Scientific)
- ESA Research Fellowships
- ESA Co-funded PhD
- ESA Early Graduate Traineeships (EGT), Internships, National trainee

Join ESA Φ-lab through CIN

The **Collaborative Innovation Network (CIN) by ESA Φ-lab**, provides to leading researchers and University Professors the **opportunity to** join ESA Φ-lab and be actively involved in **accelerating the future of Earth Observation with ESA**.

The ESA Φ-lab CIN aims to:

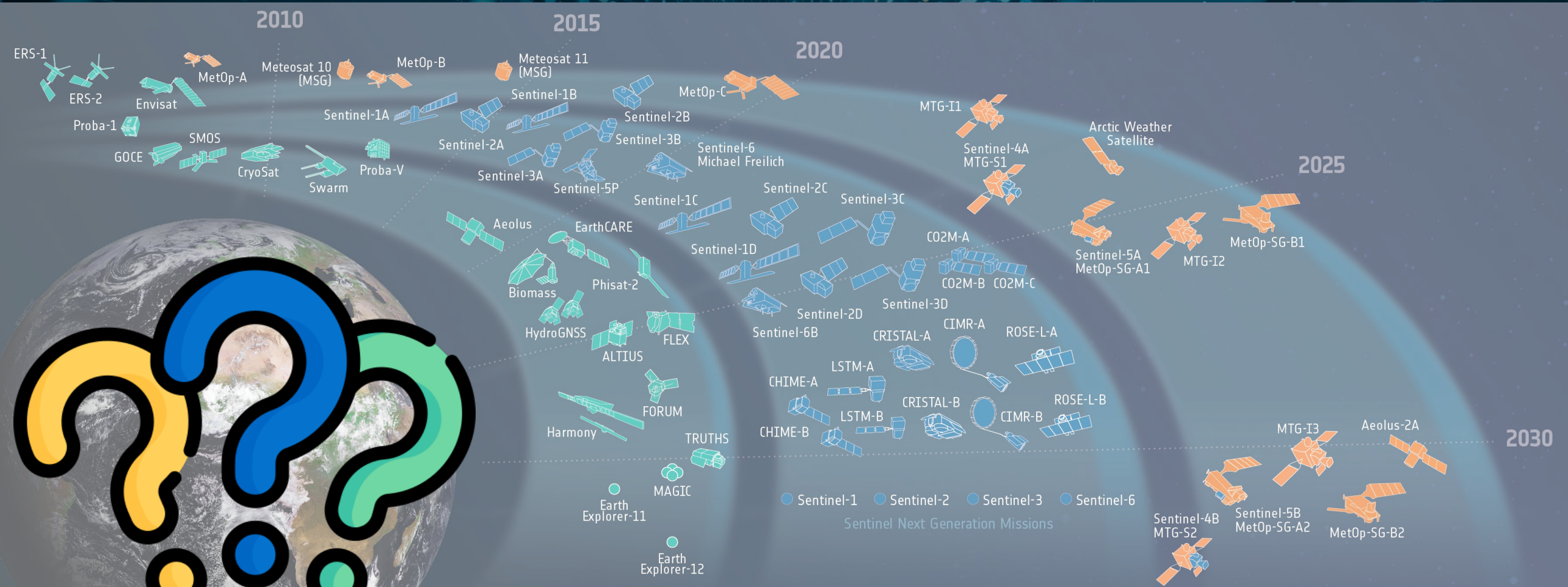
- Establish a global network through which researchers and innovators can **JOIN ESA Φ-LAB**
- **Promote knowledge** sharing and develop groundbreaking EO solutions

Check the open calls

Follow CIN on Linkedin

What to go deeper into Φ-lab disruptive innovation?

Don’t miss our spotlight session at Big Data from Space
(Day 2 – October 2nd at 17:50)



Science

Copernicus

Meteorology

ESA – European Space Agency
Internal Research Fellow, Φ-lab Division
Earth Observation Climate Action, Sustainability and Science Department (EOP-S)
Directorate of Earth Observation Programme
lorenzo.papa@esa.int | www.esa.int | https://philab.esa.int