

# Deep Learning

## Forest in HPC

Dorothea Sommer | 13.12.2022

# Content

- Deep learning use case: **Tree Species Classification with a Neural Network**
  - Concise background
  - Tailor this to our use case
  - Hands-on: inference
- **Not** a concise introduction to Deep Learning  
(checkout neural networks: [3Blue1Brown](#), machine learning: [Andrew Ng](#))

# Learning Paradigm

## Supervised Learning

*“learning with teacher”*

### Data

Observations  $\underline{x}_1, \dots, \underline{x}_n$   
Labels  $y_1, \dots, y_n$

### Aim

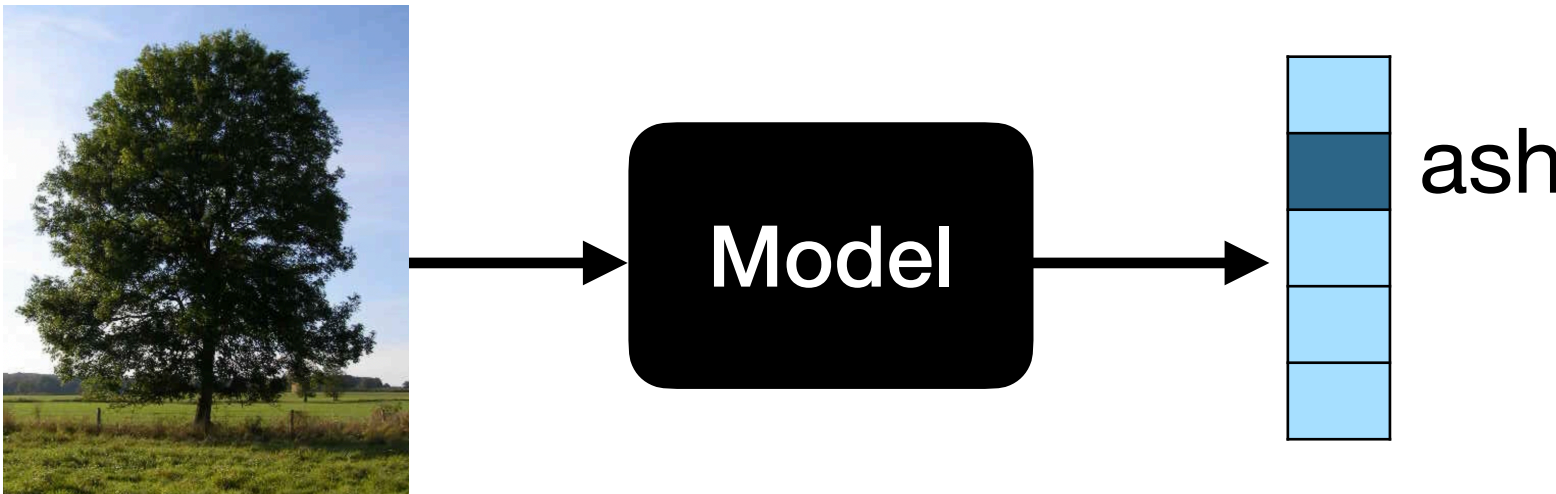


predict label of observation  
**regression, classification**

## Our Use Case

*tree species classification*

$\underline{x}_i$  tree photo  
 $y_i$  corresponding tree species



**classification**  
predict tree species  
given photo of the tree

# Deep Learning

## Supervised Learning

*learning with teacher*

### Data

Observations  $\underline{x}_1, \dots, \underline{x}_n$

Labels  $y_1, \dots, y_n$

### Aim



predict label of observation

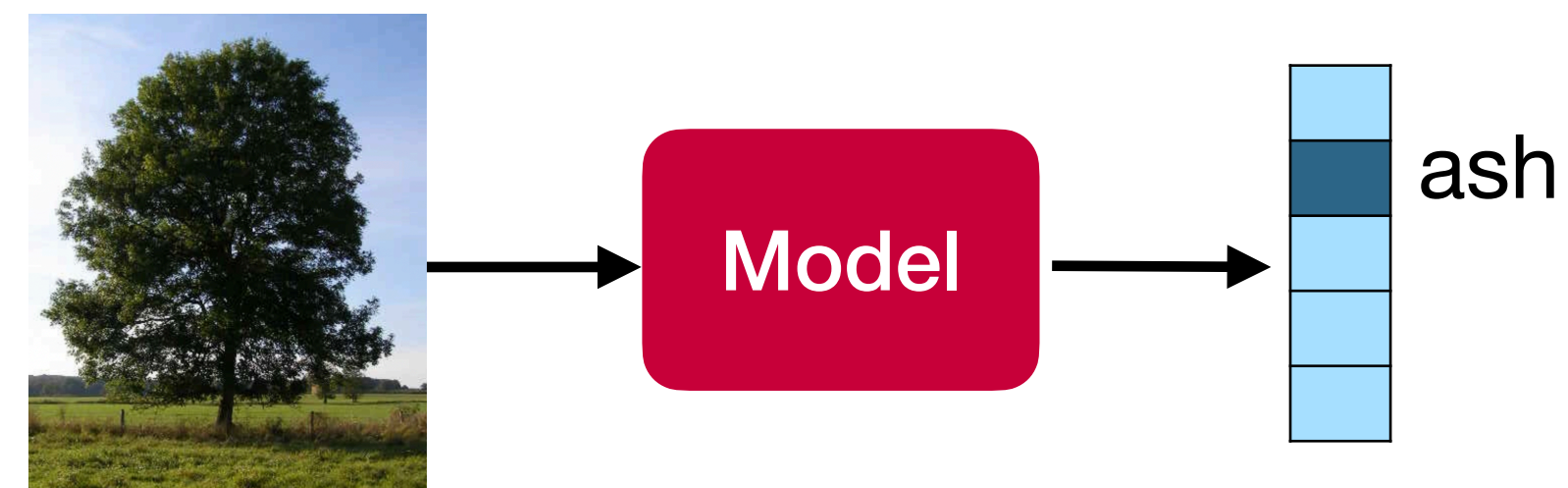
**regression, classification**

## Our Use Case

*tree species classification*

$\underline{x}_i$  tree photo

$y_i$  corresponding tree species



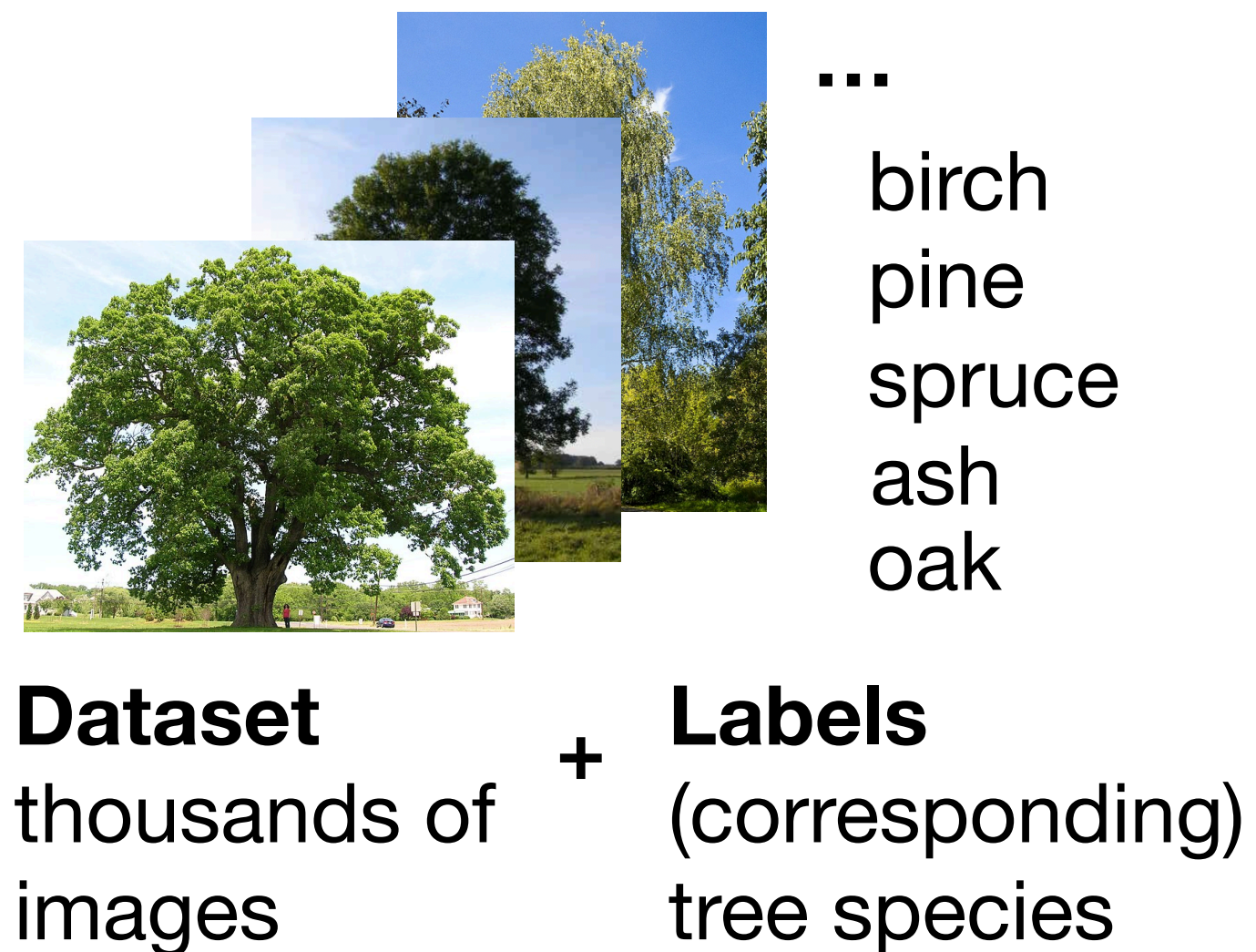
**classification**

predict tree species  
given photo of the tree

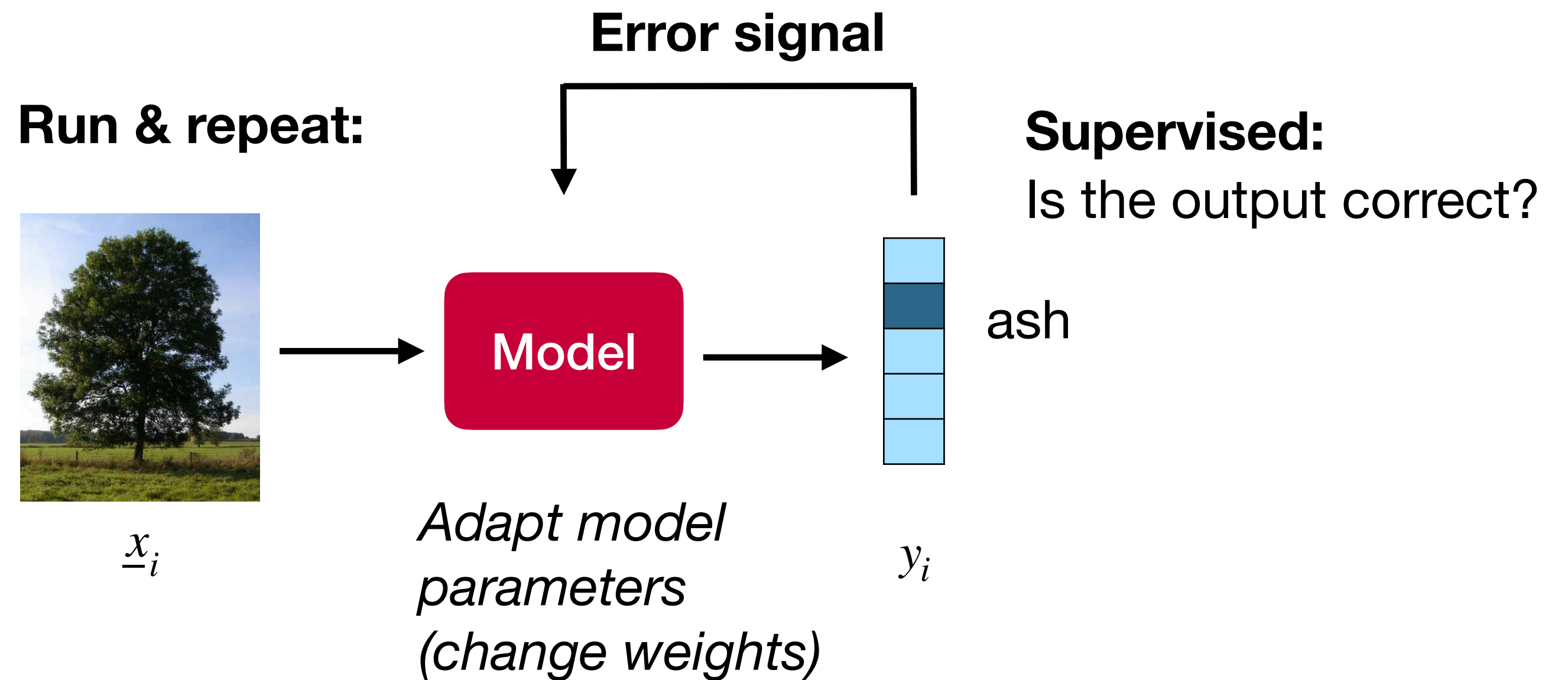
Neural  
Network

# Deep Learning in a Nutshell

## Training



## Run & repeat:

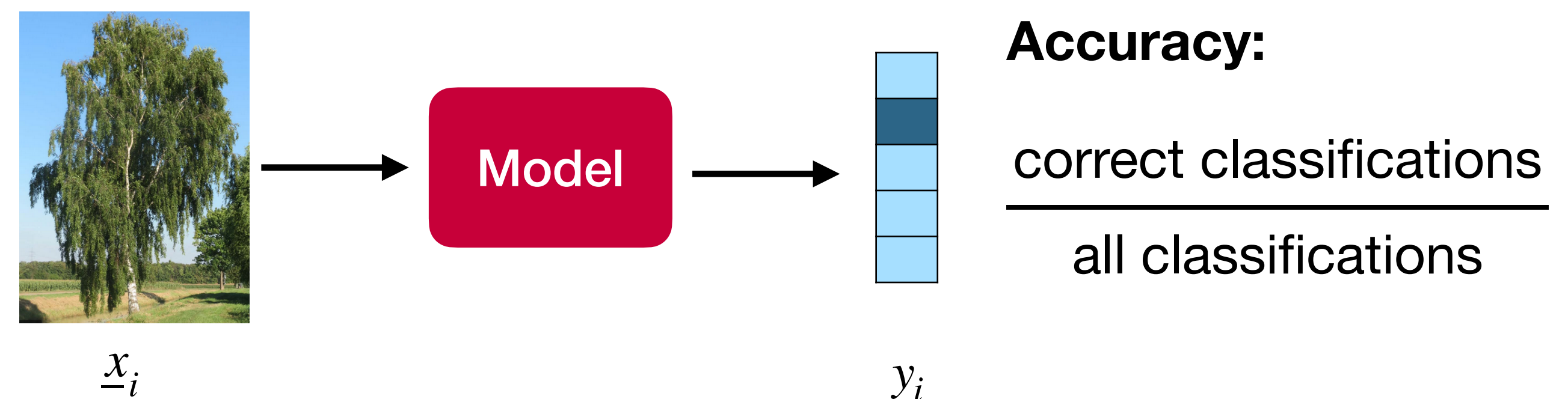


## Testing

How well does our model generalise?

Dataset of *unseen* images

## Inference:

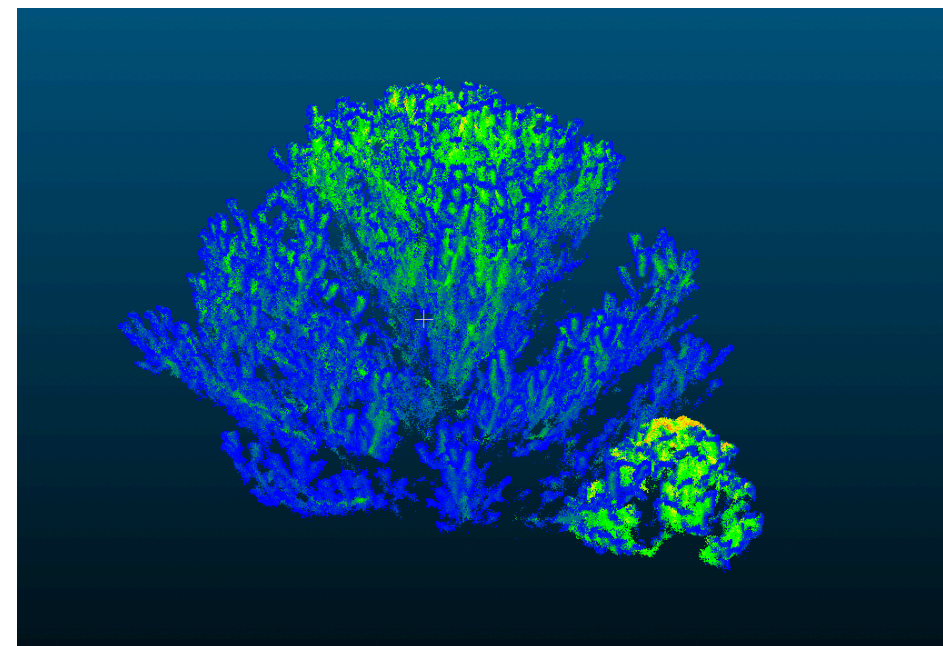




# Unboxing the Model

## What do we know about our input data?

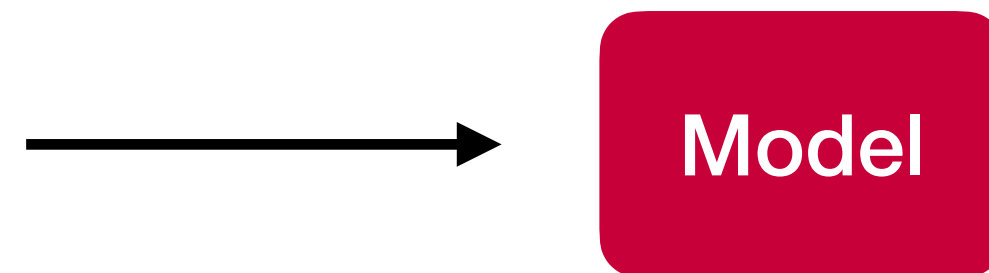
In **general**: the type of neural network depends on the **input data type**



3D point clouds

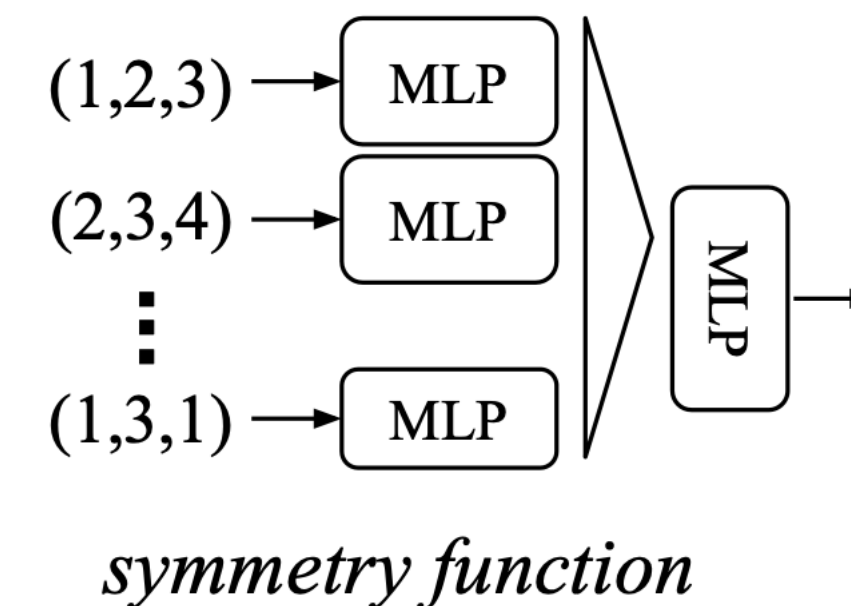
1. **unordered** set of points, a list of (x, y, z)
2. **invariance** under rigid transformations

$$f(x_1, \dots, x_n) \approx g(h(x_1), \dots, h(x_n))$$



$$h: \mathbb{R}^N \rightarrow \mathbb{R}^K \text{ neural network}$$

$$g: \underbrace{\mathbb{R}^K \times \dots \times \mathbb{R}^K}_n \rightarrow \mathbb{R} \text{ symmetric function (e.g., max pooling)}$$

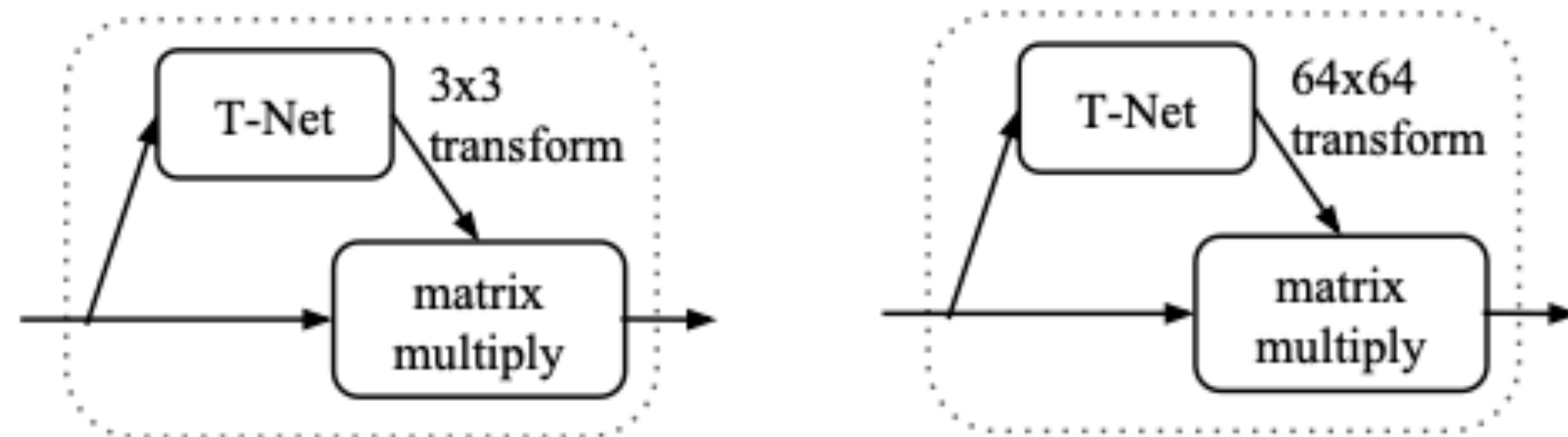
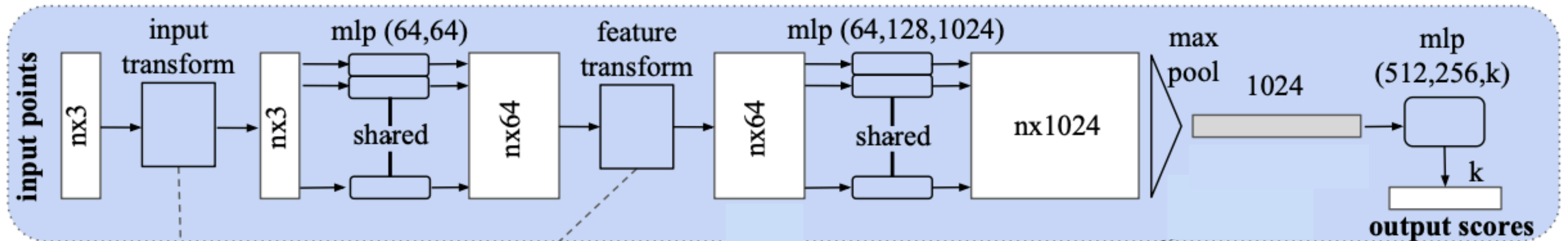


Qi et al. (2017) **PointNet**: Deep Learning on Point Sets for 3D Classification and Segmentation

# Unboxing the Model

## PointNet Architecture

*Classification Network*

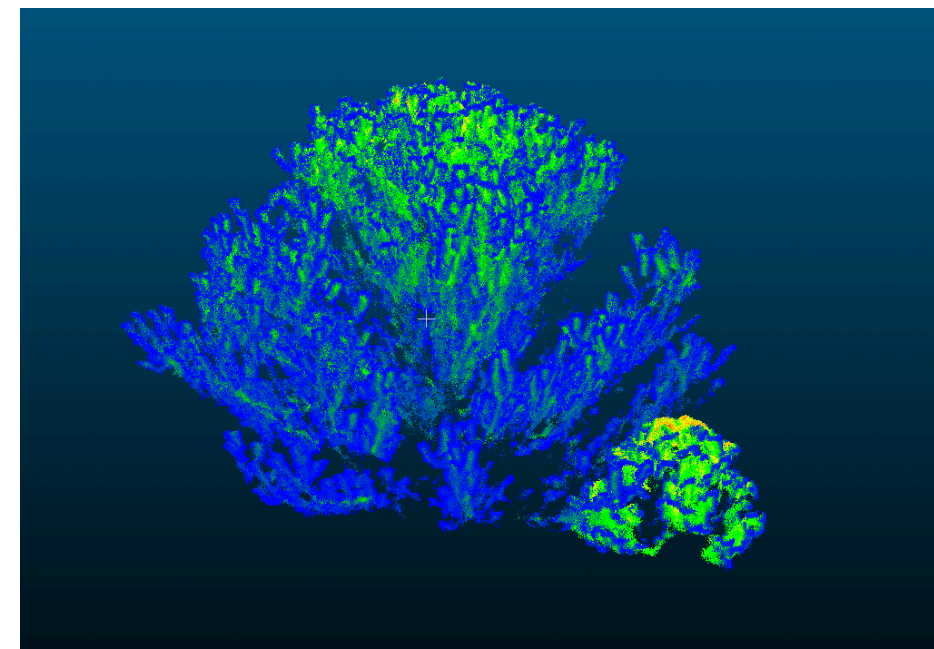


**T-Net** as a learned  
affine transformation matrix

Look into the paper for more details.

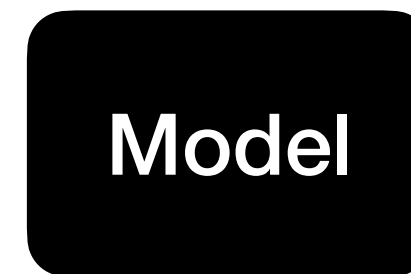
Qi et al. (2017) **PointNet**: Deep Learning on Point Sets for 3D Classification and Segmentation

# Let's switch to the code...

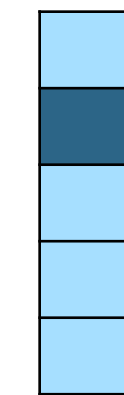


**3D point clouds**

Thanks to our Hiwi  
**Ali Doosthosseini**  
for generating this data!



**PointNet**



black oak

**classification**  
predicting tree species

**Any questions** up to this point?