



Computer vision in the new era of Artificial Intelligence and Deep Learning

Visión por computador en la nueva era de la Inteligencia Artificial y el Deep Learning

Rubén Usamentiaga*, Alberto Fernández°

***Universidad de Oviedo**

°TSK

Gijón

5 – 16 Abril 2021



<https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course>



Learning

Learning

□ How computer learn

- ▣ In supervised learning a model is build from labeled samples
 - The response under new inputs is generated using the model
- ▣ As the problem becomes more difficult, more samples are needed

Learning

- Learning based on the scientific method
 - ▣ Propose an hypothesis
 - Data follows a linear model: the output is linearly related to the input
 - ▣ Adjust the model using a portion of the samples
 - Model training
 - ▣ Evaluate the model using a different portion of the samples
 - Model evaluation

Learning

□ Supervised learning

▣ Requires

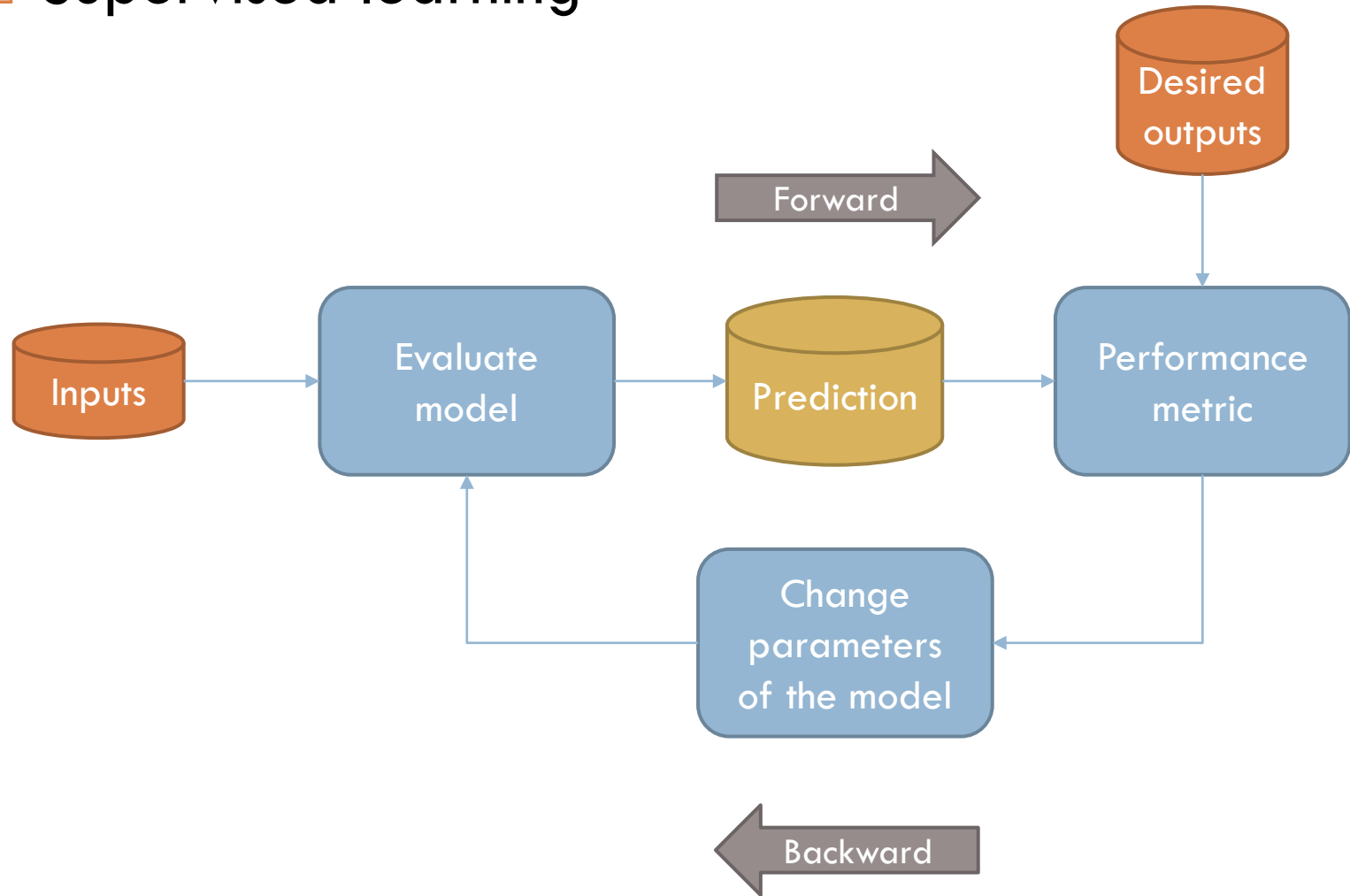
- Dataset: input data paired with desired outputs
- Model: a function that given an input produces an output

▣ Learning is just parameter estimation

- Fit the model to the dataset

Learning

□ Supervised learning



Learning

□ Training

```
for epoch in range(N):  
    # Forward pass: Compute prediction by passing the input to the model  
    prediction = model(input)  
  
    # Evaluate the model  
    loss = criterion(prediction, desired_output)  
  
    # Perform a backward pass, and update the model  
    loss.backward()  
    optimizer.step()
```

Training = Minimize loss function

Learning

□ Training result

```
prediction = trained_model(new_input)
```

An accurate model predicts the
expected result

Learning

□ Training using Stochastic Gradient Descent (SGD)

```
for epoch in range(N):  
    for batch in range(M):  
  
        # Forward pass: Compute prediction by passing the input to the model  
        prediction = model(input)  
  
        # Evaluate the model  
        loss = criterion(prediction, desired_output)  
  
        # Perform a backward pass, and update the model  
        loss.backward()  
        optimizer.step()
```

Other popular training method:
Adam (adaptive moment estimation)