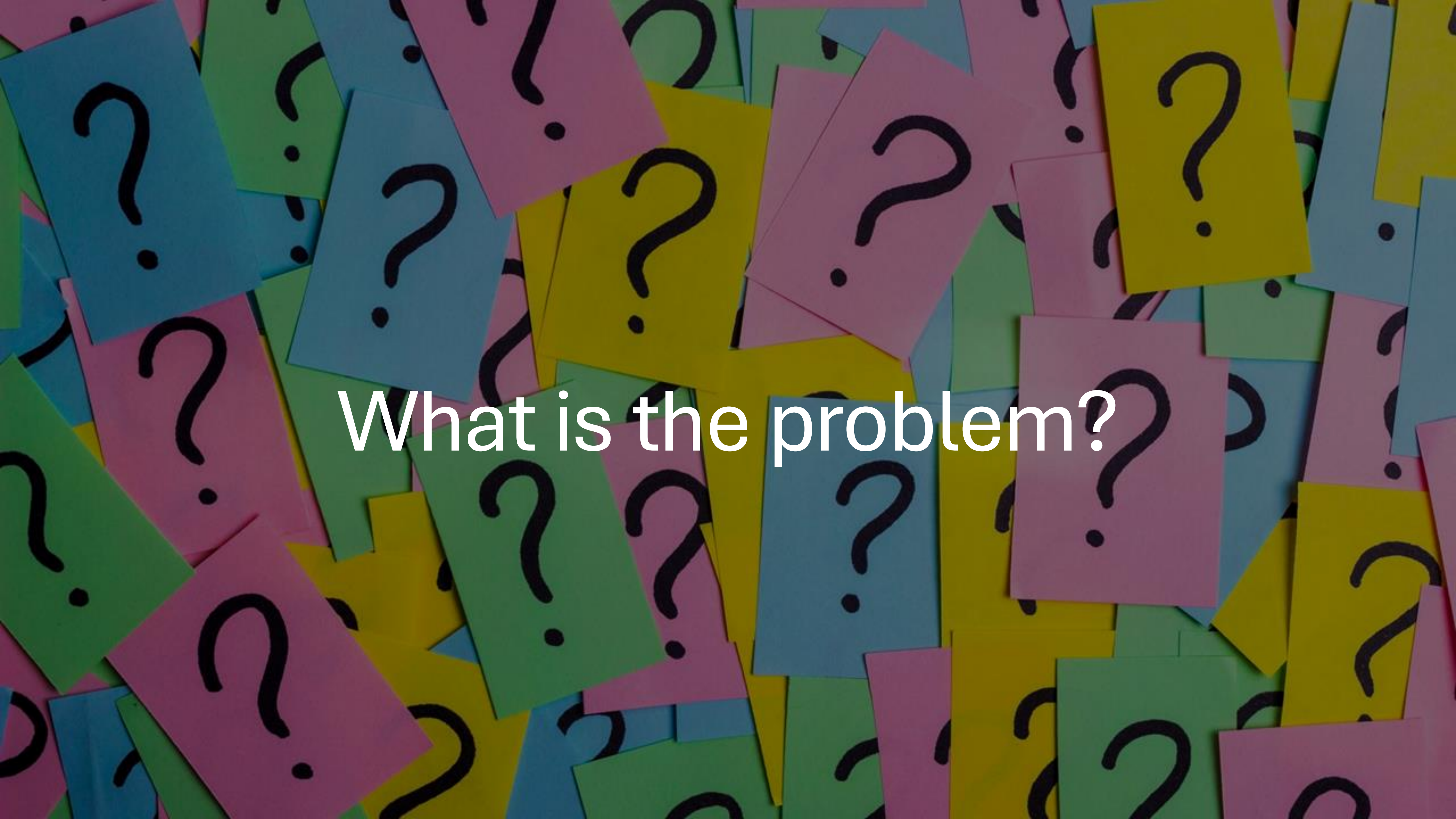
Dent.py: Training Dental Models from Zero to Hero

Part 2 of 5

Formulating the Problem

Objectives

- Understanding the Problem and Tasks
  - Understand the needs and requirements of problem formulation
  - Explore different deep learning tasks, including classification, semantic segmentation, and instance segmentation.
- How Models Learn
  - Dive into how models learn by understanding losses and optimizers.
  - Evaluate model performance through the introduction of key metrics.

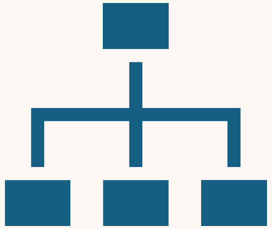


What is the problem?

Problem Formulation

- How to define your problem?
  - Define the Objective
  - Understand the Input Data
  - Establish metrics for success
  - Select the Learning Paradigm
  - Identify Constraints
  - Set Evaluation Criteria

Define the Objective



What is the task?

Classification

Segmentation

What are the desired outputs?

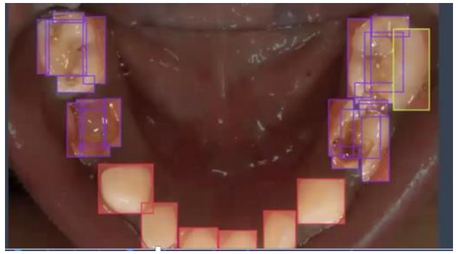
Predict lesion presence from X-Ray scans

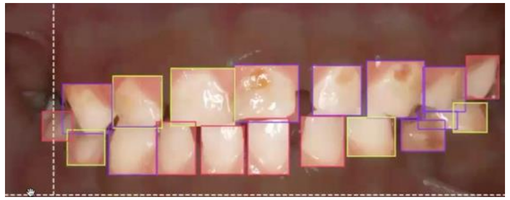
Calculate volume of segmentation masks from CBCT volumes

Recap: Tasks

Classification

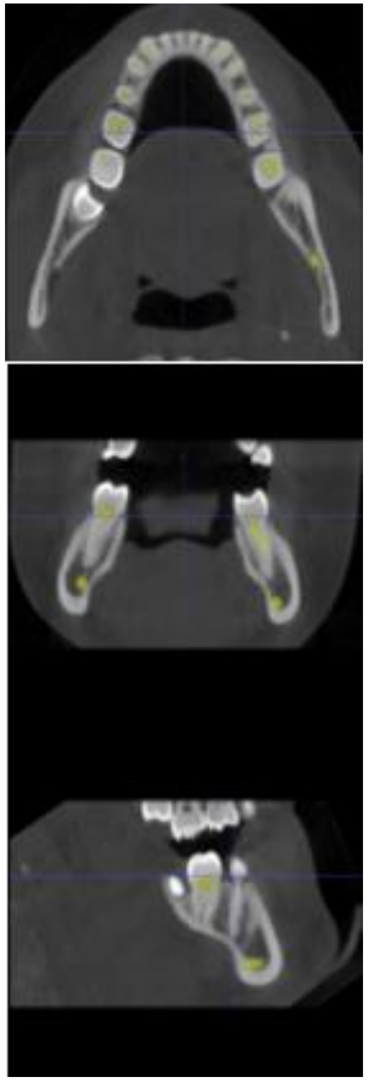
- Involve categorizing medical images or specific regions within them into predefined classes or labels
- Examples
  - Caries Detection
  - Lesion Classification
- Task Definition
  - Single Output ➔ Yes or No
    - Does this image contian caries?
  - Multiple Outputs ➔ Mild, Moderate, Severe
    - What is the level of severity of caries in these images?





Segmentation

- Involve partitioning an image into distinct regions or structures
- Examples:
  - Hard Tissue Segmentation
  - Pulp Instance Segmentation
- Task Definition
  - Single Output → Matches input size (Semantic)
    - Detect all lesions in the mouth
  - Multiple Outputs → Matches input size (Instance)
    - Detect all lesions in the mouth with their types respectively



Understand the Input Data

What data is available?

Images (2D X-rays, 3D CT, Intraoral RGB)

Text (reports, diagnosis)

Is the data labeled or unlabeled?

3D CBCT scans with annotated segmentations.

2D intraoral images with bounding boxes

Success Metrics

Map Inputs to Outputs

Define the relationship between input data and output label

Example

- Input: CBCT scan → Output: Semantic segmentation of pulp and IAN.

Choose Appropriate Metrics

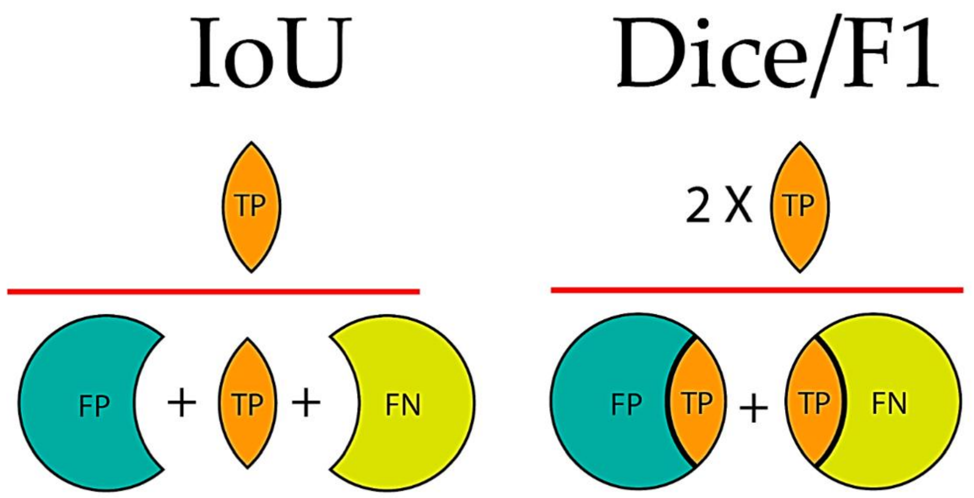
Measures Input-Output mapping success

Example

- Accuracy of segmentation of pulp
- Precision of lesion classification

Choosing the Right Metric

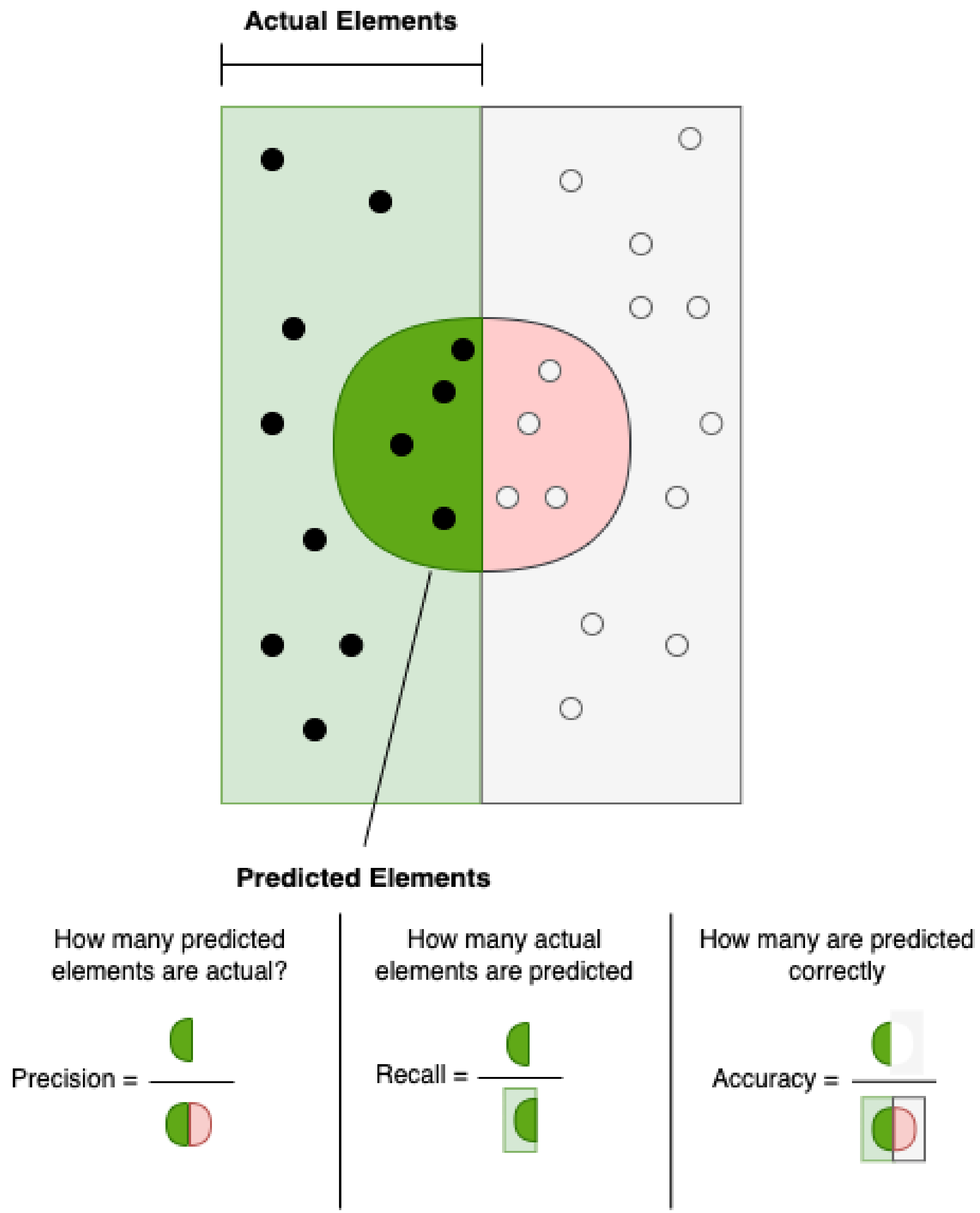
- Segmentation Tasks
  - Dice
  - IoU
  - HD95



Maxwell, A. E., Warner, T. A., & Guillén, L. A. (2021). Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies—Part 1: Literature Review. *Remote Sensing*, *13*(13), 2450. https://doi.org/10.3390/rs13132450

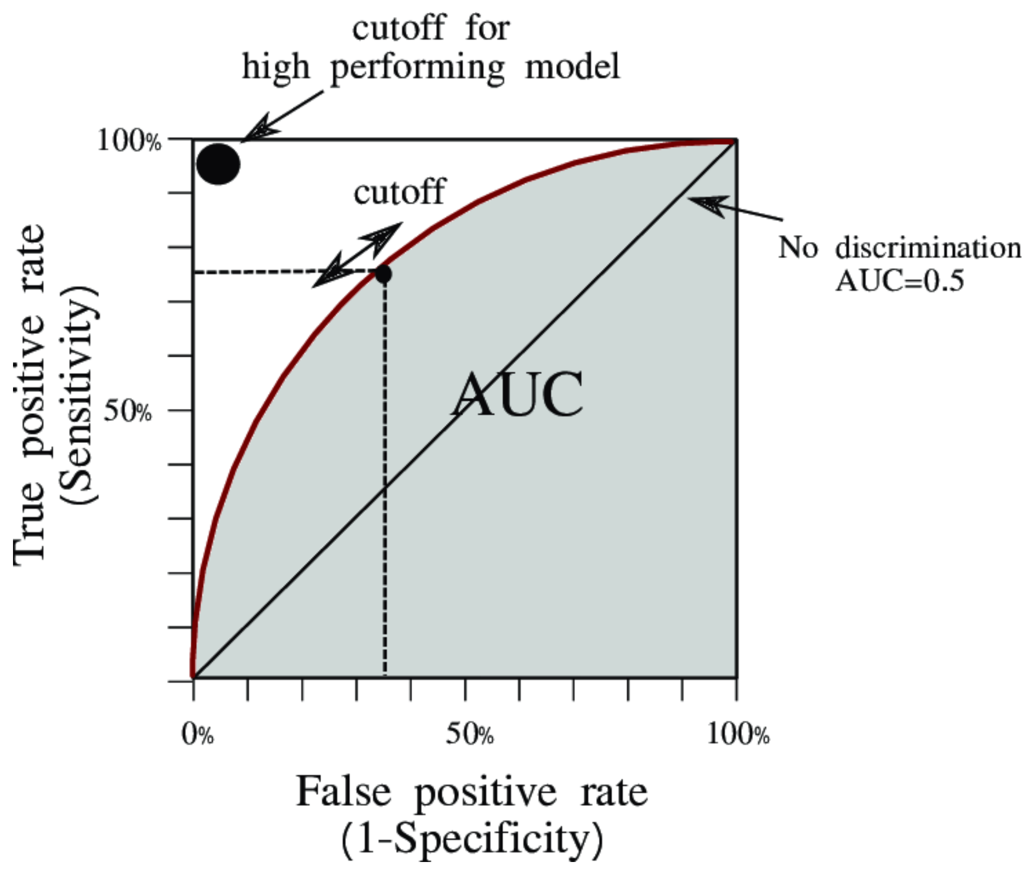
Choosing the Right Metric

- Classification Metrics
  - Precision
  - Accuracy
  - Recall



Choosing the Right Metric

- Honorable Mentions
  - MAE
  - Specificity / Sensitivity



Rodríguez-Hernández, M. & Pruneda, Rosa & Rodríguez-Díaz, Juan. (2021). Statistical Analysis of the Evolutive Effects of Language Development in the Resolution of Mathematical Problems in Primary School Education. Mathematics. 9. 1081. 10.3390/math9101081.

Learning Paradigm

Supervised: Requires labeled data.

Semi-Supervised: Requires only a subset of labeled data.

Unsupervised: Explores hidden structures in data.

Reinforcement: Learns through rewards and penalties.

Identify Constraints

Compute resources

GPU/CPU availability

Memory and Disk requirements

Real-world limitations

Label scarcity

Noisy data

Class Imbalance

Set Evaluation Criteria

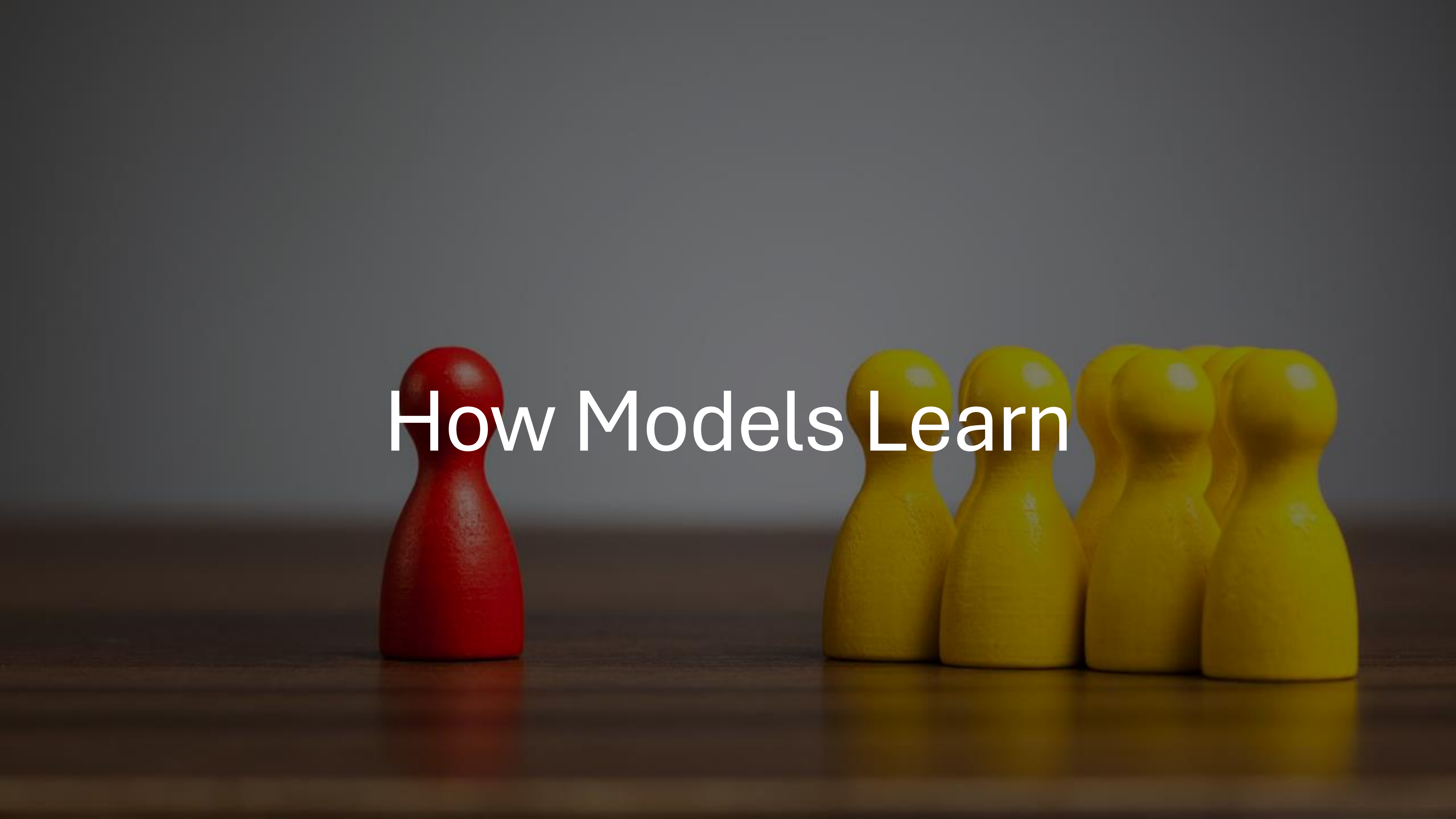
Use benchmarks to assess progress.

Define metrics and comparison baselines.

Example: Compare segmentation results with state-of-the-art models.

Visual Representation

A diagram showcasing how raw data transitions into a model output can help clarify the formulation.



How Models Learn

Key Concept

What is a model

- A nonlinear function transforming some input into output
- Our goal is trying to fit the **model weights**

$$f(x;\ \theta) = y$$

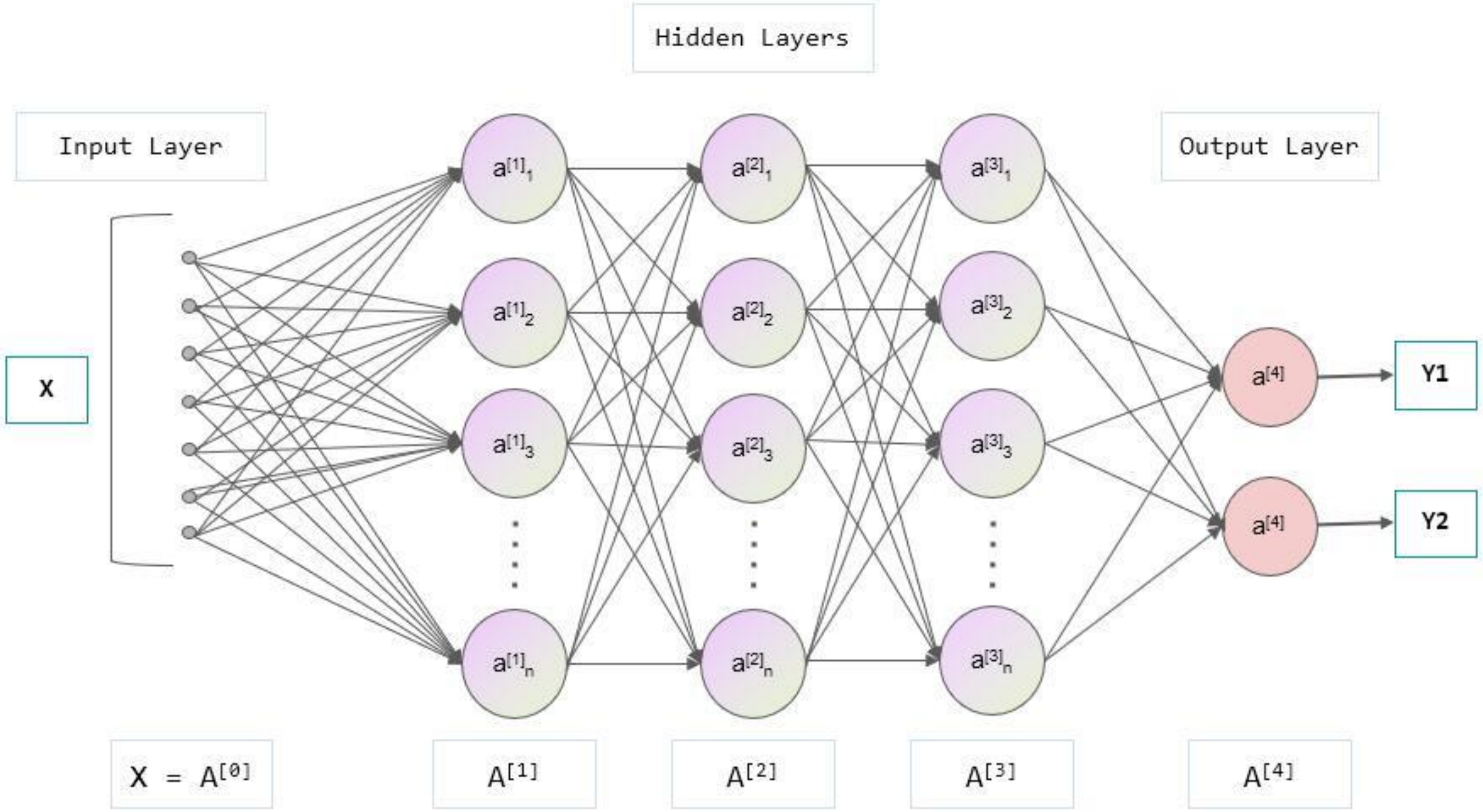
$$\begin{aligned} &f: \textit{The model (nonlinear function)} \\ &x: \textit{The input data} \\ &\theta: \textit{The parameters of the model (e.g., weights and biases)} \\ &y: \textit{The output (prediction).} \end{aligned}$$

Training Process

- Goal
  - Adjust model parameters (weights and biases) to minimize error between predictions and true outputs.
- Core Steps:
  - Forward Pass
    - Calculate predictions.
  - Loss Calculation
    - Measure the difference between predictions and true labels using a loss function (e.g., MSE, Cross-Entropy).
  - Backward Pass
    - Compute gradients to update parameters using backpropagation.

Simple Network

$$Y = F(x) \rightarrow WX + b$$



https://wiki.sipeed.com/soft/maixpy/en/course/ai/basic/dnn_basic.html

Backpropagation

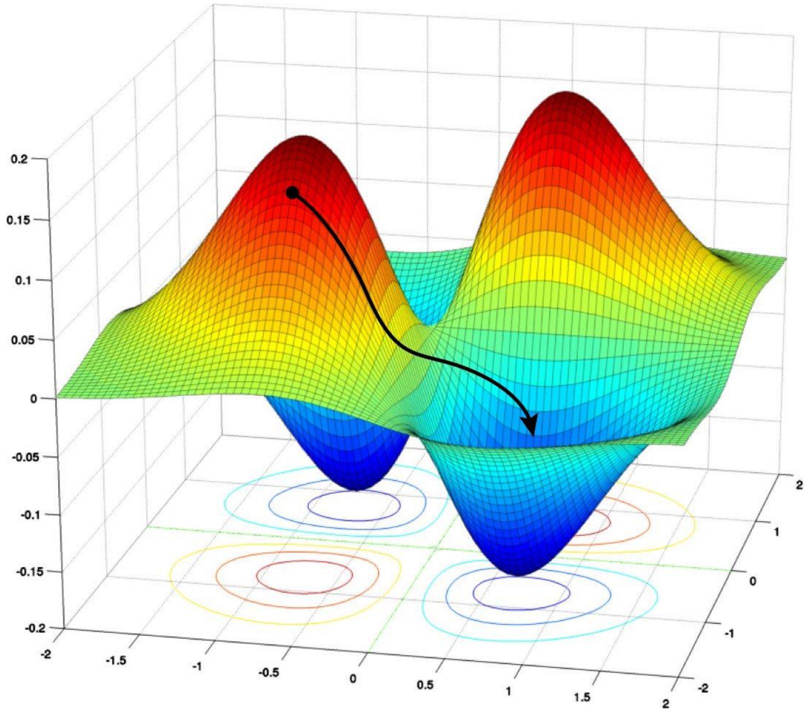
- Definition
  - A method for computing gradients of the loss function with respect to model parameters by applying the chain rule.
- Key Concepts:
  - Gradients flow backward from the output layer to earlier layers.
  - Updates are proportional to the gradient and learning rate.

$$w \leftarrow w - \eta \cdot \frac{\partial L}{\partial w}$$

$\text{where } \eta \text{ is the learning rate, } L \text{ is the loss, and } w \text{ are the weights.}$

Optimizers

Algorithms to update weights efficiently based on gradients

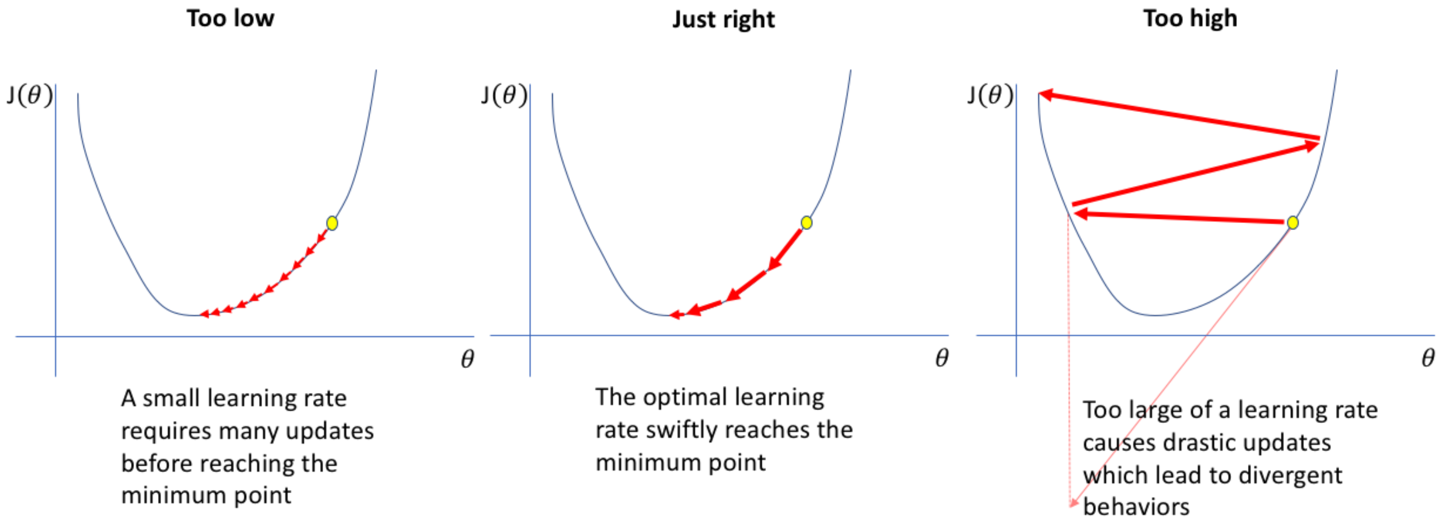


Optimizer	Key Features
SGD	Simple, uses the full gradient. Can be slow for large datasets.
Momentum	Adds velocity to updates, reducing oscillations.
RMSProp	Scales learning rate based on recent gradients.
Adam	Combines Momentum and RMSProp, adapts learning rates for each parameter.

https://medium.com/@tharangachaminda/neural-network-05-gradient-descent-for-neural-networks-cd5781d423fd

Learning Rate

- A critical hyperparameter that controls the step size in updates.
  - Too high: May overshoot minima.
  - Too low: Converges slowly or gets stuck.



https://www.jeremyjordan.me/nn-learning-rate/

Loss Functions

Measure how wrong the model's predictions are compared to the actual answers

Task Friendly	Metrics Compatible

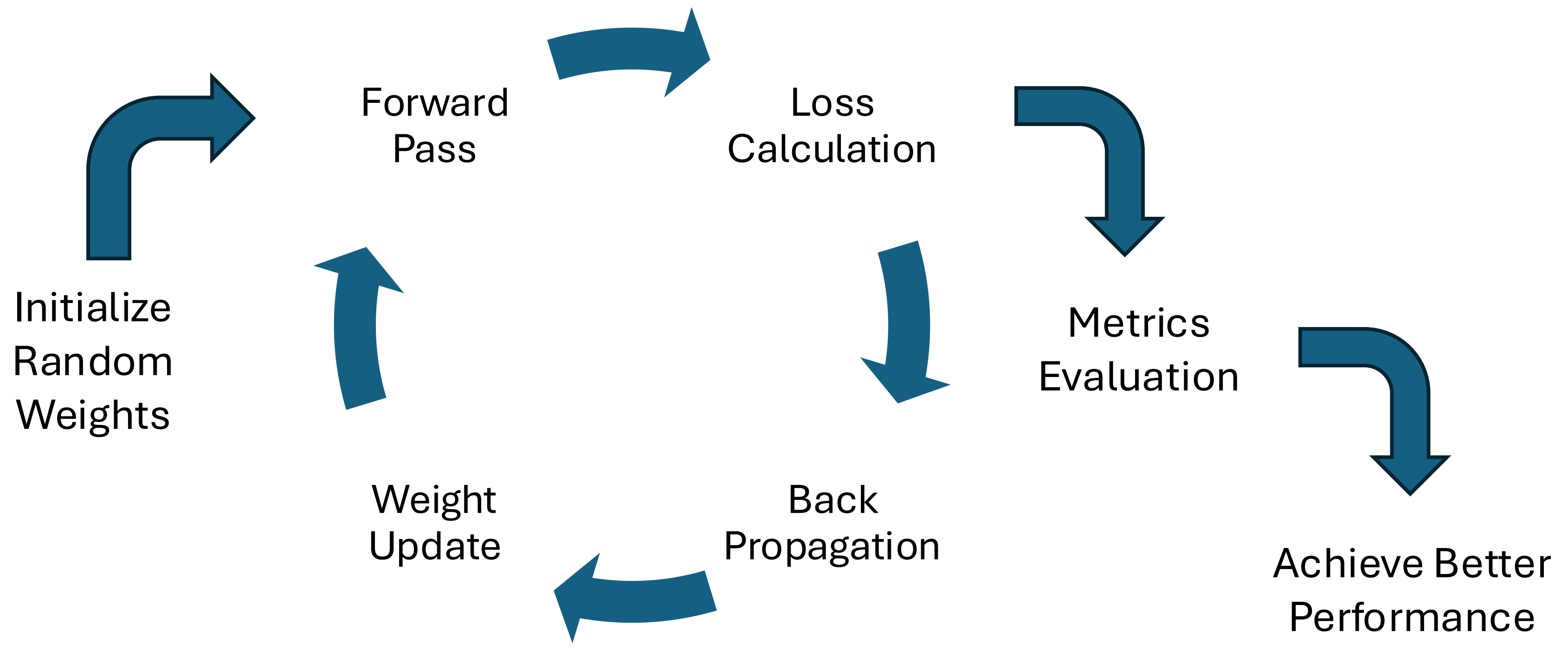
Example

Using a **CrossEntropy** loss for **Segmentation Task** while measuring **Dice Score**

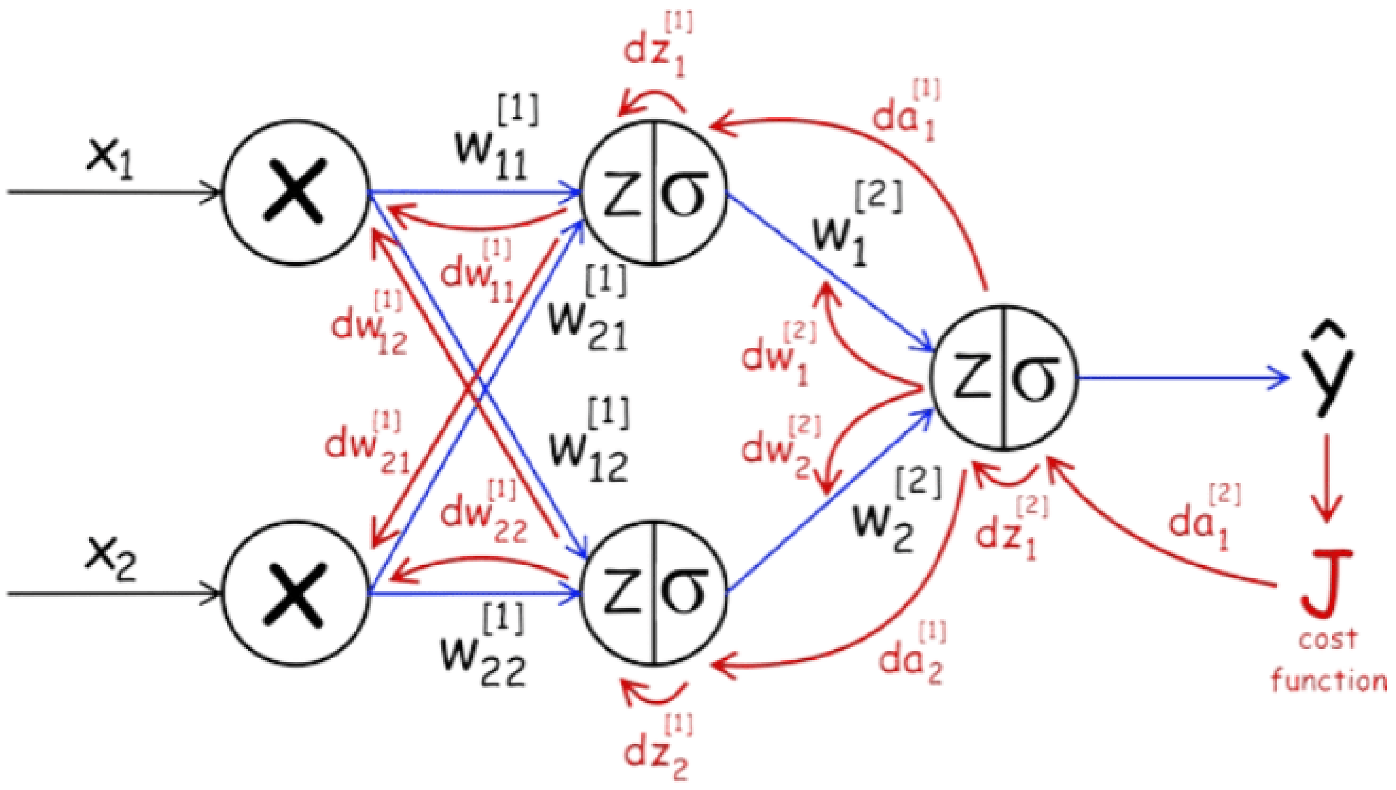
Loss Functions

- CrossEntropy
- MSE
- L1
- Focal Loss
- Combined Losses
  - DiceCrossEntropy
  - Jaccard-Focal Loss

Intuition for Learning



Summary



https://analyticsarora.com/8-unique-machine-learning-interview-questions-on-backpropagation/

Questions?

Workshop Activity

- Notebooks Link
  - https://github.com/KnightsLab/EMRA-Workshop