lec 1

Perceptron Training Rule https://www.youtube.com/watch?v=du5fyS44DR8

AND GATE Perceptron Training Rule https://www.youtube.com/watch?v=zmlzNBMsQYQ

OR GATE Perceptron Training Rule https://www.youtube.com/watch?v=uxmNDNb0u9A

Read https://chatgpt.com/c/67076a8b-3224-800d-b174-58b9ad6f1001

Perceptron Rule to design XOR https://www.youtube.com/watch?v=dM_8Y41EgsY

---

1. Composing Complicated Decision Boundaries:

- A **decision boundary** is the dividing line (or surface) that separates different classes of data in a classification task.
- For simple tasks, a **linear decision boundary** (like a straight line) might be enough, but for more complex tasks, we need **non-linear** boundaries.
- In a neural network, we can create complex decision boundaries by stacking multiple **layers of neurons**. Each neuron in a layer can create a simple linear boundary, and by combining these, we can form a more complex overall boundary.

2. Building a Network of Units with a Single Output:

- Imagine we have a network of perceptrons (basic units or neurons). Each perceptron is responsible for making a binary decision: whether an input is on one side of a boundary or the other.
- The **goal** is for the entire network to "fire" (output a positive response) if the input lies within a certain **colored area** (i.e., the desired region defined by the decision boundary).
- Each perceptron creates a line that divides the input space, and the **combined output** from all these perceptrons forms the complex boundary we need.

3. The Network Must Fire if the Input is in the Colored Area:

- For each input, the network should correctly classify whether the input belongs to the target region (colored area) or not.
- To do this, each perceptron defines a simple linear boundary, and when combined, the output from higher layers of the network will classify the input as inside or outside the desired region.

4. Each Perceptron Creates a Line (Boundary):

- A perceptron's decision boundary is linear. It draws a line in the input space that separates two regions. The neuron "fires" (outputs 1) if the input is on one side of the line and outputs 0 if it's on the other side.
- **Higher-level perceptrons** (in later layers of the network) combine these linear boundaries from the earlier neurons to form more complex, non-linear boundaries.

5. Exponential Search Over Inputs:

- The network explores all possible combinations of inputs to find the correct decision boundary. This can be seen as an **exponential search**, since the number of possible boundary configurations grows quickly with the number of neurons and inputs.

6. Must Know the Desired Output for Every Neuron:

- For the network to learn the correct boundaries, we need to know the desired output for each neuron in the network for every training example. This is part of the training process.
- Each neuron is trained on a **linearly separable task**, meaning it just needs to create a simple linear boundary (a line in 2D, a plane in 3D, etc.).

7. Combining Linear Separators:

- The network combines these **linear separators** created by individual neurons into a more complex, non-linear decision boundary.
- For example, if we need to classify inputs into regions that form more complicated shapes (like circles or curves), the neurons' linear outputs are combined in a way that achieves this complex boundary.

8. Importance of Every Neuron:

- Each neuron in the network plays a role in shaping the overall decision boundary. If any neuron fails to learn the correct output or boundary, the final decision boundary might be incorrect, leading to **wrong predictions**.

Key Concepts Recap:

- **Decision Boundaries**: The surface or line that separates different classes of data.
- **Perceptrons**: Simple units that create linear decision boundaries.
- **Higher-level Perceptrons**: Combine multiple linear boundaries to form complex, non-linear boundaries.
- **Exponential Search**: As the number of neurons increases, the network explores an exponentially growing space of possible boundaries.
- **Linearly Separable Tasks**: Each neuron solves a simple problem that can be separated by a line, but when combined, they solve more complex problems.
- **Training**: The network learns by adjusting the boundaries to fit the desired outputs. Getting even one boundary wrong can lead to incorrect final predictions.

In summary, building a neural network involves composing many simple linear decisions (boundaries) and combining them in layers to create complex patterns and non-linear decision boundaries. Each neuron plays a critical role, and the entire network must work together to form the correct decision boundary for accurate classification.