Deep Learning Course – L 4 - Detailed Notes
Logistic Regression for Cat vs Non-Cat Classification

🎯 Objective

Classify images into two categories:

- **1** → Cat (Positive class)
- **0** → Not a cat (Negative class)

This is a classic **binary classification** problem using **logistic regression**.

🐱 How Images are Represented in Machine Learning

- Each image is made up of **pixels** and has **3 color channels**: Red, Green, Blue (RGB).
- Example: A 64x64 image has:
  - 64 pixels (height) × 64 pixels (width) × 3 color channels = **12,288 values**
- These pixel values are flattened into a **1D vector** `x` of size **12,288**.

□ Notation Overview

Symbol	Meaning
`m`	Number of training examples
`n`	Number of features per example (e.g., 12,288)
`x`	Input vector (image)
`y`	True output (0 or 1)
`ŷ`	Predicted output (probability)

🐱 Logistic Regression Explained

- Logistic Regression is a **binary classification algorithm**.
- It predicts a probability that an input belongs to the **positive class (cat)**.
- Prediction ($\hat{y}$) is a **probability between 0 and 1**.
    - Closer to 1 → likely a cat
    - Closer to 0 → likely not a cat

⚙️ Key Components of the Model

🔹 *Weights (`w`)*

- A vector that adjusts the importance of each pixel (feature).

🔹 *Bias (`b`)*

- A constant added to the result to help shift the prediction.

🔹 *Linear Combination*

- `z = wᵀx + b`
  (Dot product of `w` and `x`, then add `b`)

🔹 *Sigmoid Function*

- Converts `z` into a probability:

  σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1

- This ensures $\hat{y}$ is always between 0 and 1.

🧠 Example Calculation

Given:

- `x = [2, 1, 3]`
- `w = [1, 2, 1]`
- `b = -1`

Step-by-step:

1. `z = (1×2) + (2×1) + (1×3) + (-1) = 2 + 2 + 3 - 1 = 6`
2. `ŷ = sigmoid(6) ≈ 0.997`
   → Very high probability = most likely a **cat**

Why Linear Algebra Matters

- Vectors and matrices make these operations efficient.
- Multiply inputs by weights → add bias → apply sigmoid.

Why Use Sigmoid?

- Converts any number into a probability between 0 and 1.
- Smooth, continuous, and interpretable.
- Makes it easy to classify:
    - 0.5 → Cat
    - <0.5 → Not a Cat

Link to Neural Networks

- Logistic regression = **one-layer neural network**
- Neural networks stack layers of these computations to learn **complex patterns**.

Real-Life Binary Classification Examples

1. **Health Prediction**: Smoker (1) or Non-Smoker (0)
2. **Medical Diagnosis**: Positive (1) or Negative (0) test result

---

Highlighted Questions & Answers

Q1: Why is the sigmoid function used in logistic regression?
→ It maps inputs to probabilities between 0 and 1, ideal for binary classification.

Q2: What happens if `z` is very large or very small?

- Large positive `z` → sigmoid ≈ 1
- Large negative `z` → sigmoid ≈ 0
- `z = 0` → sigmoid = 0.5

Q3: Why are weights and bias important?
→ They let the model **control how each feature affects** the prediction.

📓 Glossary (Difficult Terms Explained)

Term	Description
Binary Classification	A type of task where the goal is to assign one of two possible labels (e.g., cat or not-cat).
Sigmoid Function	A math function that squashes numbers into a range between 0 and 1, perfect for probabilities.
Weights and Bias	Parameters that the model learns to improve its predictions—like dials that tune the model's output.
Linear Algebra	A branch of math dealing with vectors and matrices, essential for machine learning computations.

✅ Final Summary

Logistic regression is a foundational model in deep learning. It's used to solve binary classification problems by converting input data into a vector, applying weights and bias, and passing the result through a sigmoid function to get a probability. Understanding this process is essential for grasping how more complex neural networks work.