Deep Learning Course – L 5 - Detailed Notes

🎯 Understanding Loss Functions, Cost Functions, and Gradient Descent | Neural Networks Explained

In this lecture, we dive deep into **loss functions**, **cost functions**, and **gradient descent** — three essential building blocks in machine learning and deep learning.

Whether you're training a **logistic regression model** or a **neural network**, these concepts help your model learn, reduce errors, and improve accuracy.

📌 Key Concepts Covered:

🔹 *Loss Function*

- Measures how far off a single prediction is from the actual result.
- Commonly used: **Binary Cross-Entropy Loss** (Negative Log Loss).
- Example:
    - If actual = 1 and predicted = 0 → High Loss.
    - If actual = 0 and predicted = 0 → Low Loss.

🔹 *Cost Function*

- The **average of all loss values** across the training data.
- Helps evaluate **overall model performance**.
- Used to guide the model toward **better predictions**.

🔹 *Gradient Descent*

- An optimization algorithm used to minimize the cost function.
- Repeatedly updates the **weights (W)** and **bias (B)** by:
    - Calculating gradients.
    - Moving in the **opposite direction** of the gradient.
- Continues until the model reaches **optimal performance**.

🧠 Why It Matters

- These methods **train the model** to predict values closer to the truth.
- **Loss** = Single data point evaluation.

- **Cost** = Entire dataset evaluation.
- **Gradient Descent** = Learning algorithm that improves the model step-by-step.

🐱 Example: Cat vs Non-Cat Image Classification

- Loss functions help identify how wrong the model is.
- Cost function summarizes the model’s performance across all images.
- Gradient descent fine-tunes parameters for better classification accuracy.

📊 Quick Comparison:

Loss Function	Cost Function
One training example	All training examples (average)
Guides one step update	Guides overall performance

✅ Final Takeaway

Understanding **loss, cost, and optimization** is crucial to mastering ML and DL models. These concepts enable your model to **learn from data**, make better predictions, and power applications like **image recognition**, **medical diagnosis**, and **recommendation systems**.

xz