Deep Learning Course – L 3 - Detailed Notes

Introduction to Neural Networks (House Price Prediction Example)

---

1. Neural Networks and Deep Learning – The Starting Point

Neural Networks are the **backbone of deep learning**. To understand them, we begin with a **real-world problem**: predicting house prices.

---

2. 🏠 House Price Prediction Using Linear Regression

Problem Setup

You're given a dataset of **6 houses** where:

- **Input (X):** Size of the house (in square feet)
- **Output (Y):** Price of the house

Goal:

Create a **function** that maps house size → predicted price.

Solution: Linear Regression

- A **basic model** in machine learning.
- Represents the data as a **straight line**:

  y=wx+by = wx + by=wx+b

  where:

  - **w** = weight (slope)
  - **b** = bias (intercept)
  - **x** = input (house size)
  - **y** = predicted output (house price)

Takeaways:

- Maps **one input** to **one output**.
- Acts like a **single neuron**: takes input → applies math → gives output.

---

3. ➕ Expanding the Model: Adding More Features

Why?

A single input (house size) is too basic for **accurate predictions**.

Additional Useful Features:

- **Number of bedrooms** – reflects family capacity.
- **Postal code** – location strongly influences price.
- **Stores nearby** – more amenities = higher value.

Benefit:

- Combining multiple features improves prediction.
- Instead of handcrafting every rule, let the **neural network learn patterns**.

Takeaways:

- **More features = more informative data = better predictions.**
- Neural networks can **learn feature importance automatically**.

---

4. 🧠 What is a Neuron? What is a Neural Network?

Neuron:

- A function that takes one or more **inputs**, applies some math, and gives an **output**.
- Example: A neuron could learn to predict house price based on size and number of bedrooms.

Neural Network Structure:

- **Input Layer:** Takes raw data (e.g., size, bedrooms, postal code).
- **Hidden Layers:** Where multiple **neurons** work together to learn complex patterns.
- **Output Layer:** Gives the final result (e.g., predicted price).

Fully Connected Network:

Each neuron in a layer is **connected to every neuron in the next layer**.

Takeaways:

- **Stacking layers = Deep Neural Network**.
- Neural networks can model **complex relationships** that are hard to design manually.
- These models are great for **structured data problems** like price prediction.

---

5. 📈 Predicted vs. Actual Price (ŷ vs. y)

ŷ (y-hat): The model's predicted price.

y: The actual price of the house.

Goal of training the model:

Minimize the **difference** between **ŷ** and **y** (known as **loss or error**).

This difference helps us understand:

- How accurate the model is.
- How to **update the model's parameters** during training to make better predictions.

Takeaways:

- Neural networks aim to **reduce prediction error** by learning from data.

---

6. 🎯 Final Thoughts

- Neural networks **evolve** from basic models like linear regression.
- We don't need to manually define rules—**neural networks learn them** through training.
- Understanding features like **location, size, and amenities** is key in prediction tasks.
- This knowledge is foundational for understanding **more advanced deep learning** later.

💡 Difficult Terms Explained

Term	Explanation
Linear Regression	A simple model that fits a straight line to data to predict an output based on input.
Feature	A characteristic or attribute of data (e.g., size, number of bedrooms).
Neuron	A basic unit in a neural network that processes inputs and gives an output.
Fully Connected Layer	A layer where every neuron is connected to all neurons in the next layer.
ŷ (y-hat)	Symbol representing the model's **predicted output**.
Loss/Error	The difference between the predicted value and actual value, used to improve the model.