Neural Networks
Resource & Practical Guide

Jayden Cruz

February 24, 2026

1. Main Points from the Lecture

What is a Neural Network?

A neural network is a mathematical model designed to recognize patterns in data. It is composed of layers of connected neurons that learn by adjusting weights to minimize error.

Core Components

Neuron Computation:

$$z = \sum_{i=1}^{n} x_i w_i + b$$

$$a = f(z)$$

- w_i = weights (importance of features)
- b = bias (shift term)
- $f(z)$ = activation function

Network Architecture

- Input Layer – receives features
- Hidden Layers – extract patterns
- Output Layer – produces prediction

The depth of a network refers to the number of hidden layers.

How Neural Networks Learn

1. Forward propagation (make prediction)
2. Compute loss (measure error)
3. Backpropagation (compute gradients)
4. Update weights (gradient descent)
5. Repeat for many epochs

Training Terminology

- Epoch = one full pass through the training dataset
- Batch = subset of data processed at once
- Train / Validation / Test Split:
  - Train – learn weights
  - Validation – tune model
  - Test – evaluate final performance

2. Practical Guide: Building a Simple Neural Network (Keras Example)

Step 1: Import Libraries

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

Step 2: Create the Model

```
model = Sequential()

model.add(Dense(16, activation=’relu’, input_dim=4))
model.add(Dense(8, activation=’relu’))
model.add(Dense(1, activation=’sigmoid’))
```

Step 3: Compile the Model

```
model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])
```

Step 4: Train the Model

```
model.fit(x_train, y_train,
          epochs=10,
          batch_size=32,
          validation_split=0.2)
```

Step 5: Evaluate the Model

```
model.evaluate(x_test, y_test)
```

Code: You're Welcome

https://colab.research.google.com/drive/1CpPn-Wa9HpsKO9yBIatmgYor6zmNOjO1?usp=sharing

3. Helpful Resources

- 3Blue1Brown Neural Networks Series:
  https://www.youtube.com/watch?v=aircAruvnKk
- TensorFlow Playground (Interactive Demo):
  https://playground.tensorflow.org
- Keras Documentation:
  https://keras.io
- Backpropagation (Wikipedia):
  https://en.wikipedia.org/wiki/Backpropagation
- Illustrated Transformer (ChatGPT Architecture):
  https://jalammar.github.io/illustrated-transformer/

4. Practice Prompts

Prompt 1: Conceptual Understanding
Explain why activation functions are necessary in neural networks. What would happen if we removed them?

Prompt 2: Training Logic
If a neural network has 1,000 training examples and uses a batch size of 100:

- How many batches are in one epoch?
- How many weight updates occur per epoch?

Prompt 3: Real-World Application
Describe how a convolutional neural network helps a self-driving car recognize pedestrians. What type of data does it process? What is the final output?

Reflection Questions (Optional)

- What is one limitation of neural networks?
- Why is generalization important?
- Where do you see neural networks used in your field?