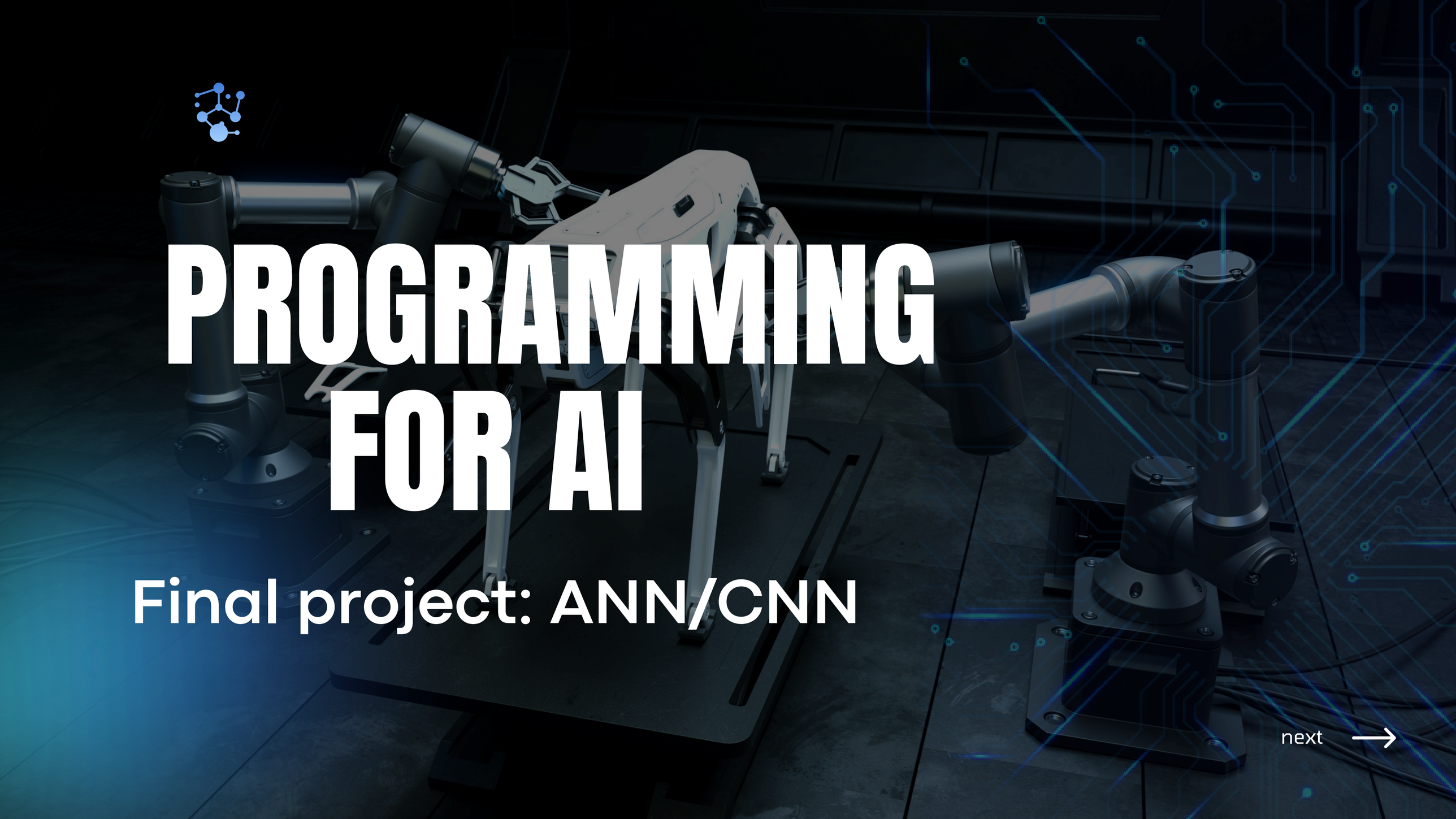


PROGRAMMING FOR AI

Final project: ANN/CNN

next →

GROUP MEMBERS:

Ayesha Tahir(SP23-BAI-019)
Eman Butt(SP23-BAI-014)

next →

ANN CLASSIFICATION

DATASET DESCRIPTION

DataSet Name : CIFAR-10

Size :

Training Set: 50,000 images

Testing Set: 10,000 images.

Features:

Input Shape: 32x32 pixels with 3 color channels (RGB)

Each image is flattened into a vector of 3,072 features (32x32x3) for ANN.

Classes: 10 categories (e.g., airplane, car, bird, cat, deer, dog, frog, horse, ship, truck).

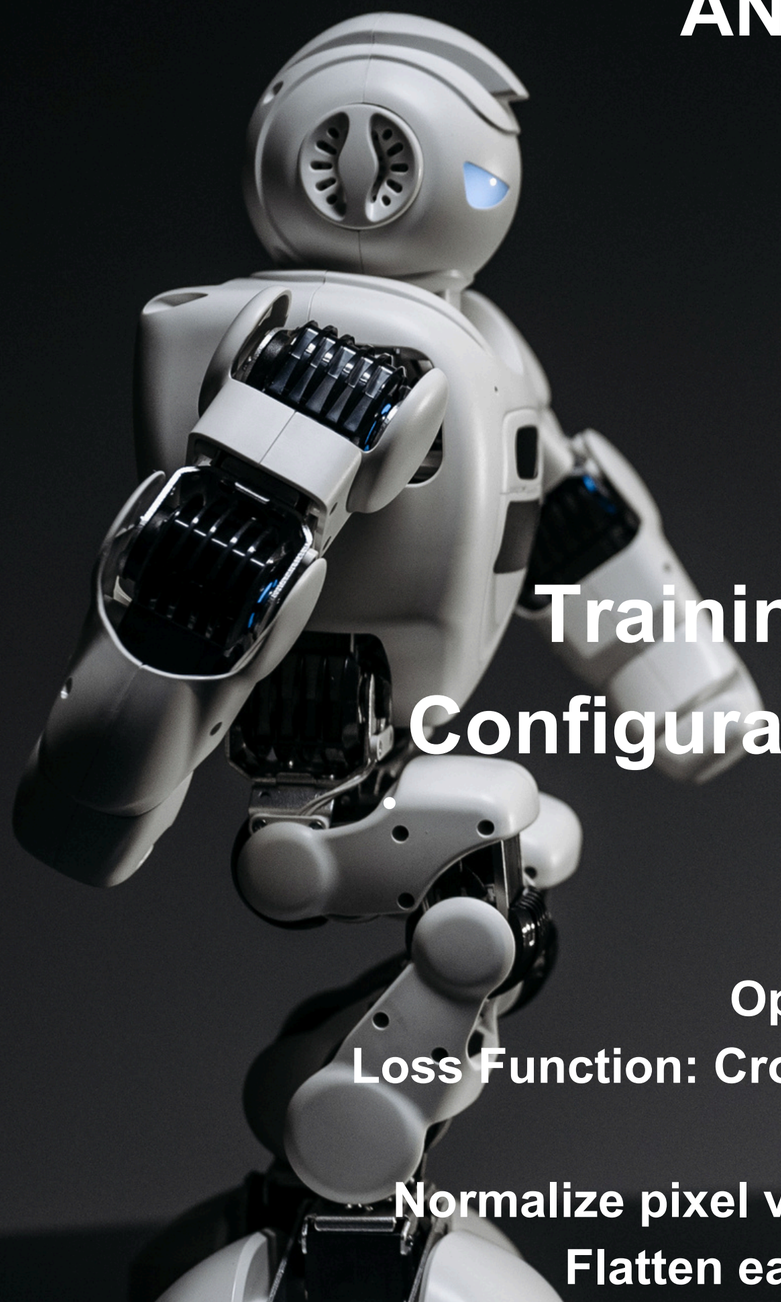
Train/Test Split:

80% training, 20% testing.

Additional validation split: 10% of training data used for validation.

next →

MODEL DETAILS



PyTorch ANNArchitecture

Input Layer:

3,072 nodes (flattened image).

Hidden Layers:

a. **Layer 1: 512 neurons with ReLU activation.**

b. **Layer 2: 256 neurons with ReLU activation.**

c. **Layer 3: 128 neurons with ReLU activation.**

Output Layer:

10 neurons with Softmax activation to calculate class probabilities

Training Configurations

- **Learning Rate: 0.001**
- **Batch Size: 64**
- **Number of Epochs: 50**
- **Optimizer: Adam (adaptive learning rate).**
- **Loss Function: CrossEntropyLoss (suitable for multi-class classification).**
- **Data Preprocessing:**
  - **Normalize pixel values to [0, 1] range for consistent gradient behavior.**
  - **Flatten each image from 32x32x3 to 1D (3,072 features).**

next →

HOUSING CALIFORNIA

DATASET
REGESSION

Size:

Training Set: 16,640 samples (80%)

Testing Set: 4,160 samples (20%)

Features:

Input Features: 8 numerical features (e.g., median income, total rooms, population).

Target Variable: Median house value (continuous).

Train/Test Split:

80% training data, 20% testing data.

Additional validation split: 10% of training data used for validation.

next →

DATASET
REGESSION

MODEL DETAILS

PyTorch ANN Architecture:

Input Layer:

8 nodes (one for each feature).

Hidden Layers:

a. **Layer 1: 128 neurons with ReLU activation.**
b. **Layer 2: 64 neurons with ReLU activation.**
c. **Layer 3: 32 neurons with ReLU activation.**

Output Layer:

1 neuron with no activation (produces a continuous output for regression).

TRAINING CONFIGURATION

Learning Rate: 0.001

Batch Size: 32

Number of Epochs: 100

Optimizer: Adam (adaptive learning rate).

Loss Function: Mean Squared Error (MSE) – minimizes the difference between predicted and actual values.

Data Preprocessing:

Normalize all features to ensure a consistent scale.

No need for flattening since features are already numerical

next →

MINST

Size:

Training Set: 60,000 images

Testing Set: 10,000 images

Features:

Input Shape: 28x28 grayscale images (1 channel).

Each pixel value ranges from 0 to 255 (converted to [0, 1] after normalization).

Classes: 10 digits (0 to 9).

Train/Test Split:

80% training, 20% testing.

Additional validation split: 10% of training data used for validation.

next →

Model Details

Keras CNN Architecture

Input Layer:

Input shape = (28, 28, 1) (grayscale image).

Convolutional Layers:

a. Conv2D Layer 1: 32 filters, 3x3 kernel, ReLU activation, padding='same'.

b. MaxPooling2D Layer 1: Pool size = 2x2.

c. Conv2D Layer 2: 64 filters, 3x3 kernel, ReLU activation, padding='same'.

d. MaxPooling2D Layer 2: Pool size = 2x2.

Flatten Layer:

Converts the 2D feature maps into a 1D vector.

Fully Connected (Dense) Layers:

a. Dense Layer 1: 128 neurons with ReLU activation.

b. Output Layer: 10 neurons with Softmax activation (for multi-class classification).

Training Configurations

Learning Rate: 0.001 (default for Adam optimizer).

Batch Size: 64

Number of Epochs: 20

Optimizer: Adam (adaptive learning rate).

Loss Function: Categorical Cross-Entropy (suitable for multi-class classification).

Metrics: Accuracy.

Data Preprocessing:

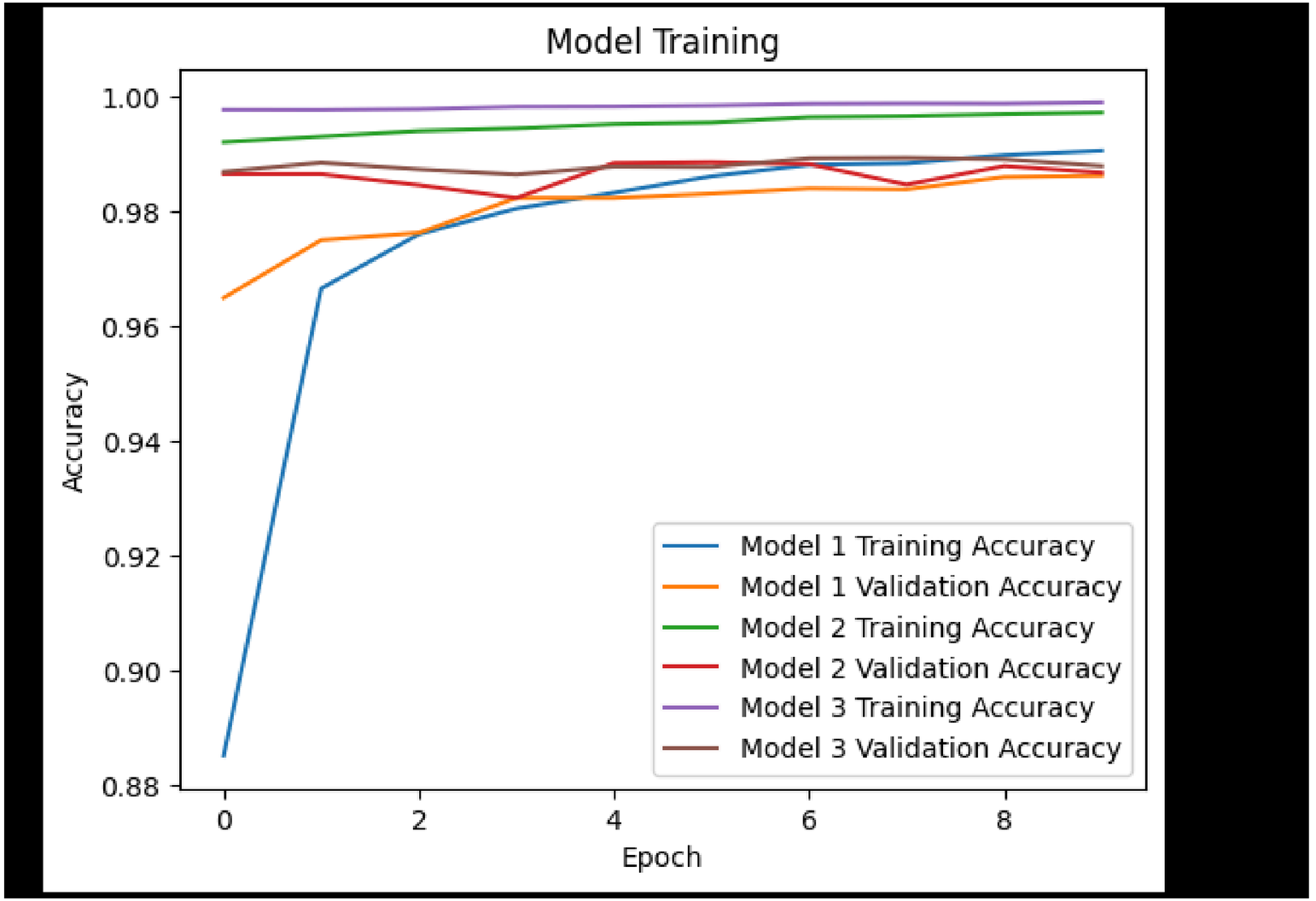
Normalize pixel values to the range [0, 1].

Reshape the input to (28, 28, 1) for compatibility with CNN.

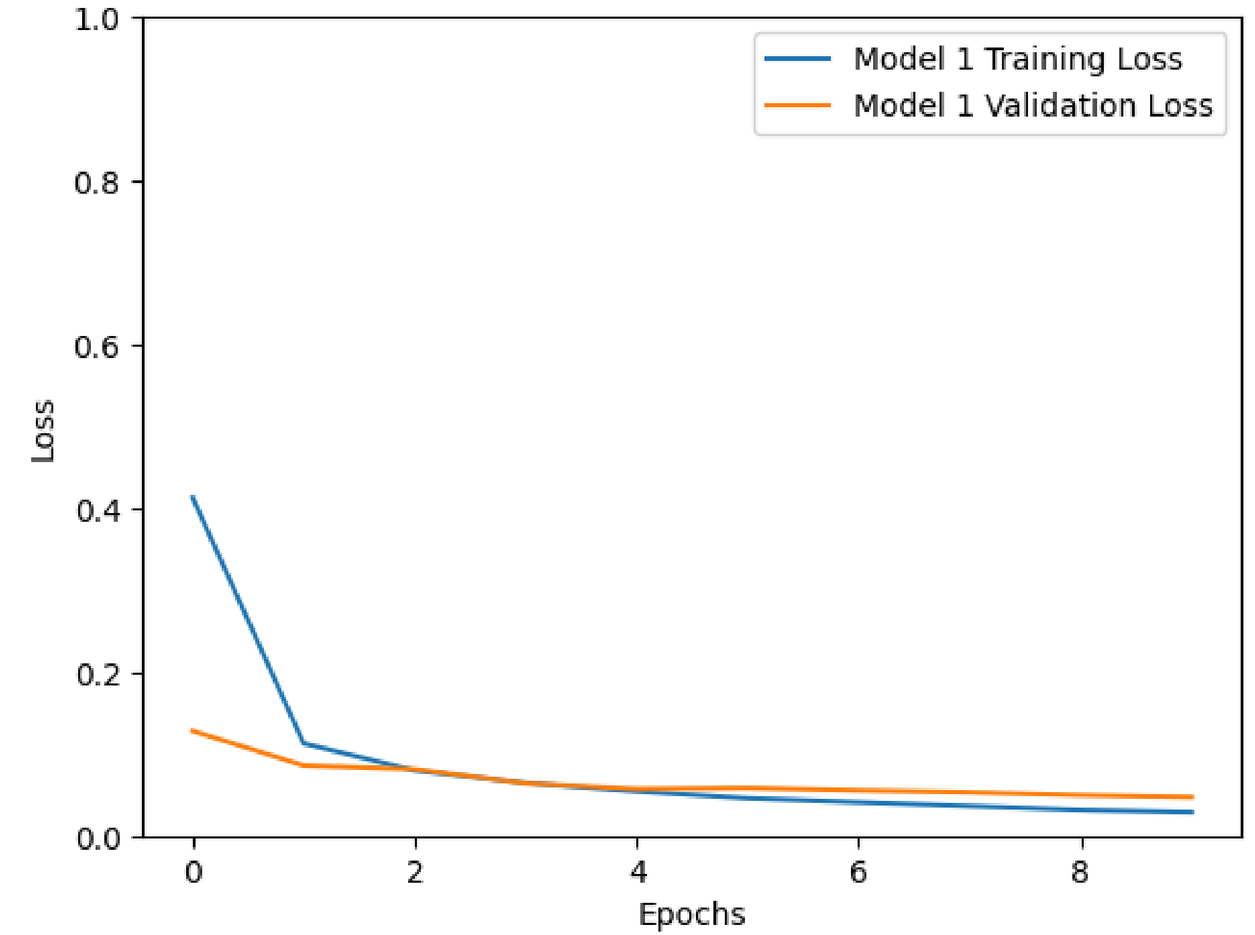
Convert labels to one-hot encoded format.

next ⟶

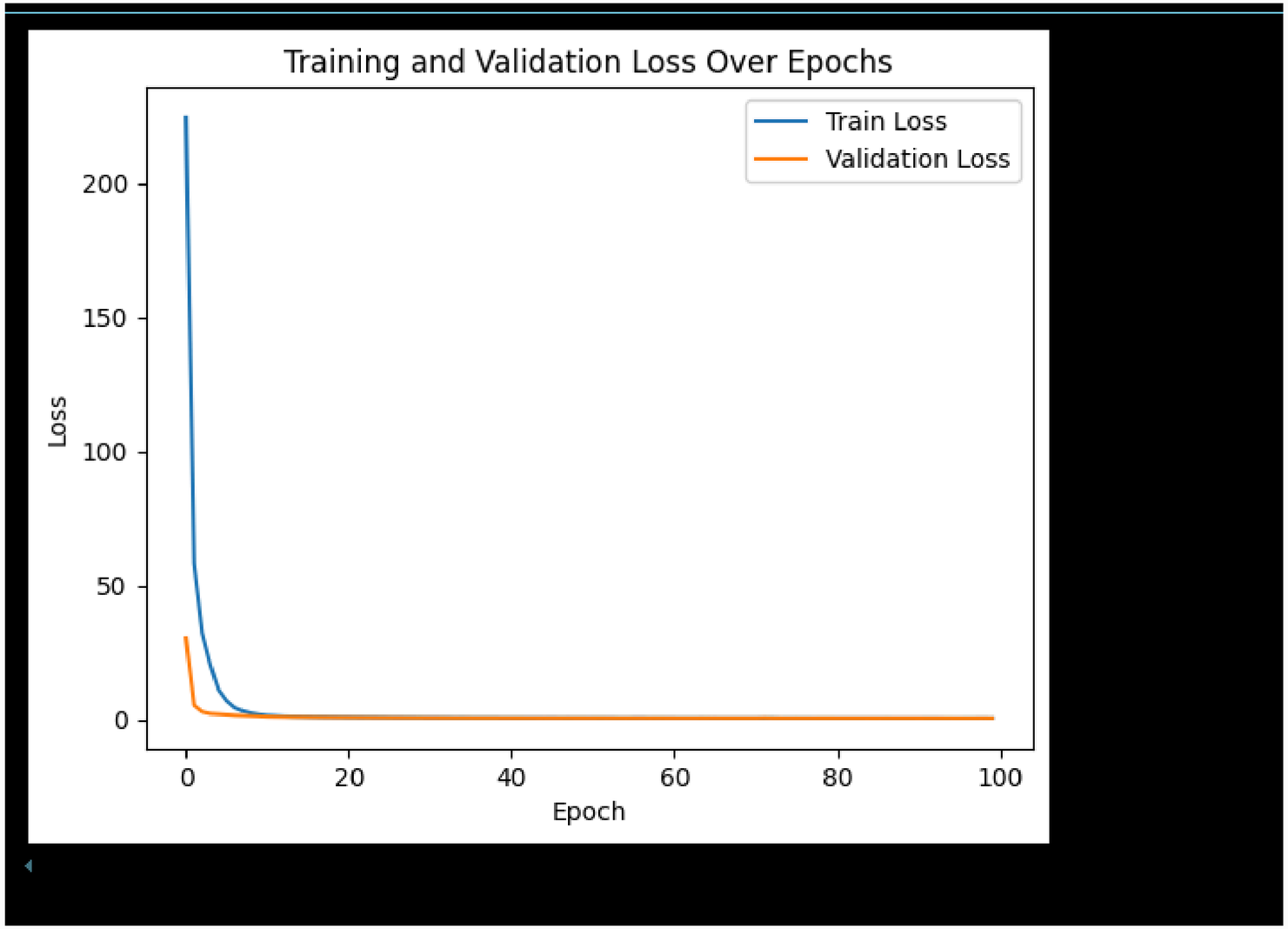
CNN Accuracy:



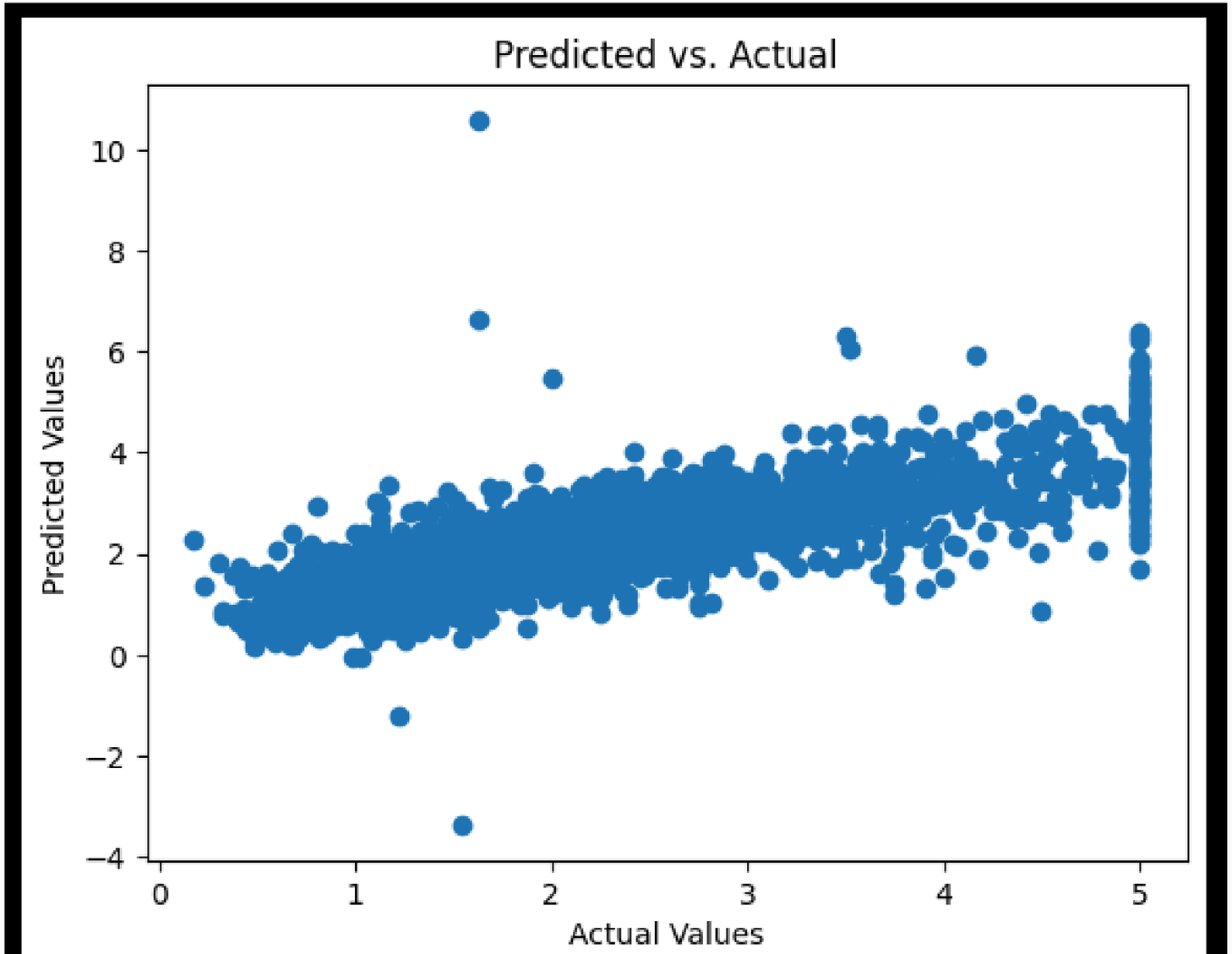
CNN Loss:


Model 1 Training Loss
Model 1 Validation Loss
1.0
0.8
0.6
0.4
0.2
0.0
Loss
0
2
4
6
8
Epochs

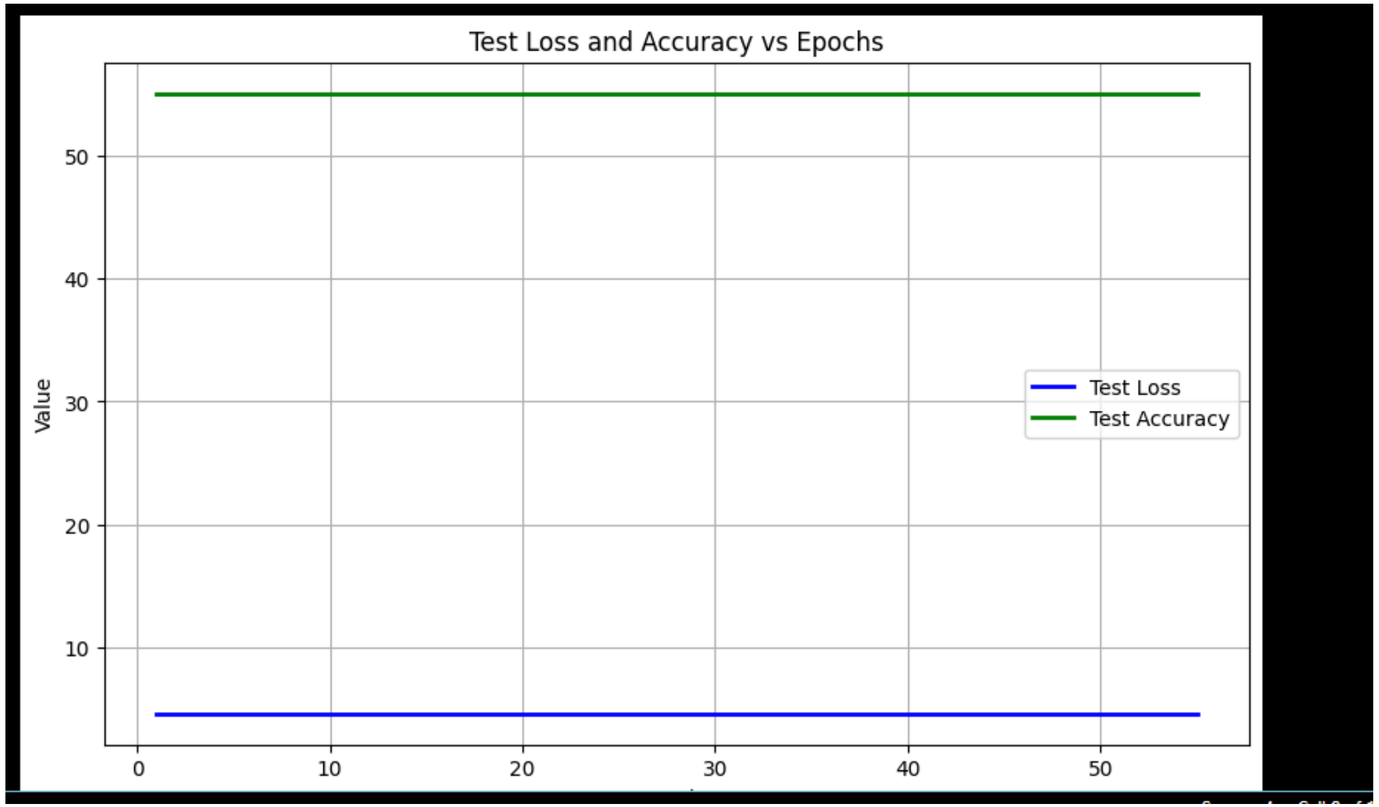
Regression:


Training and Validation Loss Over Epochs
Train Loss
Validation Loss
Loss
200
150
100
50
0
0
20
40
60
80
100
Epoch

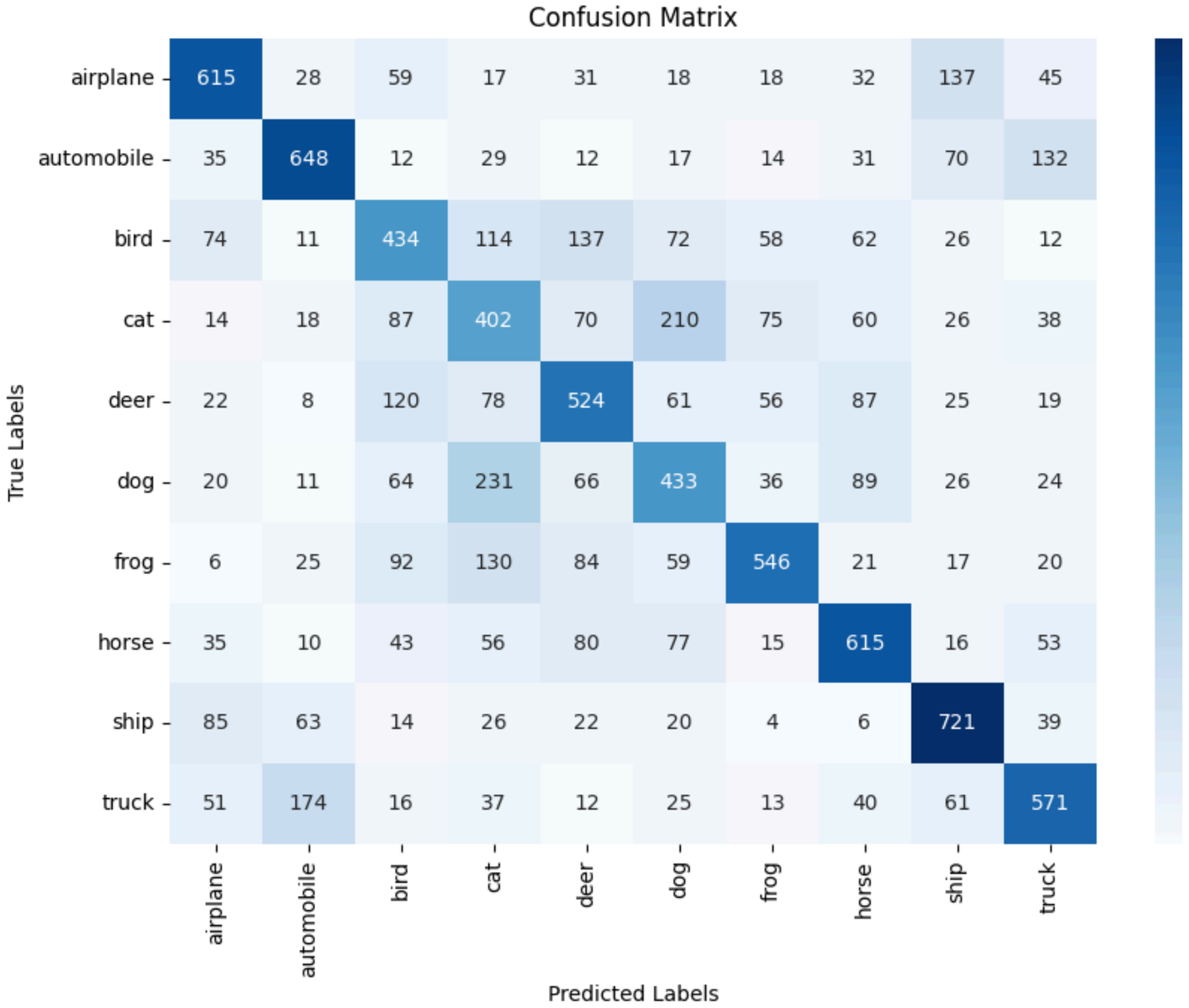
Regression:


Predicted vs. Actual
Predicted Values
Actual Values
10
8
6
4
2
0
−2
−4
0
1
2
3
4
5

ANN Test Loss vs Epoch:



Confusion matrix:cifar10



Comparison

Model	Dataset / Task	Key Hyperparams	Final Metric
PyTorch ANN	California Housing (Reg)	LR=0.01, Batch Size=512, Hidden Neurons=10, Optimizer=Adam, Loss=MSE	MSE=0.4378, RMSE=0.6617, MAE=0.4657, R^2=0.6689
PyTorch ANN	CIFAR-10 (Class)	LR=0.01, Epoch=55	Accuracy=55%
Keras CNN	MNIST (Class)	LR=0.01, Epoch=10, Batch Size=32	Accuracy=98%