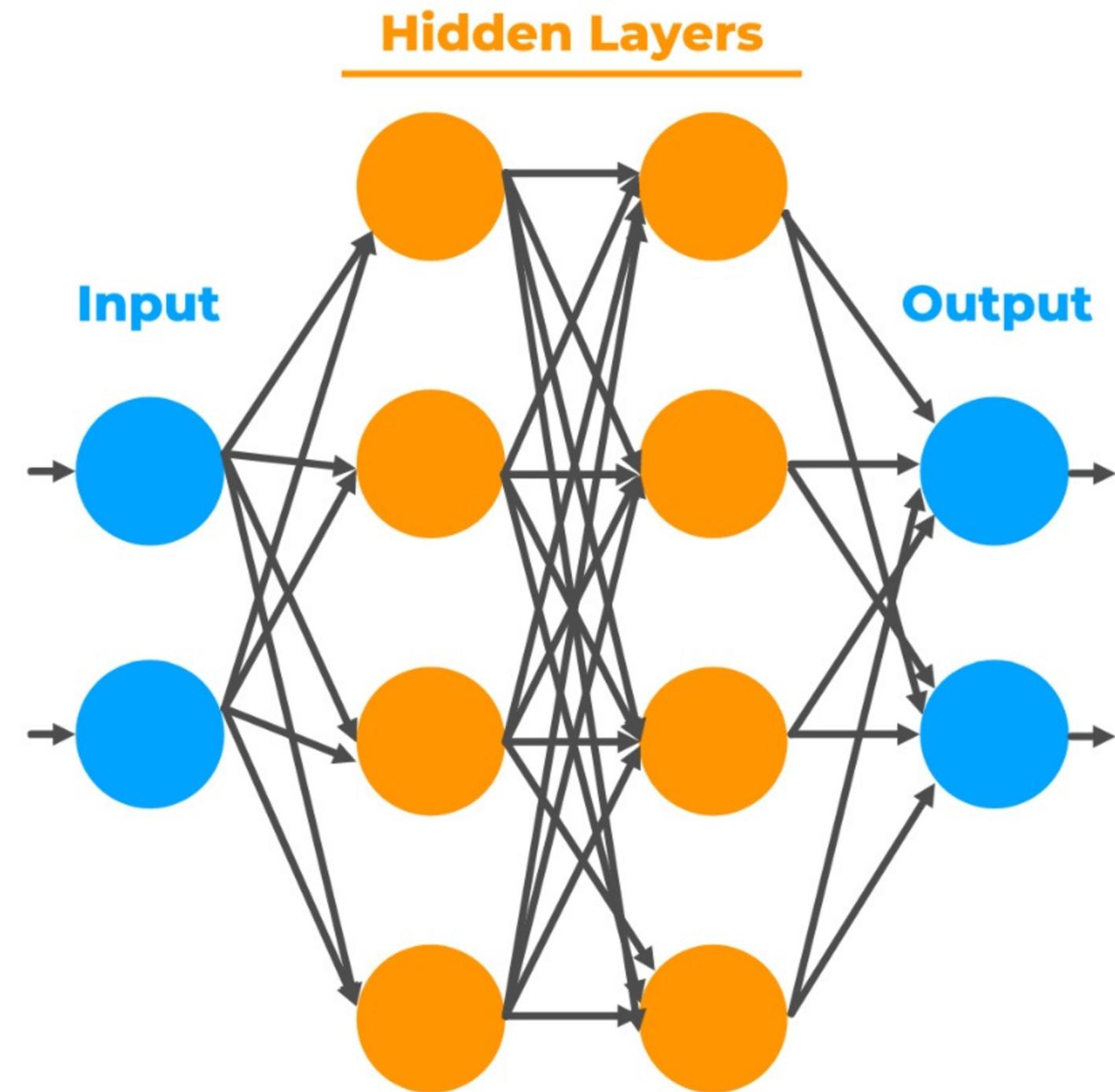


ICSI 436/536
MACHINE LEARNING

INSTRUCTOR PROF. CHONG LIU

PROJECT
PRESENTATION
GROUP 13

HANDWRITTEN DIGIT
RECOGNITION USING NEURAL
NETWORKS



AGENDA

- Dataset
- FNN Model
- Methods
- Results

PROBLEM SETUP

To train a neural network model that takes an image of a handwritten digit as input and predicts the correct digit label. Implemented by:

**Feed Forward Neural Network (FNN) or
MultiLayer Perceptron (MLP)**



DATASET

Dataset: MNIST

- 60,000 training images
- 10,000 test images
- Grayscale, 28x28 pixel images
- 10 classes (digits 0–9)

Preprocessing:

- Converted to tensors
- Normalized with dataset mean and standard deviation



DATASET EXAMPLE

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9



FNN MODEL

- Model: Multilayer Perceptron (MLP)
- **Input:** Flattened image 28x28 pixels
- **Hidden Layer 1:** 459 neurons + ReLU + Dropout
(regularization to reduce over fitting)
- Hidden Layer 2: 134 neurons + ReLU + Dropout
- **Output:** 10-class uses softmax
- **Training:** Optimized using: Adam optimizer +
CrossEntropy loss

HYPERPARAMETER

Tool: Optuna (automated tuning)

Optimized Parameters:

- Hidden sizes
- Learning rate
- Batch size

Best Configuration Found:

hidden_size1 = 459

hidden_size2 = 134

learning_rate = 0.00047

batch_size = 64

```
▶ model = Sequential([
    Flatten(input_shape=(28, 28, 1)),
    Dense(128, activation='relu'),
    Dense(64, activation='relu'),
    Dense(10, activation='softmax')
])
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
model.summary()
```

```
→ /usr/local/lib/python3.11/dist-packages/keras/src/layers/reshaping/flatten.py:37: UserWarning:
  super().__init__(**kwargs)
Model: "sequential"
```

Layer (type)	Output Shape	Param #
flatten (Flatten)	(None, 784)	0
dense (Dense)	(None, 128)	100,480
dense_1 (Dense)	(None, 64)	8,256
dense_2 (Dense)	(None, 10)	650

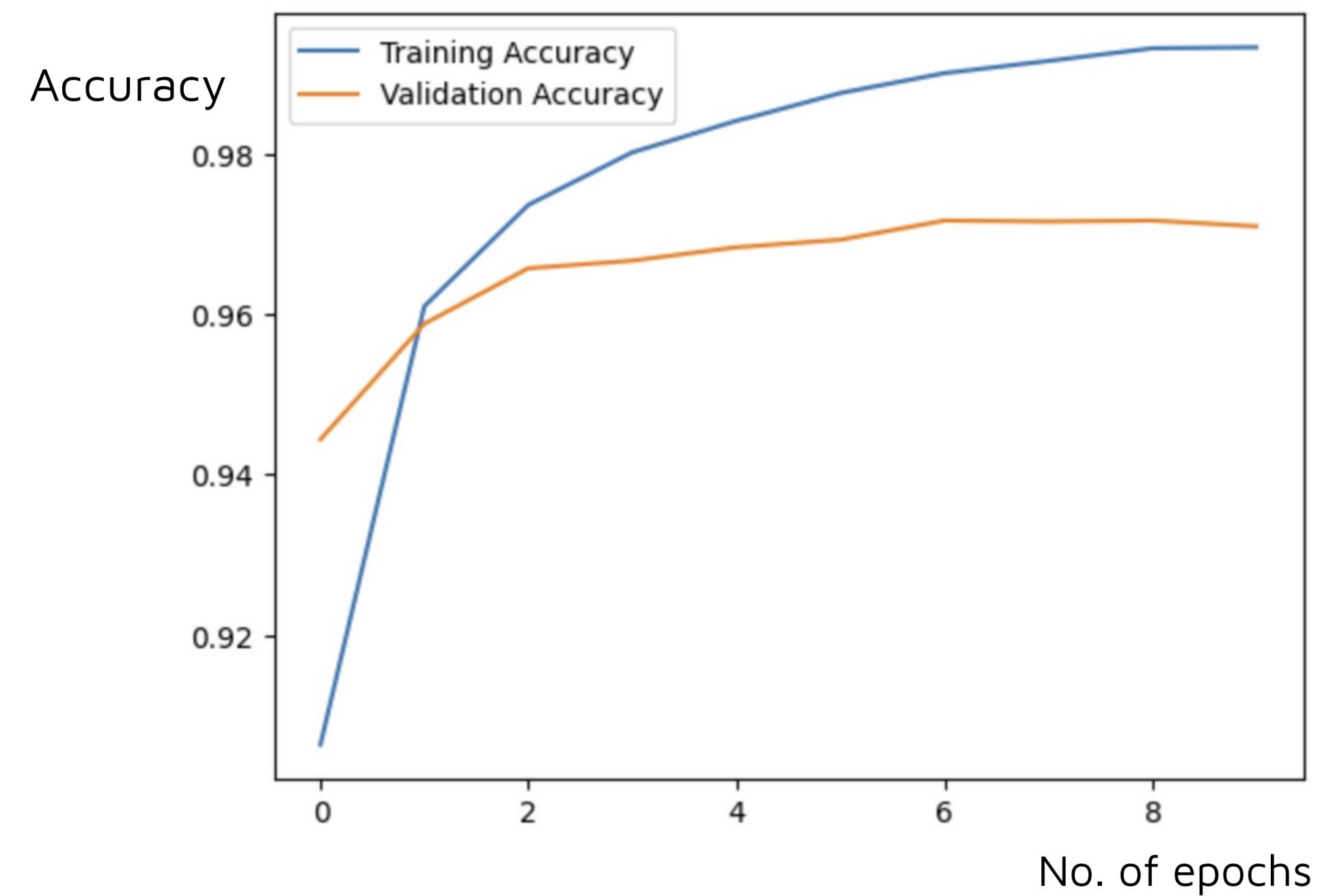
Total params: 109,386 (427.29 KB)

Trainable params: 109,386 (427.29 KB)

Non-trainable params: 0 (0.00 B)

RESULTS

Epoch 1, Loss: 0.2892
Epoch 2, Loss: 0.1233
Epoch 3, Loss: 0.0895
Epoch 4, Loss: 0.0699
Epoch 5, Loss: 0.0561
Epoch 6, Loss: 0.0470
Epoch 7, Loss: 0.0428
Epoch 8, Loss: 0.0364
Epoch 9, Loss: 0.0329
Epoch 10, Loss: 0.0304
Finished Training!



- Training: 10 epochs using best hyperparameters.
- Final Accuracy: Achieved 98.16% on test set

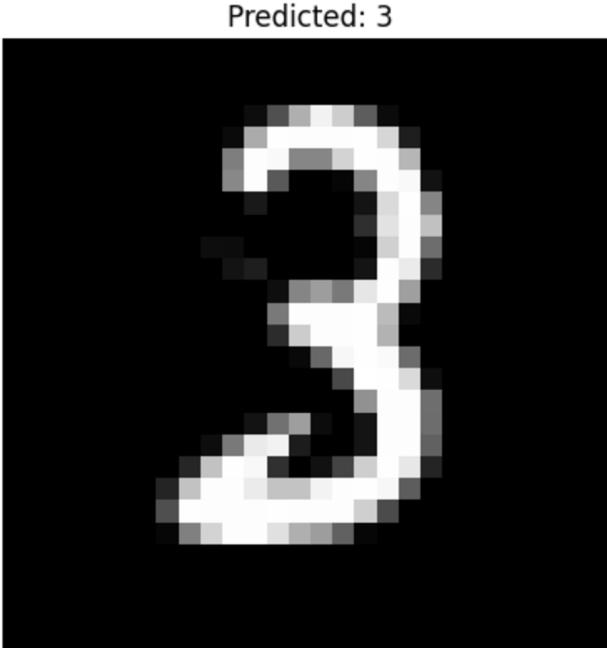
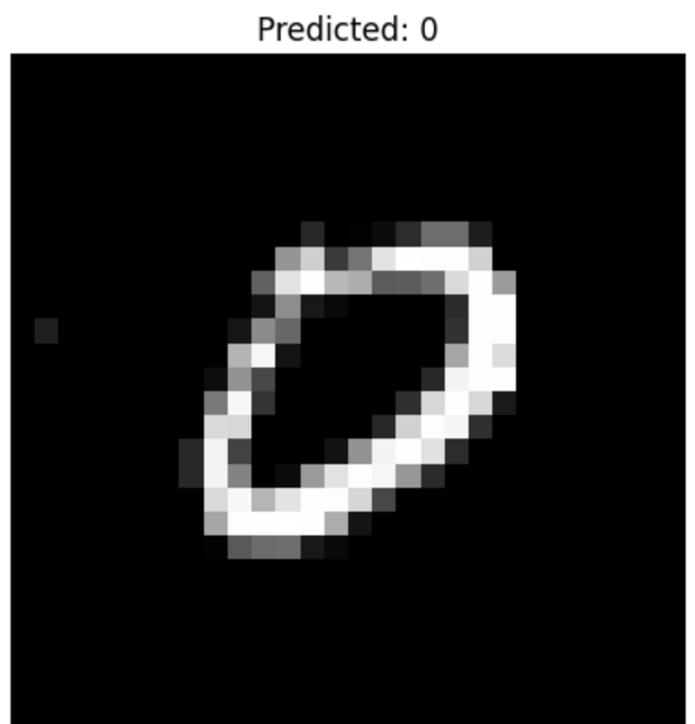
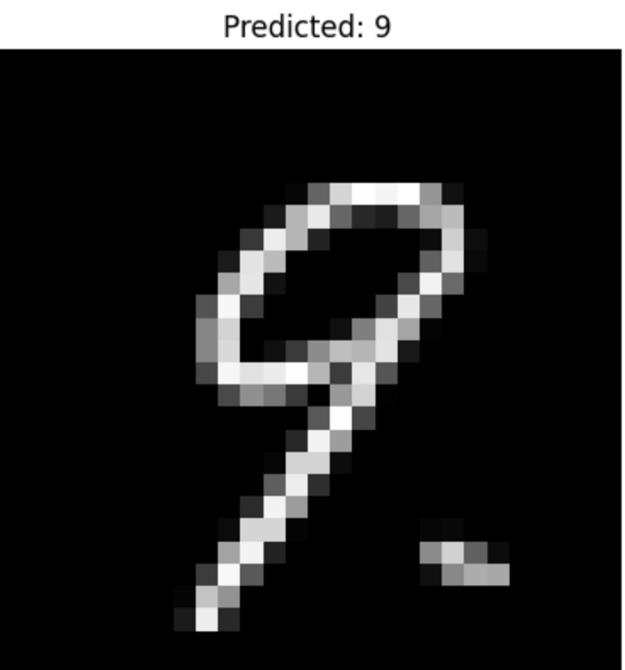
RESULTS

Fully Connected Neural Network

- Training Accuracy: ~98%
- Test Accuracy: ~97%

Observations:

- Good performance but struggled slightly with confusing digits (like '4' and '9').
- Slight overfitting noticed beyond 15 epochs without regularization.



TEAM MEMBERS

GROUP 13

- Narayani Kishor Khatavkar
- Harshini Narrra
- Snehith Reddy
- Jyotsana Parkhedkar
- Dileep Reddy Chinneluka

Thank You