

PROJECT ANN REPORT

**GROUP MEMBERS:**

Muhammad Shoaib Shahid (SP23-BAI-040)

Muhammad Adeel Tahir (SP23-BAI-029)

**SUBMITTED TO:**

DR. AKSAM IFTIKHAR

**SUBMISSION DATE:**

29-12-24

**DEPARTMENT OF COMPUTER SCIENCE**

**COMSATS UNIVERSITY ISLAMABAD, LAHORE CAMPUS**

**1. Dataset Description**

**MNIST** for classification models.

* Size: 60,000 training samples, 10,000 test samples.
* Features: 28x28 grayscale images.
* Classes: 10 categories from 0 to 9 digits picture.

California Housing for regression model.

* Size: Varies, typically around 20,000 samples.
* Features: 8 numerical features (e.g., median income, house age).
* Target: Median house value.

**2. Model Details**

**PyTorch ANN (Classification):**

* Architecture: 2 hidden layers with 64, 32 units, ReLU activations.
* Output: 10 units, softmax activation.

**PyTorch ANN (Regression):**

* Architecture: 2 hidden layers (64, 32 units), ReLU activations.
* Output: 1-unit, linear activation.

**Keras CNN (Classification):**

* Architecture: 2 Conv2D layers (32, 64 filters), MaxPooling, Flatten, Dense layers.
* Output: 10 units, softmax activation.

**3. Training Parameters:**

* Learning Rate: 0.01 for PyTorch ANN, 0.01 for ANN Regression.
* Batch Size: 64 for both models.
* Epochs: 10 for classification with ann and regression with ann , 750 for cnn.

**4. Graphs**

ANN CLASSIFICATION:

A graph of a line and a line graph

Description automatically generated with medium confidence

ANN REGRESSION:

A graph with a line graph

Description automatically generated

CNN CLASSIFICATION:

A graph of a line and a line

Description automatically generated with medium confidence

**5. Comparison Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Dataset/Task** | **Key Hyperparameter** | **Final Metrics** | **Training time** |
| ANN | Mnist/ Classification | LR =0.01, epochs=10 | Accuracy=96% | 5 min |
| ANN | California housing/ Regression | LR =0.01, epochs=10 | MSE=0.35284,  MAE=0.4130 | 6 min |
| CNN | Mnist/ Classification | LR=0.01, eopocs =750 | Accuracy=98% | 6 min |

**6. Conclusion:**

The PyTorch ANN is flexible we have custom implemetations but has lower accuracy on image data. The Keras CNN achieves high accuracy due to effective spatial feature extraction but requires more computational resources.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_