

Final Exam PROJECT REPORT

GROUP MEMBERS:

UMER ABBAS(SP23-BAI-053)

MUHMMMAD ZAIMAL ZIA(SP23-BAI-043)

SUBMITTED TO: DR. AKSAM IFTIKHAR

SUBMISSION DATE: 29-12-24

DEPARTMENT OF COMPUTER SCIENCE

COMSATS UNIVERSITY ISLAMABAD, LAHORE CAMPUS

**1. Dataset Description**

We have used **FashionMNIST** for classification models.

* Size: 60,000 training samples, 10,000 test samples.
* Features: 28x28 grayscale images.
* Classes: 10 categories:
* T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, Ankle boot.

We have used California Housing for regression model.

* Size: 20,640 samples.
* Features: 8 numerical features (MedInc, HouseAge, AveRooms, AveBedrms Population, Aveo cup, Latitude, Longitude)
* Target: Median house value.

**2. Model Details**

**PyTorch ANN (Classification):**

* Architecture: 2 hidden layers (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 Configurations:**

We have tried different combinations of epochs, optimizers (Adam and SGD) and learning rate out which these configurations perform well.

* Learning Rate: 0.01 for PyTorch ANN, 0.001 for ANN Regression.
* Batch Size: 64 for all models.
* Epochs: 10 for classification for both Ann and CNN, 1000 for regression.

**4. Performance Metrics:**

**Classification (PyTorch ANN):**

* Accuracy: 81.95%
* Precision :82%
* Recall :82%
* f1-score:82%

**Regression (PyTorch ANN):**

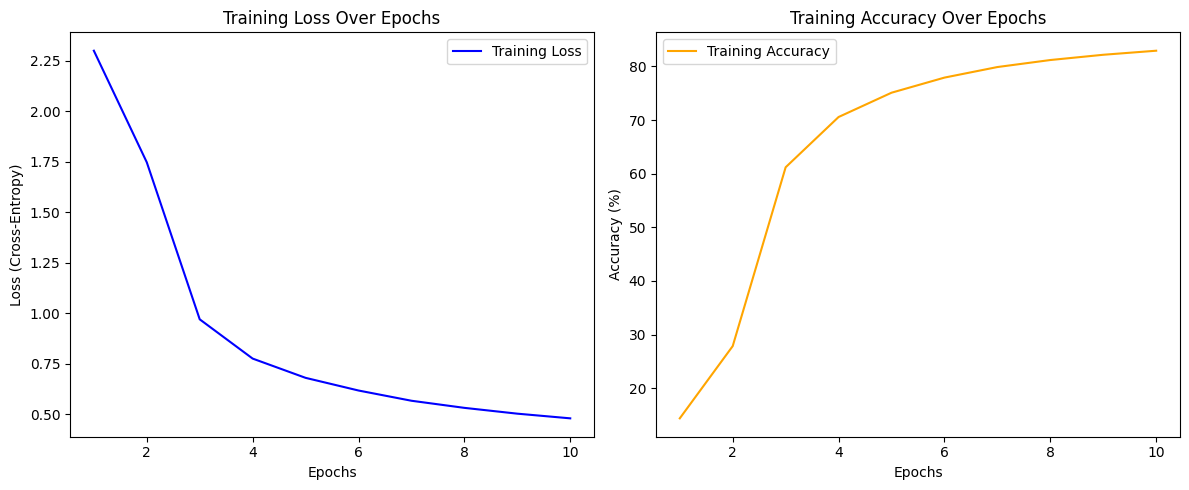
* MSE: 0.3319
* MAE: 0.3992
* R² Score: 0.7467

**Classification (Keras CNN):**

* Accuracy: 90.98%
* Precision :91%
* Recall :91%
* f1-score:91%

**5. Graphs And Confusion metrics:**

**ANN CLASSIFICATION:**



A diagram of a graph

Description automatically generated with medium confidence

**ANN Regression:**

A graph with a line

Description automatically generated

**CNN Classification:**

A graph of different colored lines

Description automatically generated with medium confidence

A diagram of clothing items

Description automatically generated with medium confidence

**6. Comparison Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Dataset** | **Key Hyperparameter** | **Final Metrics** | **Training time** |
| ANN Classification | Fashion Mnist | LR =0.01, epochs=10 | Accuracy=81% | 4 min |
| ANN Regression | California housing | LR =0.001, epochs=1000 | MSE=0.35284,  MAE=0.4130 | 2 min |
| CNN Classification | Fashion Mnist | LR=0.001, eopocs =10 | Accuracy=90% | 5 min |

**7. Conclusion:**

**Strengths:**

* PyTorch ANN: Flexibility in custom implementations.
* Keras CNN: High accuracy on image data due to spatial feature extraction.
* Good convergence rate for regression model.

**Weaknesses:**

* PyTorch ANN: Lower accuracy on image data compared to CNN.
* Keras CNN: More complex architecture requires more computational resources.

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