ARIZONA STATE UNIVERSITY

Assignment #1

Student name: *Kunal Sunil Kasodekar*

Course: *CSE 598 Advanced topics in machine learning security, privacy and fairness –*
Professor: *Chaowei Xiao*
Due date: *September 26th, 2022*

Environment/Setup:

WSL with Ubuntu 18.04 and Python 3.7 with the latest stable version of Pytorch is set up in a Conda virtual environment. The system has no GPU and makes use of an i5vPro 10 Gen CPU for training. However, for faster iterations in training and attacking the model, resolving some bugs and prototyping I shifted to Colab for GPU Access. The Colab notebook link is: https://colab.research.google.com/drive/14HIPB1Ehj2qafuD28-L1m3Y58fdIpXKU?usp=sharing. *Errors were resolved using Colab and the bugs were fixed in the local codebase.* The codebase is maintained using Git.

Submission Files/Format:

The output file contains the codebase for the whole assignment including the ipynb notebook (), py file, latex files used for submission and graphs.

Task 1: Train a Fashion-MNIST Classification

Task 1 was to train the Fashion-MNIST Dataset with the following properties: 10 Classes, 50,000 training images, 10,000 validation images and 10,000 test images. Lenet-5 with the architecture given below is used to train the dataset. The Architecture was obtained from the class slides, 5th presentation, page 45. The questions and answer w.r.t to the task are given below:

Architecture of the network:

LeNet-5

Layer	Output Size	Weight Size
Input	1 x 28 x 28	
Conv (C_{out}=20, K=5, P=2, S=1)	20 x 28 x 28	20 x 1 x 5 x 5
ReLU	20 x 28 x 28	
MaxPool(K=2, S=2)	20 x 14 x 14	
Conv (C_{out}=50, K=5, P=2, S=1)	50 x 14 x 14	50 x 20 x 5 x 5
ReLU	50 x 14 x 14	
MaxPool(K=2, S=2)	50 x 7 x 7	
Flatten	2450	
Linear (2450 -> 500)	500	2450 x 500
ReLU	500	
Linear (500 -> 10)	10	500 x 10

Answer. We make use of two 2D convolution layers, with:

- Kernel Size = 5x5
- Stride = 1
- Padding = 2
- Activation Function: Relu

A max-pooling layer of size 2 and stride 2 is applied after the convolutions. After the convolutions and pooling the tensor is flattened and passed onto fully connected linear layers with Layer1 output=500 and layer2 output=10 (The number of classes to detect) Relu is used as an Activation for all the trainable layers. Reference image for the Architecture is given above.

Receptive Field of the network:

- Inputs Local Receptive Field - 1x1
- Convolution 1 - 5x5
- Max Pool 1 - 10x10
- Convolution 2 - 14x14
- Max Pool 2 - 28x28

Hence Global Receptive field before the final FCN Layers is 28x28 implying that all the input pixels have been seen by the network.

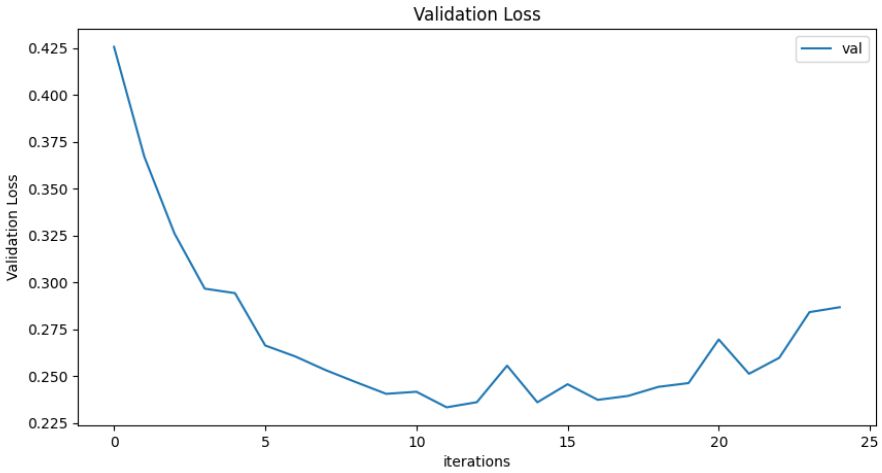
Optimizer and parameters:

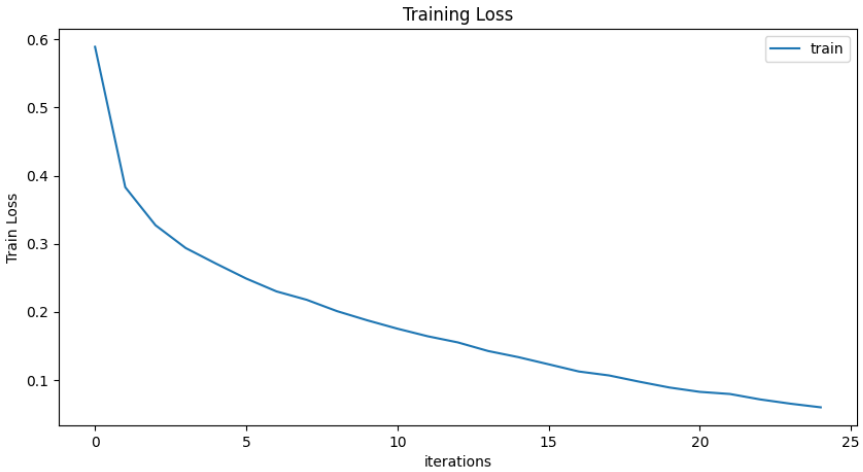
Answer. After trying out all optimizers like SGD, RMSProp and Adam it was found that Adam gave the lowest cross-entropy loss on validation and the best accuracy on the validation dataset. Other Optimal Hyperparameters including training parameters for the model are:

- Learning Rate = 3e-4
- Weight Decay = 1e-4
- Number of Epochs = 25
- Batch size = 64

Training and Validation Plots:

Answer.





Best Model Accuracy:

Answer. After multiple experiments, the best accuracy on the Test Dataset is **91.83** Percent. The corresponding accuracy on the validation dataset is 92.25 Percent. The model is saved as Lenet5-FashionMNISTtrialv2.pt in the model folder using torch.save().

Task 2: Implement attacks for the Fashion-MNIST Classification

The Attack success rate of the FGSM attack (Epsilon = 25/255) on Test Dataset is:

Answer. The Attack success rate is: **86.36** Percent

The Attack success rate of the PGD attack (Epsilon = 25/255) on different attack steps:

Answer. The Attack success rate for respective attack steps is:

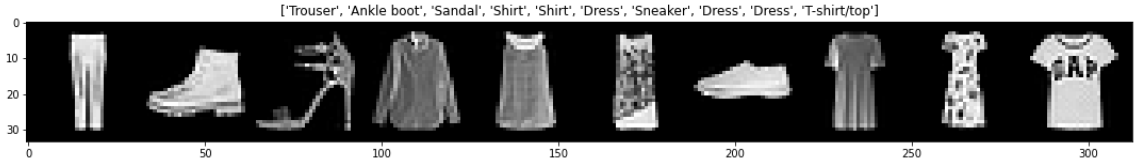
- 1: 20.19 Percent
- 2: 37.24 Percent

- 5: 75.05 Percent
- 10: **96.97** Percent

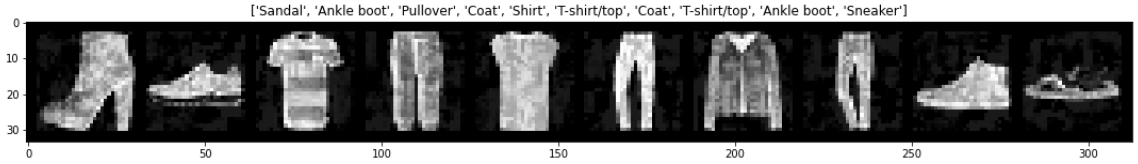
Visualizing the 10-step PGD attack using 10 Adversarial Images (Test)

Answer.

Random Images Predicted by LeNet



PGD (n=10) Attacked Images with Predicted Labels



PGD (n=10) Attacked Images with Actual Labels