Non-Targeted White-Box Evasion Attacks on the Fashion MNIST Dataset

Kyle Otstot

September 26, 2022

1 Training a LeNet-5 CNN on Fashion MNIST

In this portion of the report, I trained a model to classify images from the Fashion MNIST dataset. Specifically, I used the LeNet-5 model architecture along with other standard hyperparameter choices. To determine which setting performed best on Fashion MNIST, I implemented a grid search to test all possible combinations with the help of ASU's GPU cluster. The grid search– including hyperparameters and tested values– are reported in Table 1 below.

Type	Setting	Values
Network	Pooling	**Average**, Max
	Activation	Tanh, **ReLU**
	Dropout	0.1, **0.2**, 0.3
Training	Batch size	**32**, 64, 128
	Number of epochs	**100**
Optimization	Algorithm	**Adam**, SGD
	Learning rate	5e-4, **1e-3**, 2e-3, 5e-3, 1e-2
	Weight decay	**1e-4**, 1e-3, 1e-2

Table 1: Grid search of hyperparameters

The combination of hyperparameters that achieved the best test accuracy is highlighted in **bold** font. Additionally, the LeNet-5 architecture is outlined in Table 2, and the training + validation loss is plotted against time in Figure 2. This figure shows that our choice of number of epochs is appropriate, since the two losses appear to converge. With the best combination of hyperparameters, the LeNet-5 achieved a test accuracy of **92.05%**. I believe that if we used more feature maps in the convolution layers, we could increase the test accuracy further; however, we kept the general LeNet architecture fixed and focused on more specific hyperparameters.

#	Layer	Setting
1	Conv2D	In channels = 1, out channels = 6, kernel = (5 x 5), padding = 2
2	ReLU	x
3	Avg. Pool	Kernel = (2 x 2), stride = 2
4	Conv2D	In channels = 6, out channels = 16, kernel = (5 x 5), padding = 2
5	ReLU	x
6	Avg. Pool	Kernel = (2 x 2), stride = 2
7	Flatten	x
8	Dropout	probability = 0.2
9	Linear	In features = 400, out features = 120
10	ReLU	x
11	Dropout	probability = 0.2
12	Linear	In features = 120, out features = 84
13	ReLU	x
14	Dropout	probability = 0.2
15	Linear	In features = 84, out features = 10

Table 2: LeNet-5 architecture

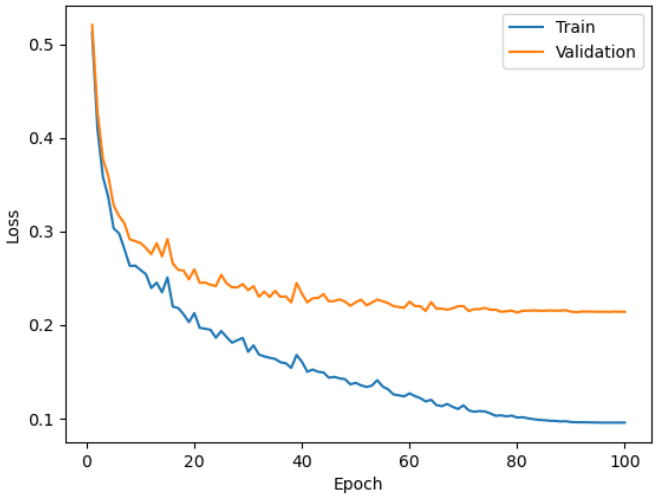


Figure 1: Train and validation loss reported over time

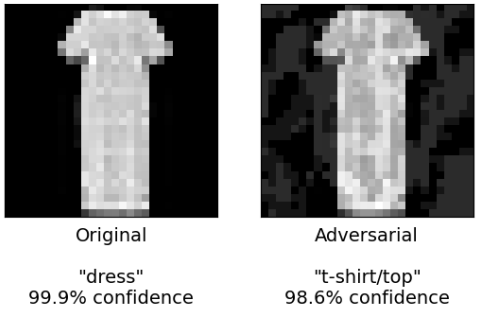
2 Attacking the CNN with FGSM and PGD

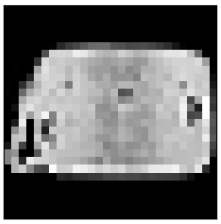
In this portion of the report, I implemented two adversarial attacks– *fast gradient sign method (FGSM)* and *projected gradient descent (PGD)*. The first attack is parameterized by ϵ, while the second attack is defined by ϵ, α and n. For simplicity, I set $\alpha = \epsilon = 25/255$ and received very good results from it. In Table 3, I report a series of metrics for the original images, images attacked by FGSM, and images attacked by PGD of varying steps. We can see that as the number of steps for PGD increases, the attack becomes more effective: for example, the original images are classified correctly 92.05% of the time, but a 10-step PGD attack reduces the accuracy to 0.3%. The success rates for each attack can be interpreted as the number of label flips, which are detailed in the last four rows of the table.

			PGD			
Metrics	Original	FGSM	steps = 1	2	5	10
Loss	0.224	3.85	3.85	6.95	10.78	11.89
Accuracy	0.921	0.185	0.185	0.046	0.007	0.003
% of C $\rightarrow$ C	x	0.201	0.201	0.050	0.008	0.003
% of C $\rightarrow$ I	x	0.799	0.799	0.950	0.992	0.997
% of I $\rightarrow$ C	x	0.004	0.004	0	0	0
% of I $\rightarrow$ I	x	0.996	0.996	1	1	1

Table 3: Metrics for both adversarial attacks. "% of C $\rightarrow$ I" = portion of correctly-classified images that were incorrectly classified after being attacked.

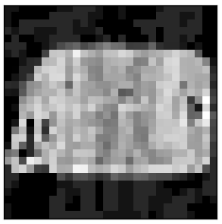
Lastly, I recorded qualitative data for ten randomly selected test images attacked by the 10-step PGD algorithm, found below.





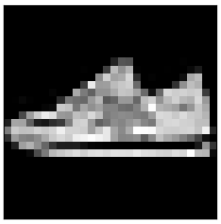
Original

"bag"
100.0% confidence



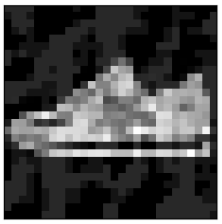
Adversarial

"shirt"
99.7% confidence



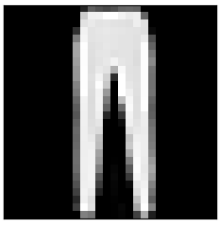
Original

"sneaker"
100.0% confidence



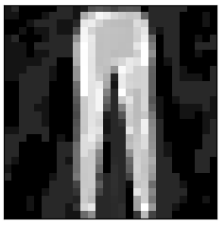
Adversarial

"coat"
99.7% confidence



Original

"trouser"
100.0% confidence



Adversarial

"dress"
100.0% confidence

Original

"t-shirt/top"
98.7% confidence

Adversarial

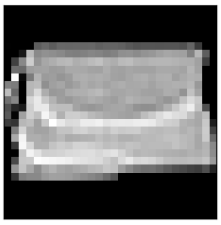
"shirt"
99.8% confidence

Original

"pullover"
99.1% confidence

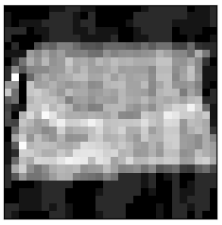
Adversarial

"coat"
50.7% confidence



Original

"bag"
100.0% confidence

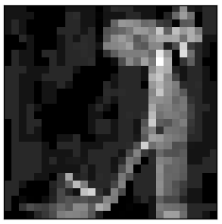


Adversarial

"shirt"
95.4% confidence

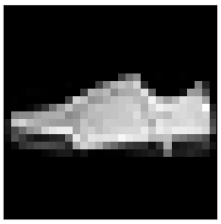
Original

"sandal"
100.0% confidence



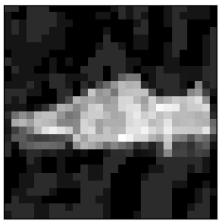
Adversarial

"ankle boot"
100.0% confidence



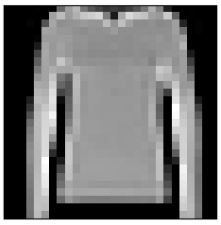
Original

"sneaker"
100.0% confidence



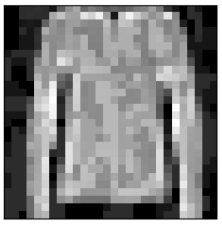
Adversarial

"sandal"
99.9% confidence



Original

"pullover"
98.5% confidence



Adversarial

"coat"
99.3% confidence