Paper: <https://arxiv.org/pdf/1808.05537.pdf>

code: <https://github.com/tianzheng4/Distributionally-Adversarial-Attack>

1. From terminal/cmd:

**ssh** [**daryln@10.4.0.15**](mailto:daryln@10.4.0.15) **or ssh daryln@<server ip address>**

1. In server
   1. If container is not already running

command to run:

**docker run -it -p 8888:8888 --name daa\_attack -v /home/daryln/adversarial\_attacks:/home --gpus=1 tensorflow/daa\_attack:latest**

* 1. If container is already running (should already be named)
     1. **docker start daa\_attack**
     2. **docker attach daa\_attack**

1. DAA attack directory is under /home as Distributionally-Adversarial-Attack
2. **mnist-challenge-master** and **fashion-mnist-master** contains the mnist and fashion-mnist dataset with the DAA attacks in them
3. **cd mnist-challenge-master**
4. Run attack using

**python blob\_rand.py** or **python dgf\_rand.py** (for the 2 different variations of DAA: DAA-BLOB and DAA-DGF)

1. An **attack.npy** file will be generated from running either of the above attacks. This **attack.npy** file contains the adversarial images of the respective imageset used (MNIST or fashion-MNIST)

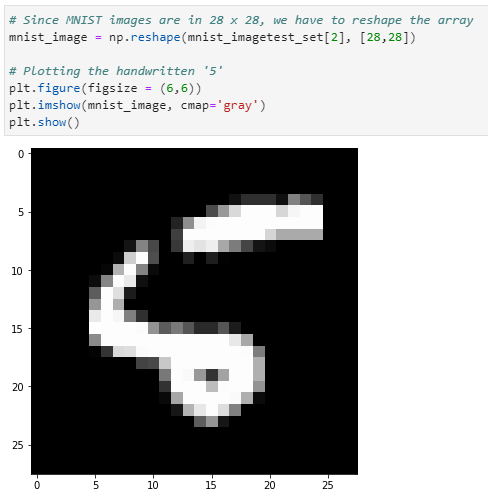
To view the imageset

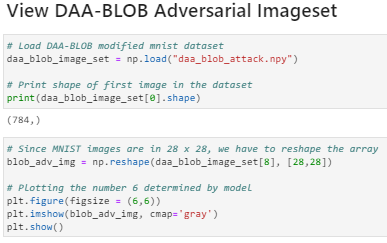
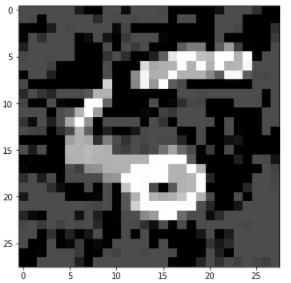
The adversarial images are stored in an imageset which is converted into a numpy array (hence .npy). To view them, it is recommended to open a Jupyter notebook for this. There is already one generated in the directory (**view\_adv\_img.ipynb)**.

To use Jupyter

Command to run: **jupyter lab --no-browser --ip=0.0.0.0 --port=8888 --allow-root**

Open the notebook and the code should already be written for you.

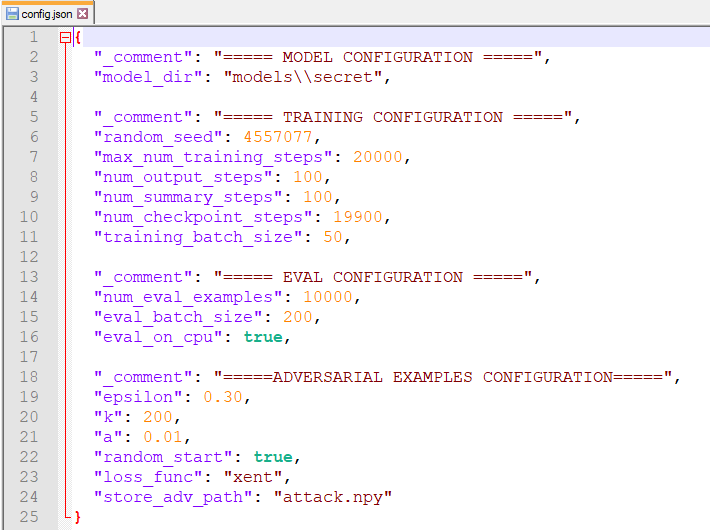
 

1. To evaluate the attack

**python run\_attack.py**

1. To evaluate a neural network

Edit the **config.json** file



Select the model you want under model\_dir (use **python fetch\_model.py** to acquire the type of model you want to use for testing/training)

Under *store\_adv\_path*, change the file name to **daa\_blob\_attack.npy** or **daa\_dgf\_attack.npy** to evaluate these adversarial images.

Then enter the command **python run\_attack.py** and a **pred.npy** file will be created

1. This **pred.npy** indicates the model predictions on the adversarial images
2. Accuracy of each variation

**DAA-BLOB DAA-DGF**

