**Adverserial Learning – Assigmnent 1 – Eitamar Saraf, Moshe Nasletashvili**

1. Training Neural Network on MNIST:

We took an MLP with one hidden layer (256 neurons), to classify the input image (784 pixels) to 10 classes.

The parameters of the training are as follows:

# of epochs: 15

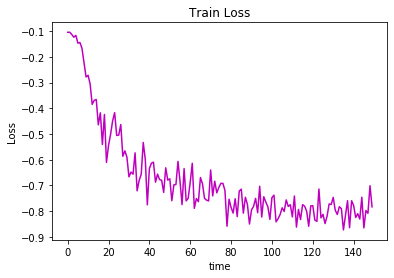
Batch size: 64

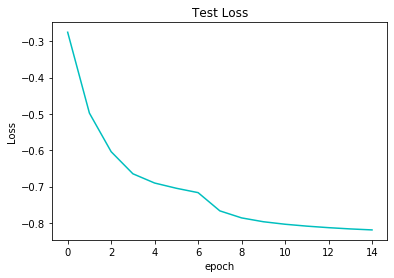
Learning rate: 3e-3

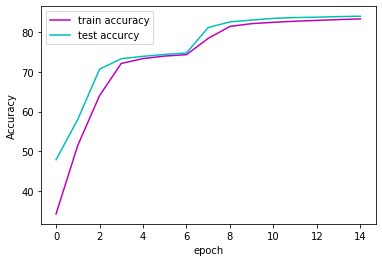
Weight decay: 1e-4

Loss and accuracy over training:

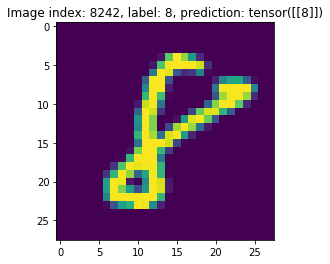
Train Accuracy: 83.4%, Test Accuracy: 84.1%



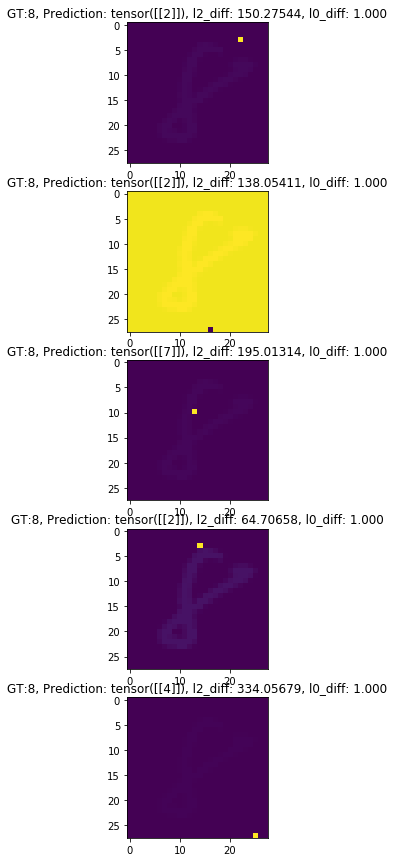
****



1. The image taken randomly:



3-4. The adversarial examples with their false predictions, and metrics:



We can see that the closest the perturbated pixel to the digit itself , the easier it is to make the model confuse. That's cool! It indicates that the regions that count more are the ones with the digit on them, and not the background.

5+6. We trained an SVM model on 10^4 samples of MNIST (we took less examples due to the fact that SVM is a smaller model so there is no need for that amount of data. In addition, the hinge loss function is much harder to optimize on that amount of data.

As for metrics on the dataset, we took 10^3 samples from the test set to get the accuracy on the test set. The results are:

Train Accuracy: 98.1%, Test Accuracy: 92.7%

The classifier classified correctly the picked image from section 2, and misclassified the perturbated images, but misclassified them differently:

Results are in the next page:

