Interpretability for Detecting Adversarial Attacks

Task: MNIST digit recognition

Terminology:

Task Network – The model performing the digit recognition task.

Task Input Images – Input to the task network. Either unaltered images from the MNIST data set or adversarial attack images generated using one of three adversarial attack method.
(LBFGSAttack, DeepFoolAttack, GradientAttack from: https://github.com/bethgelab/foolbox)

Relevancy Map Image - Relevancy map decision explanations from classifications made by the task network on task input images. (Generated by LRP Deep Taylor Composition) - https://github.com/VigneshSrinivasan10/interprettensor

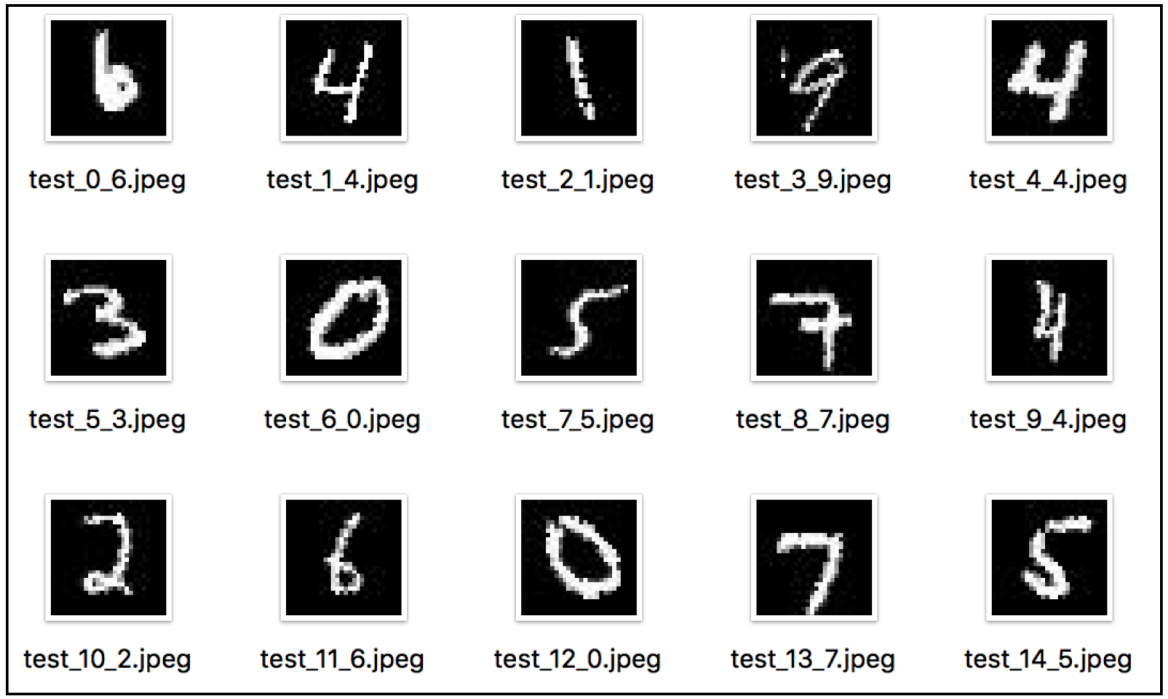
Input Image Adversarial Classifier (Input Classifier) – A model trained on the task input images being fed to the task network (so either an original MNIST image or an adversarial attack image).

Relevancy Map Adversarial Classifier (Relevancy Classifier) – A model trained on the relevancy maps generated from the task network classifications.

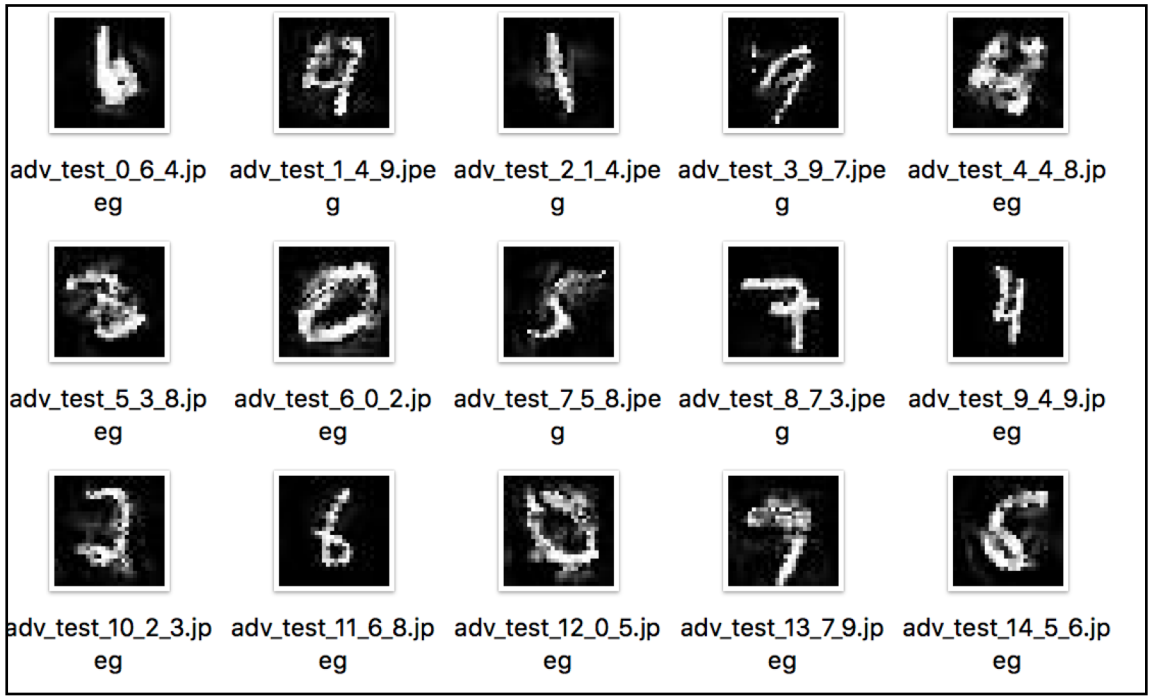
Interpretability for Detecting Adversarial Attacks

Task Input Images - LBFGSAttack

Original Images (file format = test_[id]_[ground_truth].jpeg



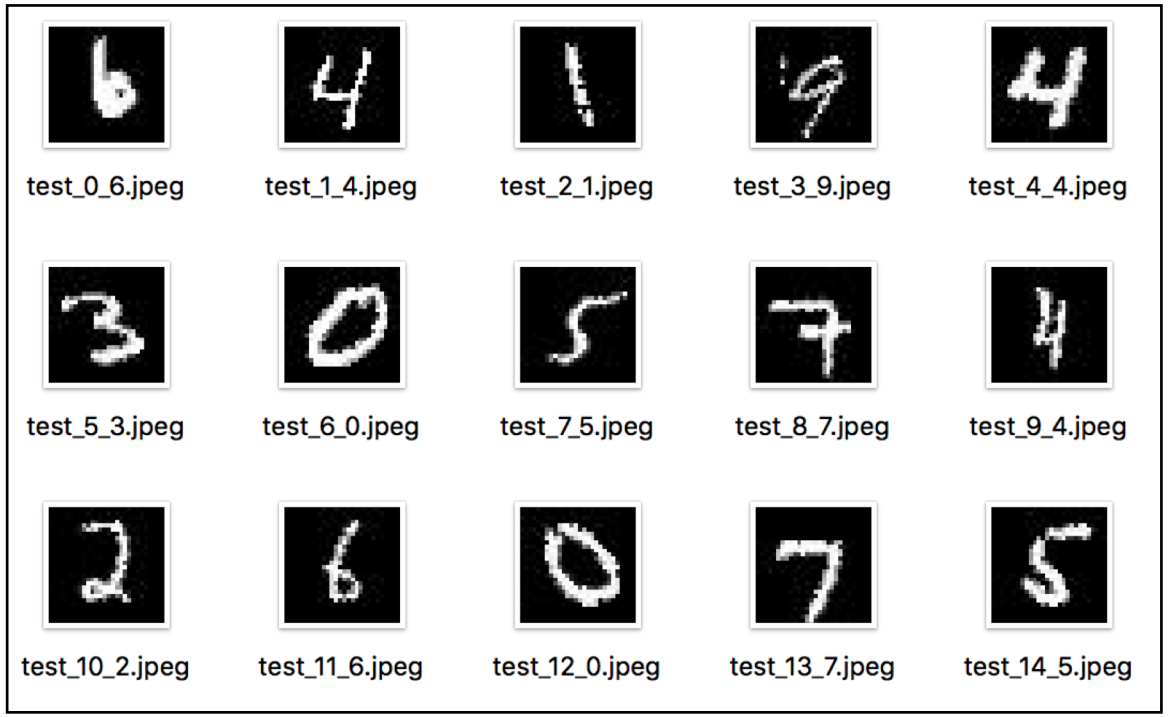
Adversarial Images (adv_test_[id]_[ground_truth]_[classification].jpeg



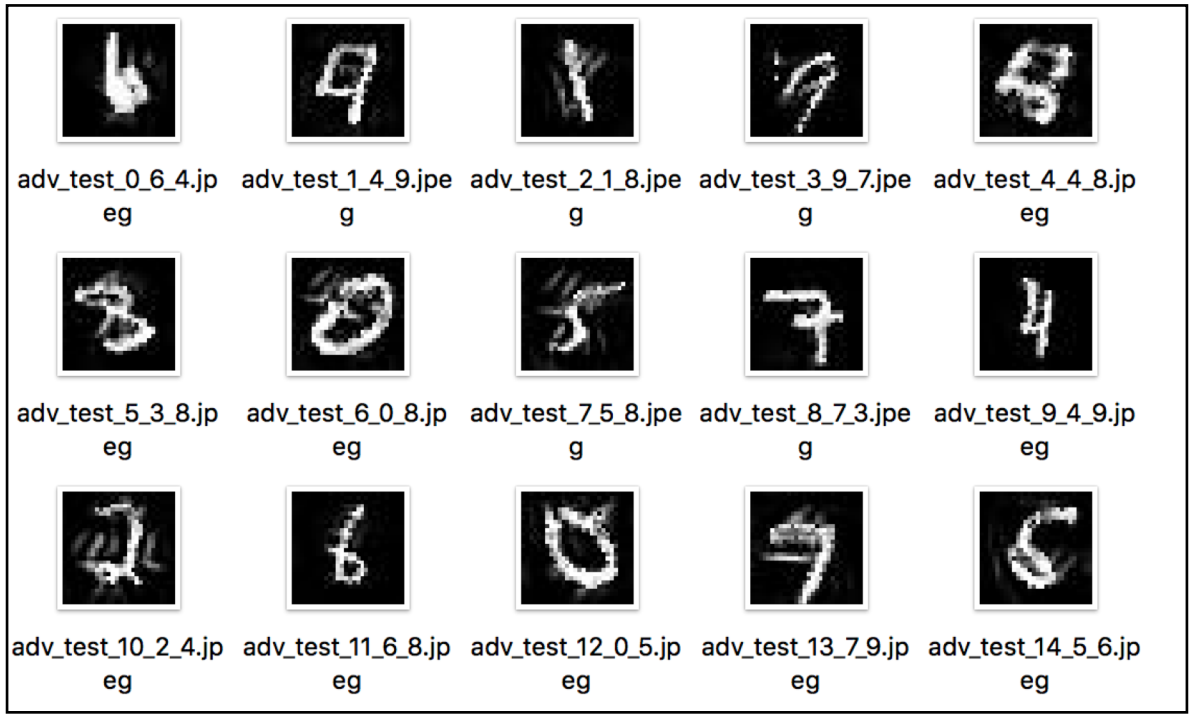
Interpretability for Detecting Adversarial Attacks

Task Input Images – DeepFoolAttack

Original Images (file format = test_[id]_[ground_truth].jpeg



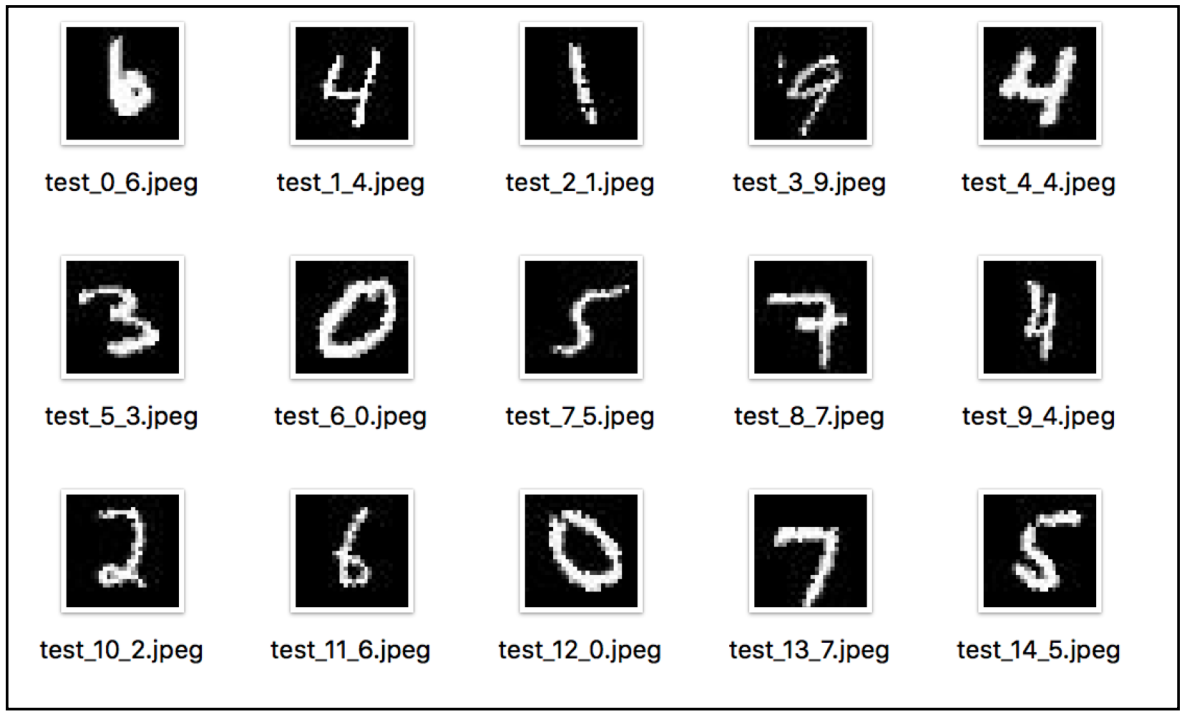
Adversarial Images (adv_test_[id]_[ground_truth]_[classification].jpeg



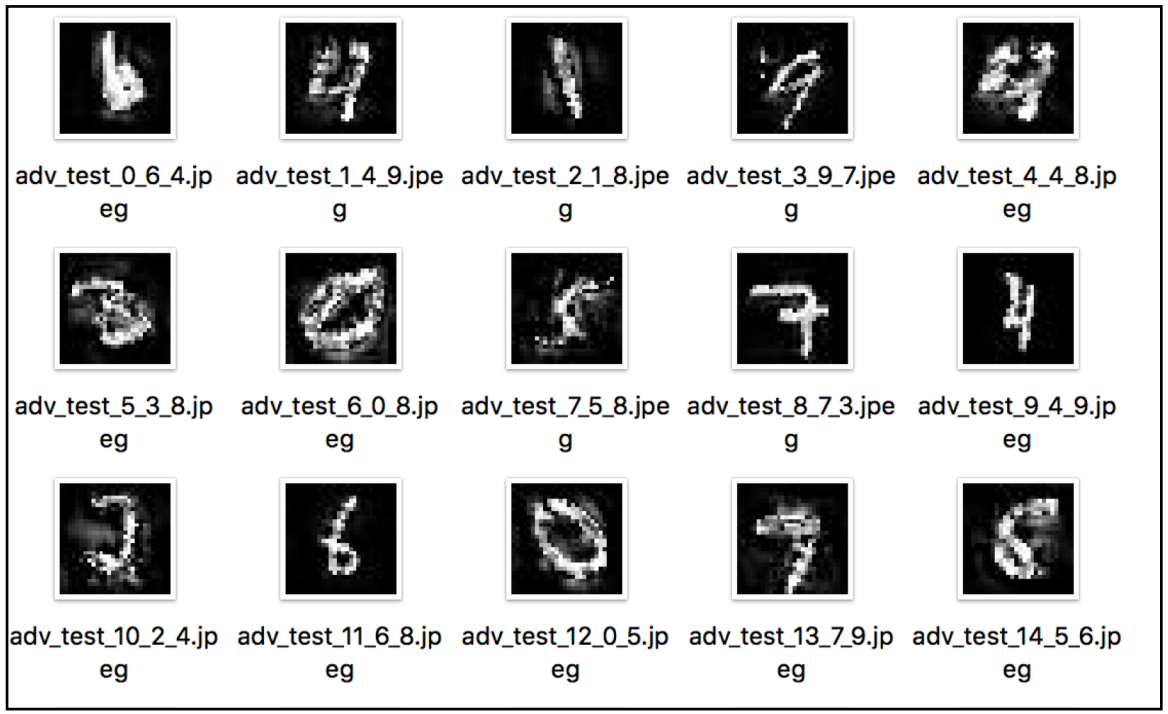
Interpretability for Detecting Adversarial Attacks

Task Input Images - GradientAttack

Original Images (file format = test_[id]_[ground_truth].jpeg



Adversarial Images (adv_test_[id]_[ground_truth]_[classification].jpeg



Interpretability for Detecting Adversarial Attacks

Task Input Imageas

Some attack images seem of a quality that would stop them being a viable attack.

Do we improve the mnist attack algorithms (fine tuning, try another algorithm, etc)?

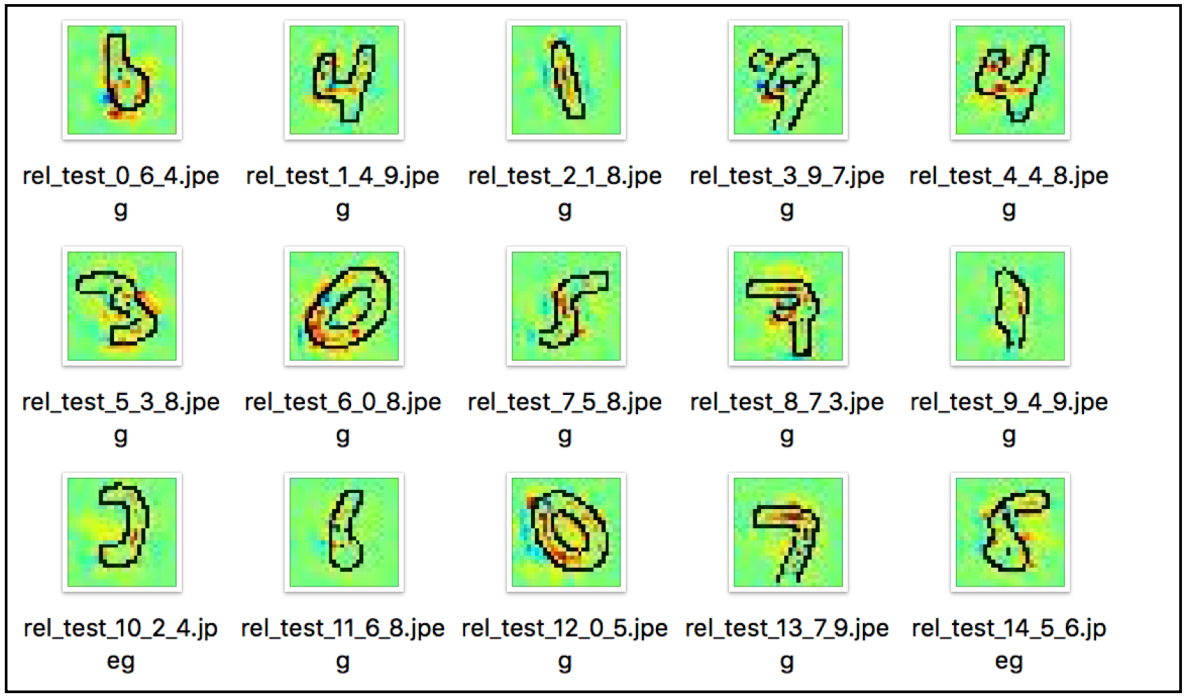
Do we change task to something that will generate attacks which are less detectable to the human eye?

Interpretability for Detecting Adversarial Attacks

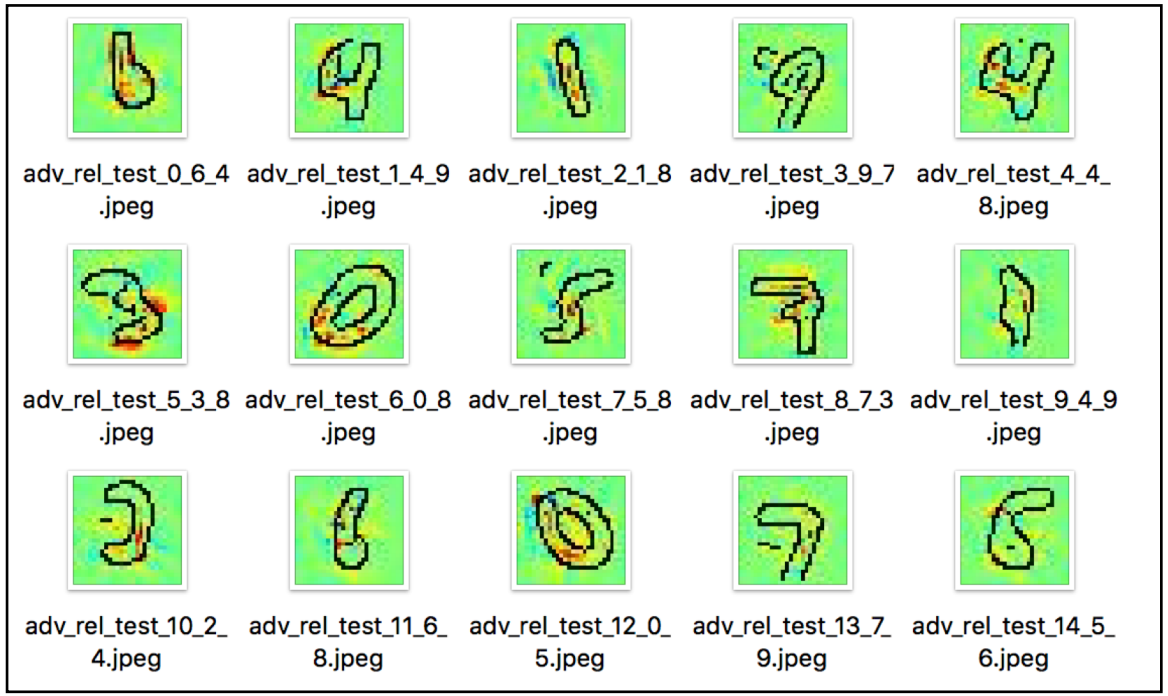
Relevancy Images - LBFGSAttack

Red is evidence for the class, Blue is evidence against the class, Green is neutral

Original Images (file format = rel_test_[id]_[ground_truth].jpeg



Adversarial Images (adv_rel_test_[id]_[ground_truth]_[classification].jpeg

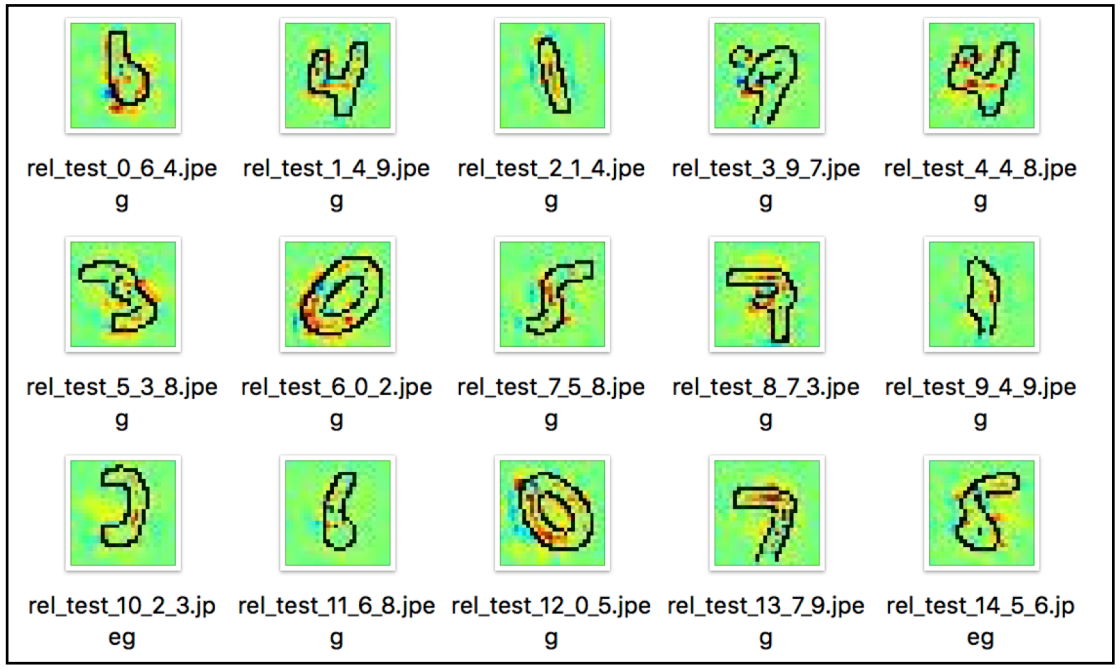


Interpretability for Detecting Adversarial Attacks

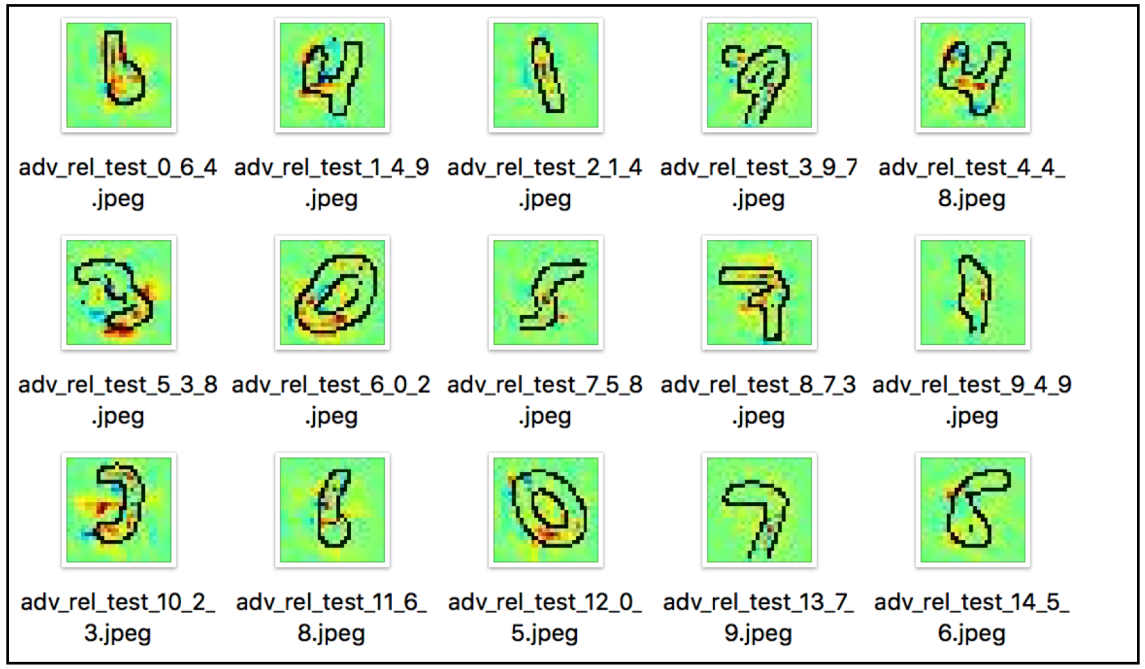
Relevancy Images - DeepFoolAttack

Red is evidence for the class, Blue is evidence against the class, Green is neutral

Original Images (file format = rel_test_[id]_[ground_truth].jpeg



Adversarial Images (adv_rel_test_[id]_[ground_truth]_[classification].jpeg

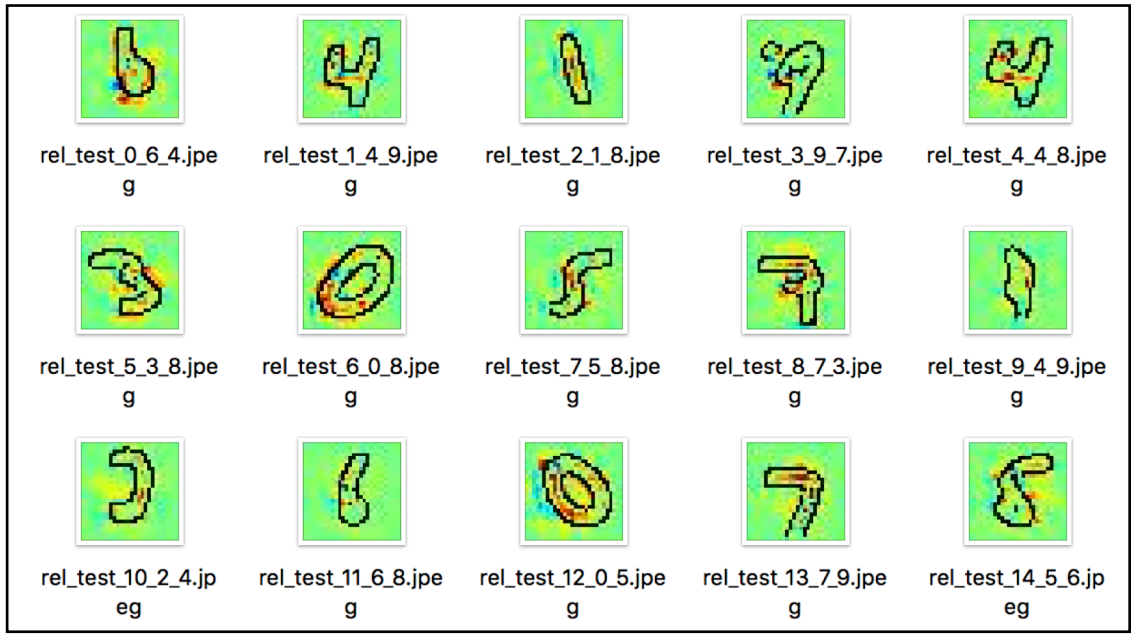


Interpretability for Detecting Adversarial Attacks

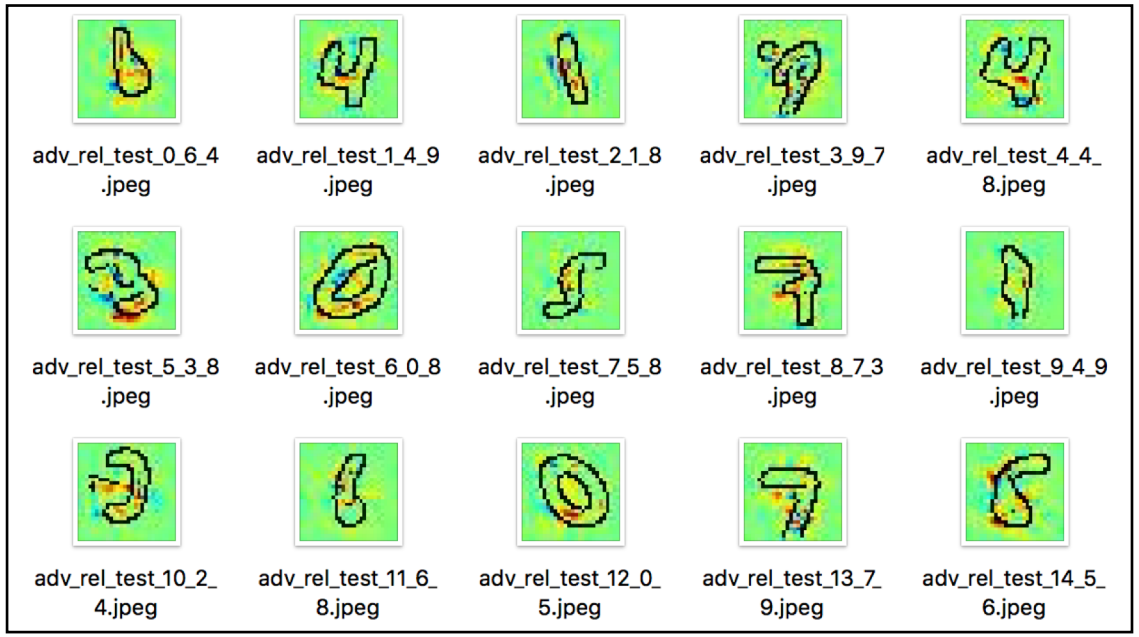
Relevancy Images - GradientAttack

Red is evidence for the class, Blue is evidence against the class, Green is neutral

Original Images (file format = rel_test_[id]_[ground_truth].jpeg



Adversarial Images (adv_rel_test_[id]_[ground_truth]_[classification].jpeg



Interpretability for Detecting Adversarial Attacks

Input Image Adversarial Classifier (Input Classifier)

```
layers = [
    imageInputLayer([28 28 1])

    convolution2dLayer(3,8,'Padding',1)
    batchNormalizationLayer
    reluLayer

    maxPooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,16,'Padding',1)
    batchNormalizationLayer
    reluLayer

    maxPooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,32,'Padding',1)
    batchNormalizationLayer
    reluLayer

    fullyConnectedLayer(2)
    softmaxLayer
    classificationLayer];
```

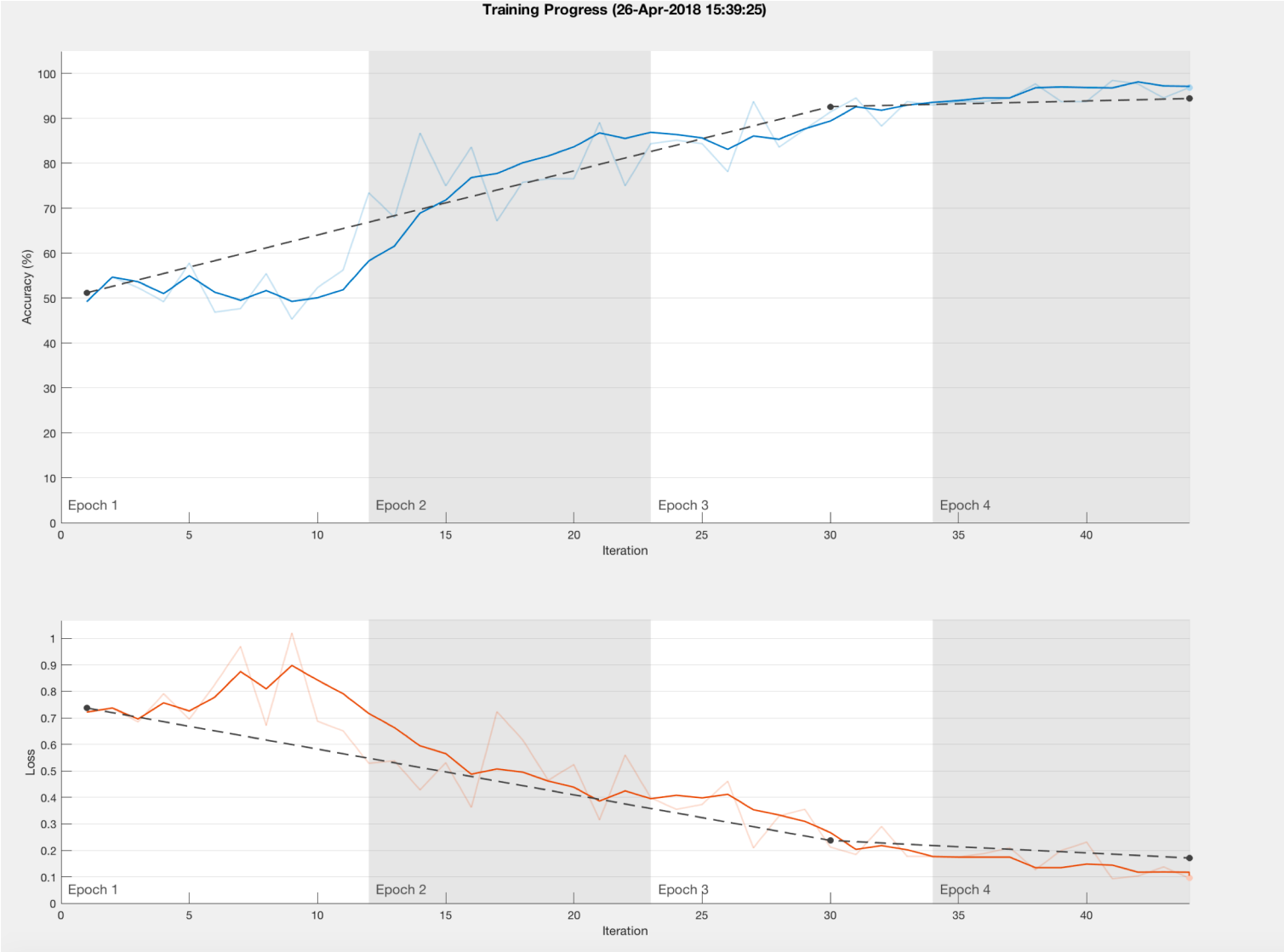
Relevancy Map Adversarial Classifier (Relevancy Classifier)

```
layers = [
    imageInputLayer([28 28 3])

    convolution2dLayer(3,8,'Padding',1)
    batchNormalizationLayer
    reluLayer

    maxPooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,16,'Padding',1)
    batchNormalizationLayer
    reluLayer

    maxPooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,32,'Padding',1)
    batchNormalizationLayer
    reluLayer

    fullyConnectedLayer(2)
    softmaxLayer
    classificationLayer];
```

Other structures were tried for Relevancy Classifier (very small gains (if any) were seen to the upcoming shown results)

Interpretability for Detecting Adversarial Attacks

Input Classifier – LBFGSAttack - 94.4%



Results

Validation accuracy:	94.40%
Training finished:	Reached final iteration

Training Time

Start time:	26-Apr-2018 15:39:25
Elapsed time:	13 sec

Training Cycle

Epoch:	4 of 4
Iteration:	44 of 44
Iterations per epoch:	11
Maximum iterations:	44

Validation

Frequency:	30 iterations
Patience:	5

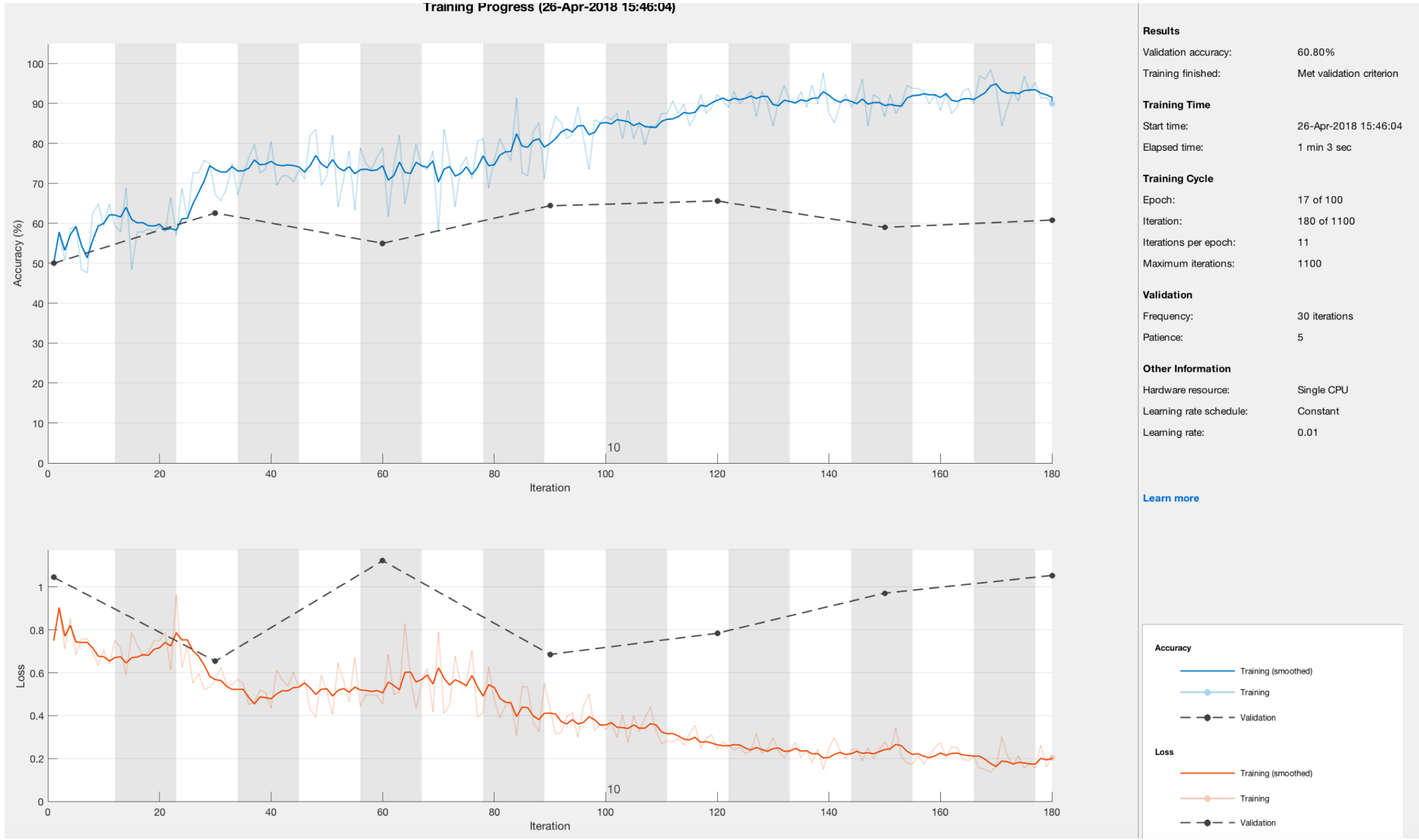
Other Information

Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.01

Learn more

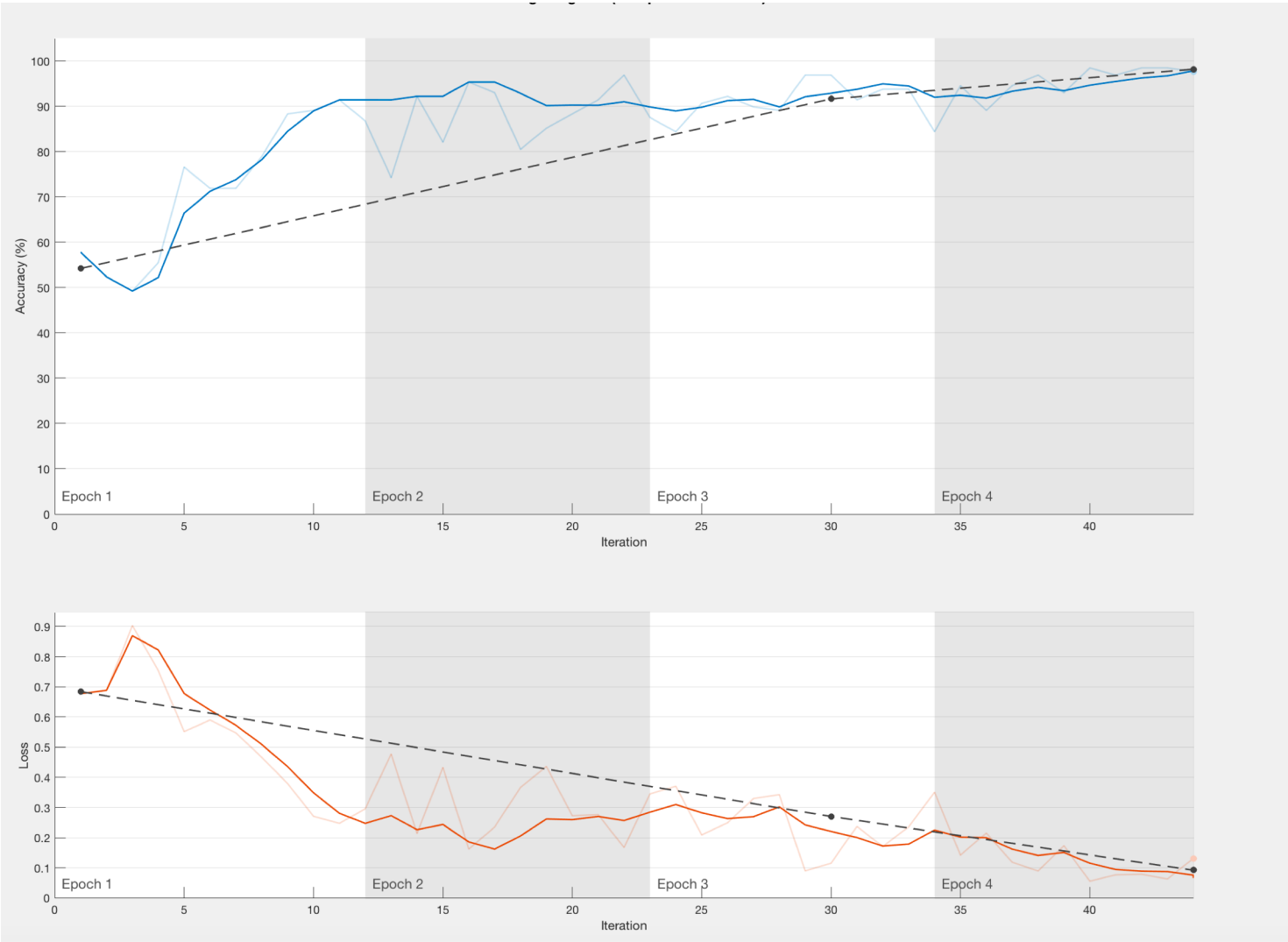
Interpretability for Detecting Adversarial Attacks

Relevancy Classifier – LBFGSAttack – 60.8%



Interpretability for Detecting Adversarial Attacks

Input Classifier – DeepFoolAttack – 98.16%



Results

Validation accuracy:	98.16%
Training finished:	Reached final iteration

Training Time

Start time:	26-Apr-2018 15:26:46
Elapsed time:	13 sec

Training Cycle

Epoch:	4 of 4
Iteration:	44 of 44
Iterations per epoch:	11
Maximum iterations:	44

Validation

Frequency:	30 iterations
Patience:	5

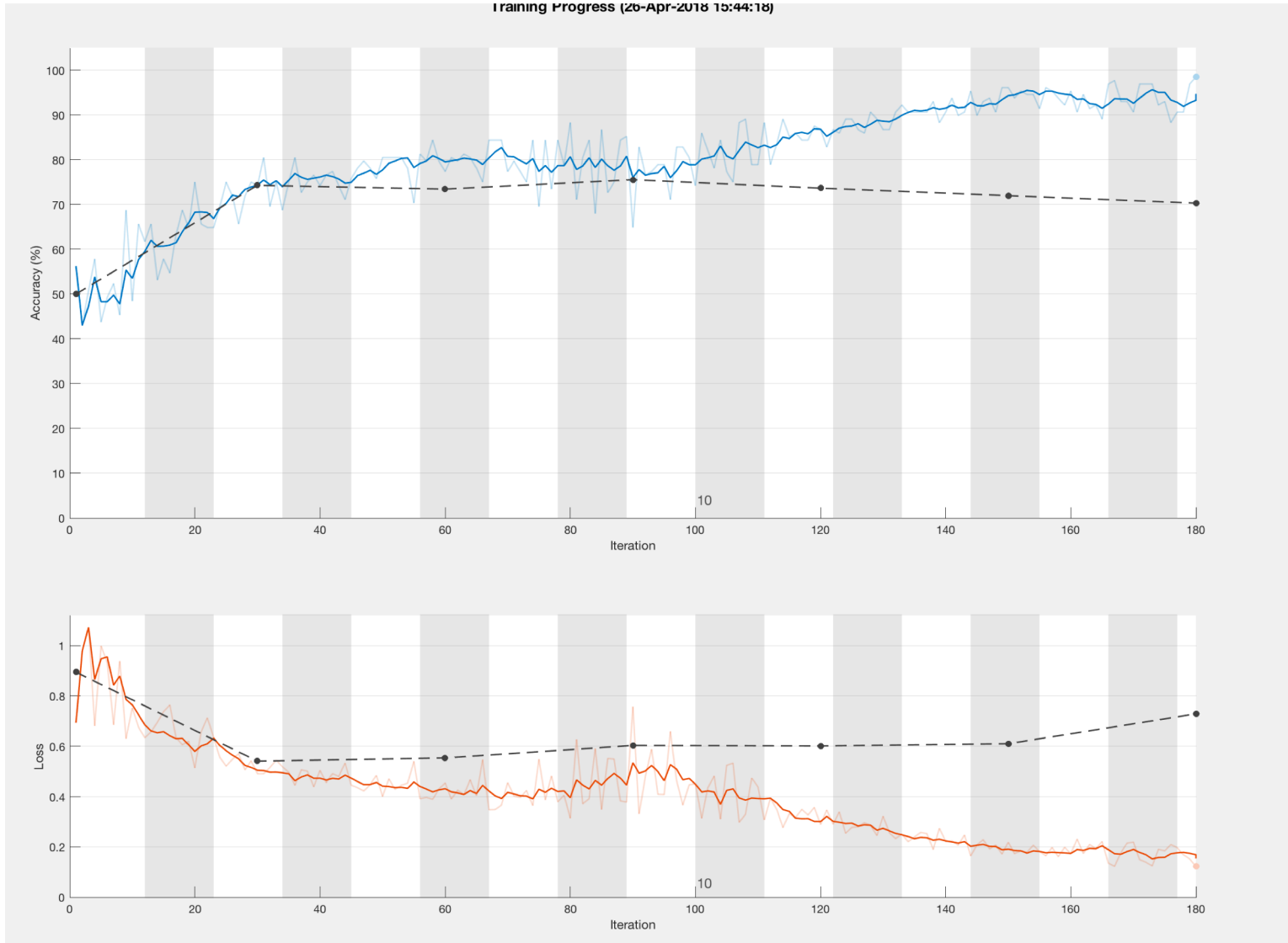
Other Information

Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.01

Learn more

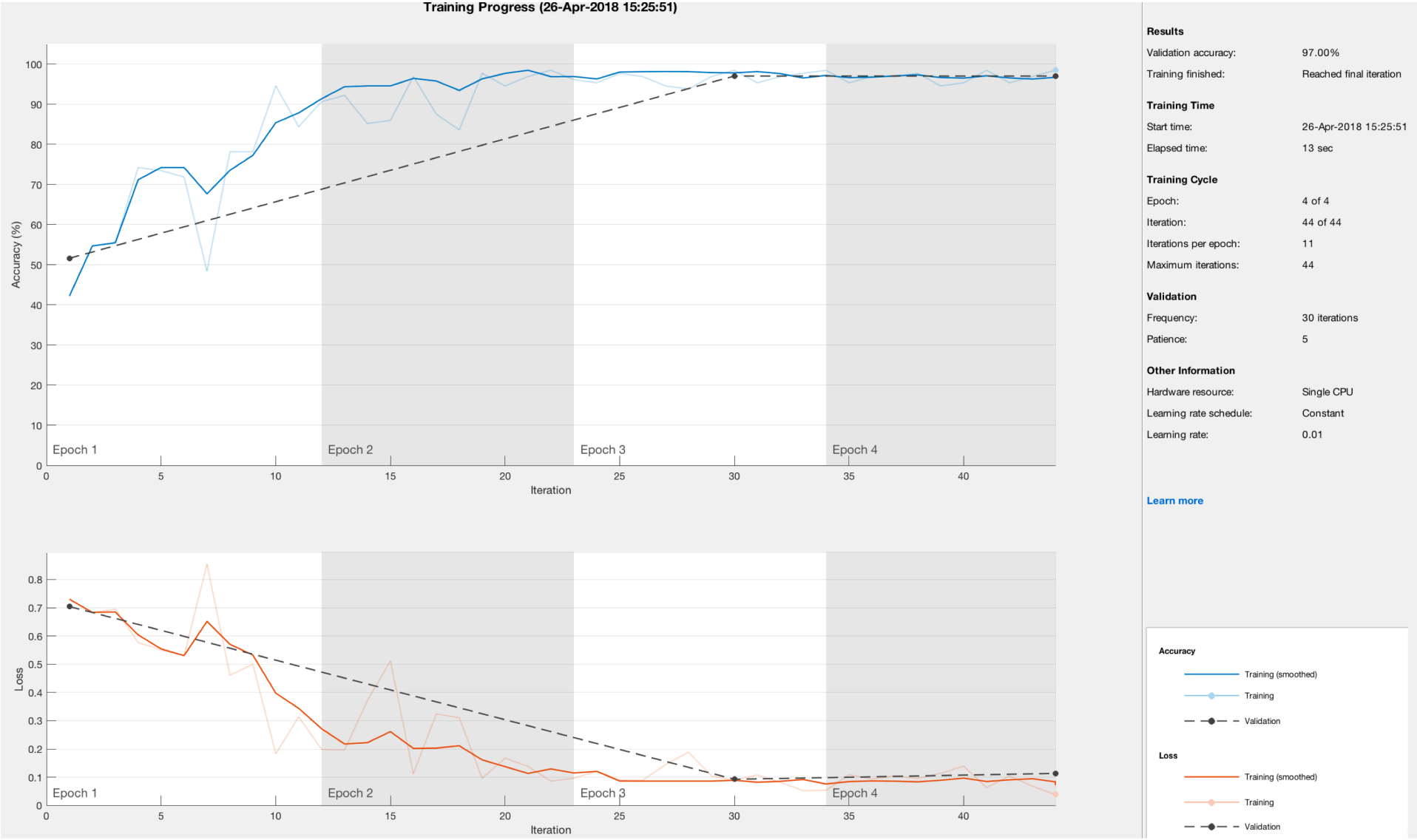
Interpretability for Detecting Adversarial Attacks

Relevancy Classifier – DeepFoolAttack – 70.29%



Interpretability for Detecting Adversarial Attacks

Input Classifier – GradientAttack – 97%



Interpretability for Detecting Adversarial Attacks

Relevancy Classifier – GradientAttack – 79.2%