Targeted Adviserial Example Generation

Trent Wall
ECE 6930
Fall 2019

Background:
Neural Networks are becoming more prevalent. As neural networks begin to be used in more safety critical systems it is important to be able to identify the images that can trick the neural networks. One method for generating adviserial examples is the Fast Gradient Signed Method (FGSM). This method makes minor changes to the image to create an image that looks correct to a human, but a neural network will misclassify the image. In this project. I am simplifying the problem of generating targeted examples downto just the mnist dataset.

Abstract:
The purpose of this experiment was to come up with a method to generate targeted adviserial examples. An adversarial example is an image that to a human looks like one object, but to a neural network the image looks like a different class. A targeted adviserial example is an adversarial example that to a neural network looks like a specific predetermined class.

Process:
I decided to see if watermarking an image with a lighter version of the target class would make the neural network more likely to classify the image as the target class. My hypothesis was that adding the watermark to the image after the image has been processed by the FGSM algorithm will generate more targeted adversarial examples, than adding the watermark to the image before applying the FGSM algorithm.

The watermark was created by selecting a target image and scaling it by a constant between Zero and One. Then taking an image from the MNIST dataset and lowering the pixel intensities of the image by One minus the constant. Finally, I added the two images together.

Results:
The following table shows the results obtained by the FGSM algorithm without any watermarking applied. In this test the original image was an image of a Five.

The next graph shows the results obtained by passing an image of Five to the FGSM and then applying the watermark to the new image.

The final table shows the results obtained by applying the watermark to the value of a Five, and then the result was passed to the FGSM algorithm.

Results for FGSM:

Results FGSM

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.0	0	0	0	0	0	0	0	0	1000	0

Results when water mark applied after FGSM:

Results After FGSM Target: Zero

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	998	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	4	0	0	0	0	0	0	0	996	0
0.46	1000	0	0	0	0	0	0	0	0	0

Results After FGSM Target: One

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	996	0
0.16	0	0	0	0	0	0	0	0	999	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Two

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	33	0	0	0	0	0	967	0
0.36	0	0	939	0	0	0	0	0	61	0
0.41	0	0	1000	0	0	0	0	0	0	0

0.46	0	0	1000	0	0	0	0	0	0	0

Results After FGSM Target: Three

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	993	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Four

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Five

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	820	0
0.16	0	0	0	0	0	0	0	0	632	0
0.21	0	0	0	0	0	0	0	0	258	0
0.26	0	0	0	0	0	0	0	0	18	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	0	0
0.41	0	0	0	0	0	0	0	0	0	0
0.46	0	0	0	0	0	0	0	0	0	0

Results After FGSM Target: Six

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0

0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Seven

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Eight

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	996	0
0.16	0	0	0	0	0	0	0	0	999	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results After FGSM Target: Nine

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results when watermark added before FGSM:

Results Before FGSM Target: Zero

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	1000	0	0	0	0	0	0	0	0	0

Results Before FGSM Target: One

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Two

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	0	1000
0.16	0	0	0	0	0	0	0	0	0	1000
0.21	0	0	0	0	0	0	0	0	0	1000
0.26	0	0	0	0	0	0	0	0	0	1000
0.31	0	0	0	0	0	0	0	0	0	1000
0.36	0	0	0	0	0	0	0	0	0	1000
0.41	0	0	0	0	0	0	0	0	0	1000
0.46	0	0	1000	0	0	0	0	0	0	0

Results Before FGSM Target:

	Three Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Four

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	0	1000
0.21	0	0	0	0	0	0	0	0	30	970
0.26	0	0	0	0	0	0	0	0	998	2
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	15	0	0	0	985	0

Results Before FGSM Target: Five

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	620	380
0.26	0	0	0	0	22	0	0	0	863	115
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Six

	Adviserial examples generated								
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0

0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Seven

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Eight

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Results Before FGSM Target: Nines

	Adviserial examples generated									
Watermark intensity	0	1	2	3	4	5	6	7	8	9
0.11	0	0	0	0	0	0	0	0	1000	0
0.16	0	0	0	0	0	0	0	0	1000	0
0.21	0	0	0	0	0	0	0	0	1000	0
0.26	0	0	0	0	0	0	0	0	1000	0
0.31	0	0	0	0	0	0	0	0	1000	0
0.36	0	0	0	0	0	0	0	0	1000	0
0.41	0	0	0	0	0	0	0	0	1000	0
0.46	0	0	0	0	0	0	0	0	1000	0

Discussion of the verification :
This algorithm only generated a few targeted adviserial examples. Using the watermark after FGSM I was able to successfully target 0,2, and 8. Eight was the easiest value to target, but the FGSM algorithm alone will generate adversarial examples of class 8 so I do not think that 8 should be considered in this study. Other than Eight, Two was the easiest value to target using the watermark I was able to successfully generate a Two by only changing the values by 0.31. Using the watermark before applying FGSM generated targeted adviserial examples for the values Zero, Two, and Four. Overall applying the watermark before the FGSM performed worse than applying the watermark after using the FGSM algorithm.

One unique finding was that watermarking the image of a Five with the value Five. This yielded a different result than just applying FGSM to the value Five. This is probably because I am applying a different image of a Five so they do not line up exactly. This leads me to believe that using a different image for the watermark would yield different adviserail examples based on the subtle differences in the watermark images.

Another interesting case was watermarking the Five with a Two. When applying the watermark after the FGSM this generated a lot of Eights, and eventually some Twos. When I applied the watermark before the FGSM the algorithm generated a lot of Nines, and then some Twos. Most of the values for the watermark generated Eights until the watermark was dark enough to generate the targeted value. But, for some reason applying the watermark with the value of Two before the FGSM generated Nines instead of Eights. I would think that most numbers like Four and Seven should also generate a large portion of Nines. It is possible that the Five watermarked with a Four or Seven looked too much like a Nine so FGSM changed it to be classified as an Eight. Whereas Five watermarked with a Two did not look quite as much like Nine, so FGSM changed it to be classified as a Nine.

The FGSM alone takes a matter of seconds to run, and generates a file that is less than 1megabyte. After adding the watermark to the image the code takes about half an hour to run all the iterations. With the watermark the code generates a text file that is less than 1 megabyte.

I would like to improve this algorithm by switching out the watermark with an outline of the target number. I would hope that this would make the adviserial images less obvious to humans. It would be interesting to generate an “average” image for the watermark. This average might be based on the features of all the training images of the targeted value.

Conclusion:
The watermarking method was effective in generating targeted examples. The best results came from applying the watermark after running the FGSM algorithm. The algorithm successfully generated targeted examples for the values Zero, Two, and Four. The watermark was darker than expected, but overall the project worked as expected, and the watermarking was ineffective at converting a Five into many classes.

Citations:

Goodfellow, Ian J, et al. “Explaining and Harnessing Adversarial Examples.” ArXiv.org, 20 Mar. 2015, https://arxiv.org/abs/1412.6572.

“Formal-Verification-Research/DLV_intellifeatures.” GitHub, https://github.com/formal-verification-research/DLV_intellifeatures.