CS5260 Assignment 1 **Name: Tey Shiwei** **Matric Number: A0112101M/E0403440**

Reference and credits to: https://github.com/ej0cl6/pytorch-adversarial-examples

1. Problem definition

- To generate adversarial image that satisfy equation 2 (minimum L2 norm to original dataset) and equation 3 (Strongly classified as one of the digits.

2. Attacks used

- 3 basic approach used: FGSM, Basic Iterative Method (BIM) and DeepFool
  - Results: BIM has the best results compared to FGSM and DeepFool (bad)
  - Insights: FGSM moves 1 step in 1 direction, while BIM moves in smaller step and updating its direction in every iteration, hence BIM better
  - DeepFool (might not) doesn't work well as its intention is not to maximize L2

2. Attacks	S per image
FGSM	0.4-0.5
BIM	0.6-0.65
Deep Fool	0.056

3. Iteration and EPS study for BIM

- Various eps (0.1 to 100) and iteration (10 to 100) experiments were carried out
  - Results: Different number has different optimal eps and iterations. As a result, the best adv images were generated from different eps.
  - Approach: Run the adv image for all different eps, pick the images that:
    - Satisfy equation 2 and equation 3 (Most of the images are strongly classified, with softmax value > 0.9
    - Has a higher S value than previous images

3. Iter & eps study	S per image
eps = 10, iter =50	<0.62
eps = 30, iter =50	<0.62
eps = 50, iter =50	0.629
eps = 70, iter =50	0.632
eps = 80, iter =30	0.632
eps = 80, iter =50	0.635
eps = 80, iter =100	0.632
eps = 100, iter =50	0.63

Table of top 10 S values for images from each classes

	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5	Sample 6	Sample 7	Sample 8	Sample 9	Sample 10	**Best S image**
Class 0	0.640308	0.646847	0.639241	0.648771	0.652281	0.651243	0.643064	0.644627	0.64331	0.645361	***0.652281***
Class 1	0.627965	0.620932	0.625481	0.622646	0.632302	0.623298	0.619828	0.626931	0.622618	0.622374	***0.632302***
Class 2	0.636075	0.638385	0.641224	0.637146	0.643249	0.639293	0.6397	0.636436	0.638745	0.637335	***0.643249***
Class 3	0.643939	0.644158	0.642983	0.645433	0.649041	0.645471	0.643126	0.646198	0.643019	0.650307	***0.650307***
Class 4	0.636529	0.63734	0.638941	0.635762	0.636761	0.636464	0.638695	0.638841	0.636232	0.635425	***0.638941***
Class 5	0.641449	0.642012	0.641342	0.640071	0.642782	0.644155	0.645287	0.641202	0.640327	0.643314	***0.645287***
Class 6	0.635762	0.632281	0.637677	0.63059	0.631957	0.637671	0.630494	0.630775	0.634784	0.633988	***0.637677***
Class 7	0.65211	0.653548	0.650595	0.64991	0.649623	0.64828	0.648276	0.656827	0.648359	0.651638	***0.656827***
Class 8	0.652828	0.655322	0.654534	0.654474	0.657675	0.657072	0.653087	0.653966	0.652561	0.660495	***0.660495***
Class 9	0.644738	0.645903	0.64568	0.644817	0.648993	0.644792	0.650583	0.649157	0.647352	0.647907	***0.650583***

Green colour = high values

Sum of S value: 64.20

Highest S with best S images from each class: ***64.68***

4. Future improvements and insights obtained

- The adversarial images generated were not the best images
- Some preliminary studies have been done by perturbing one of the pixels of the images (img still valid)
  - Pixels at the sides are at its optimal value
  - Perturbing pixels near middle by 0.05-0.1 will increase the value of S per image by 0.0001 to 0.0004
  - Which means, the adv images can be further improve by perturbing the pixels to the furthest point near the boundary
- However, the adv images submitted were adv images from different source of images
- Examples of the generated adversarial images:

0
1
2
3
4
5
6
7
8
9