Adversarial Attack

For this assignment, I implemented adversarial attacks on various pre-trained classifiers using the Fast Gradient Sign Method (FGSM) attack.

Label Mapping of ImageNet and ImageNette source:

Table:1

	Label_Name	Label_in_ImageNet	Label_in_ImageNette
'n01440764'	'tench'	0	0
'n02102040'	'English springer'	217	1
'n02979186'	'cassette player'	482	2
'n03000684'	'chain saw'	491	3
'n03028079'	'church'	497	4
'n03394916'	'French horn'	566	5
'n03417042'	'garbage truck'	569	6
'n03425413'	'gas pump'	571	7
'n03445777'	'golf ball'	574	8
'n03888257'	'parachute'	701	9

Setting:

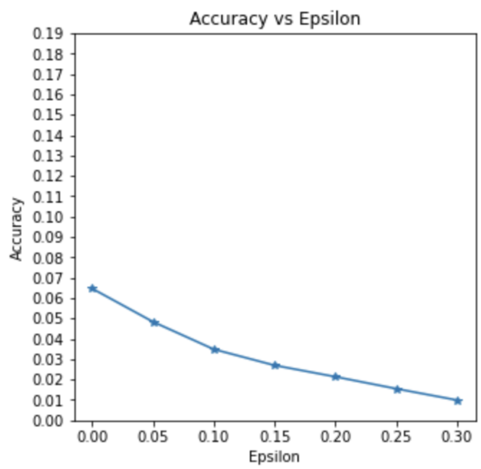
- **Pretrained Network:** Case-1: Alexnet, Case-2: ResNet, Case-3: Convnext_tiny
- **Attack:** Fast Gradient Sign Method (FGSM)
- **Accuracy vs Epsilon:**

From the accuracy versus epsilon plot(Fig:1,2,3), we can see that as epsilon increases the test accuracy decreases. This is because larger epsilons mean we take a larger step in the direction that will maximize the loss.

The trend in the curve is not linear even though the epsilon values are linearly spaced.

Case-1: AlexNet

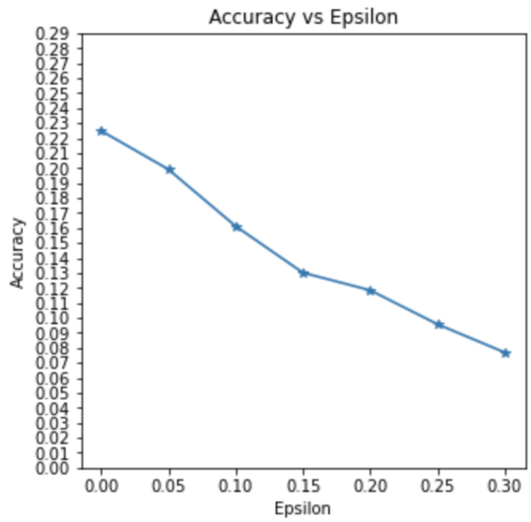
Fig:1




```
Epsilon: 0      Test Accuracy = 254 / 3925 = 0.06471337579617835
Epsilon: 0.05   Test Accuracy = 190 / 3925 = 0.048407643312101914
Epsilon: 0.1    Test Accuracy = 137 / 3925 = 0.03490445859872612
Epsilon: 0.15   Test Accuracy = 106 / 3925 = 0.02700636942675159
Epsilon: 0.2    Test Accuracy = 84 / 3925 = 0.02140127388535032
Epsilon: 0.25   Test Accuracy = 61 / 3925 = 0.01554140127388535
Epsilon: 0.3    Test Accuracy = 39 / 3925 = 0.009936305732484076
```

Case-2: ResNet

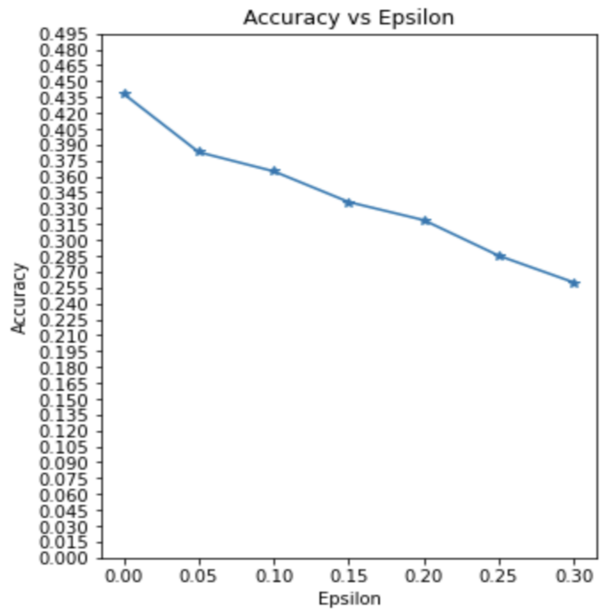
Fig:2




```
Epsilon: 0      Test Accuracy = 883 / 3925 = 0.22496815286624203
Epsilon: 0.05   Test Accuracy = 782 / 3925 = 0.19923566878980892
Epsilon: 0.1    Test Accuracy = 633 / 3925 = 0.16127388535031847
Epsilon: 0.15   Test Accuracy = 511 / 3925 = 0.13019108280254776
Epsilon: 0.2    Test Accuracy = 465 / 3925 = 0.11847133757961784
Epsilon: 0.25   Test Accuracy = 376 / 3925 = 0.09579617834394905
Epsilon: 0.3    Test Accuracy = 302 / 3925 = 0.07694267515923567
```

Case-3: Convnext_tiny

Fig: 3




```
Epsilon: 0      Test Accuracy = 1719 / 3925 = 0.43796178343949044
Epsilon: 0.05   Test Accuracy = 1503 / 3925 = 0.38292993630573247
Epsilon: 0.1    Test Accuracy = 1432 / 3925 = 0.3648407643312102
Epsilon: 0.15   Test Accuracy = 1318 / 3925 = 0.33579617834394904
Epsilon: 0.2    Test Accuracy = 1251 / 3925 = 0.31872611464968154
Epsilon: 0.25   Test Accuracy = 1119 / 3925 = 0.2850955414012739
Epsilon: 0.3    Test Accuracy = 1020 / 3925 = 0.25987261146496815
```

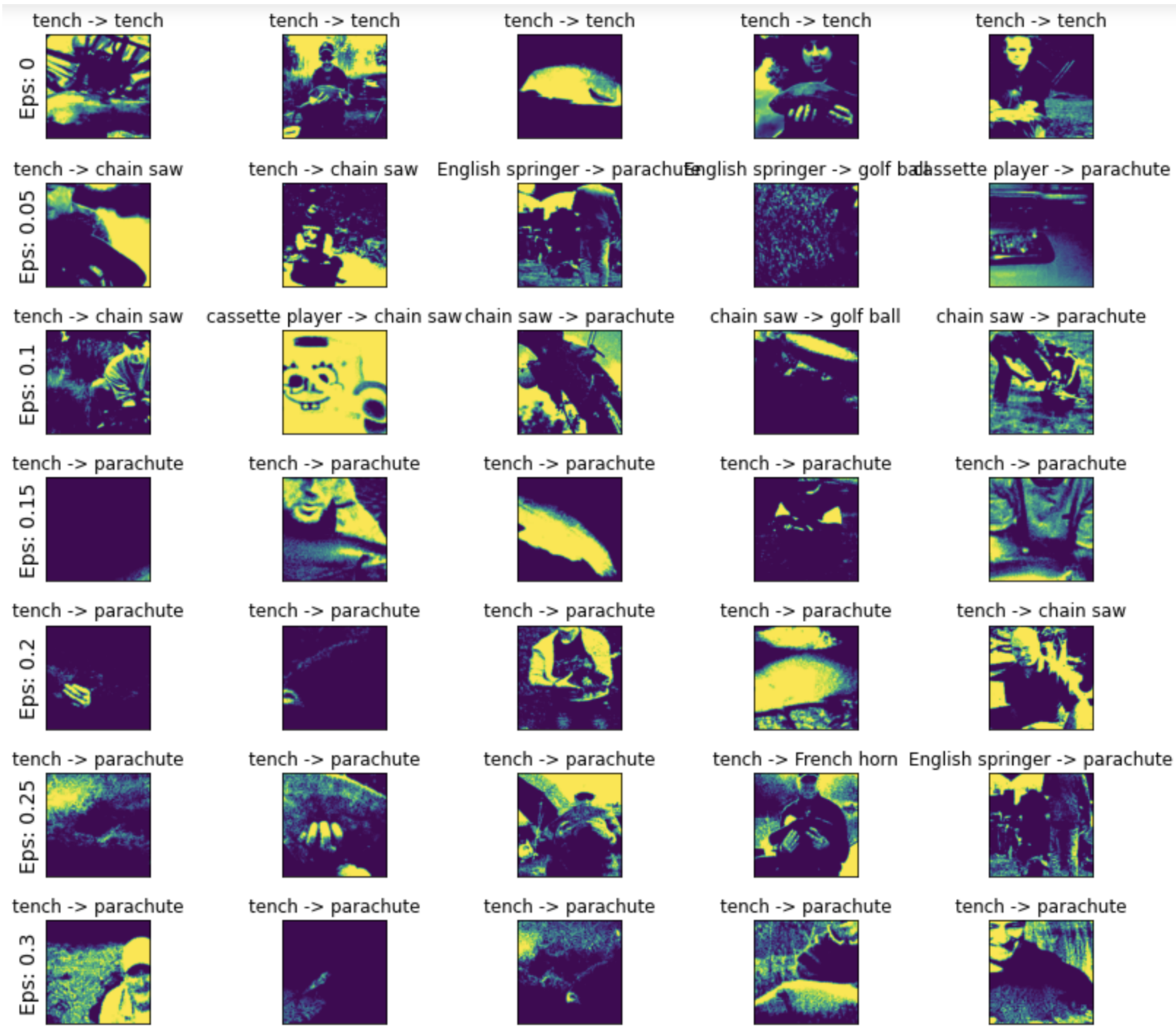
- **Sample Adversarial Examples:**

In this case, as epsilon increases the test accuracy decreases BUT the perturbations become more easily perceptible. In reality, there is a tradeoff between accuracy degradation and perceptibility that an attacker must consider. Here, we show some examples of successful adversarial examples at each epsilon value. Each row of the plot shows a different epsilon value. The title of each image shows the “original classification -> adversarial classification.”

FGSM attack seems a strong attack as it is able to manipulate the classifier to change its classification and thereby reducing the model’s classification accuracy.

Notice, the perturbations start to become evident at \epsilon=0.15ϵ=0.15 and are quite evident at \epsilon=0.3ϵ=0.3. However, in all cases humans are still capable of identifying the correct class despite the added noise.

Coloured outputs



Grayscale outputs

Unmapped outputs