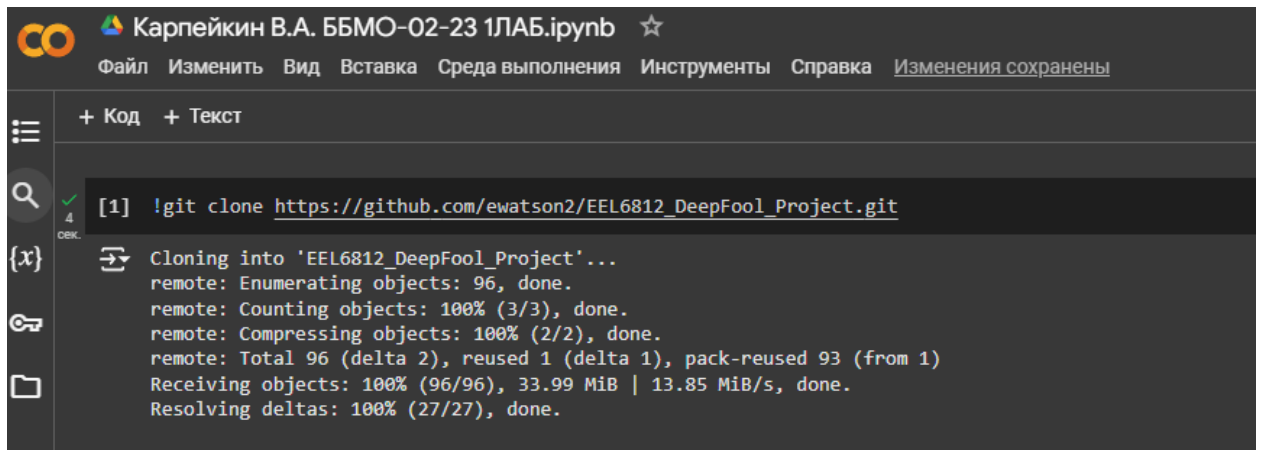


# Карпейкин В.А. ББМО-02-23 Номер 10

## Клонирование репозитория

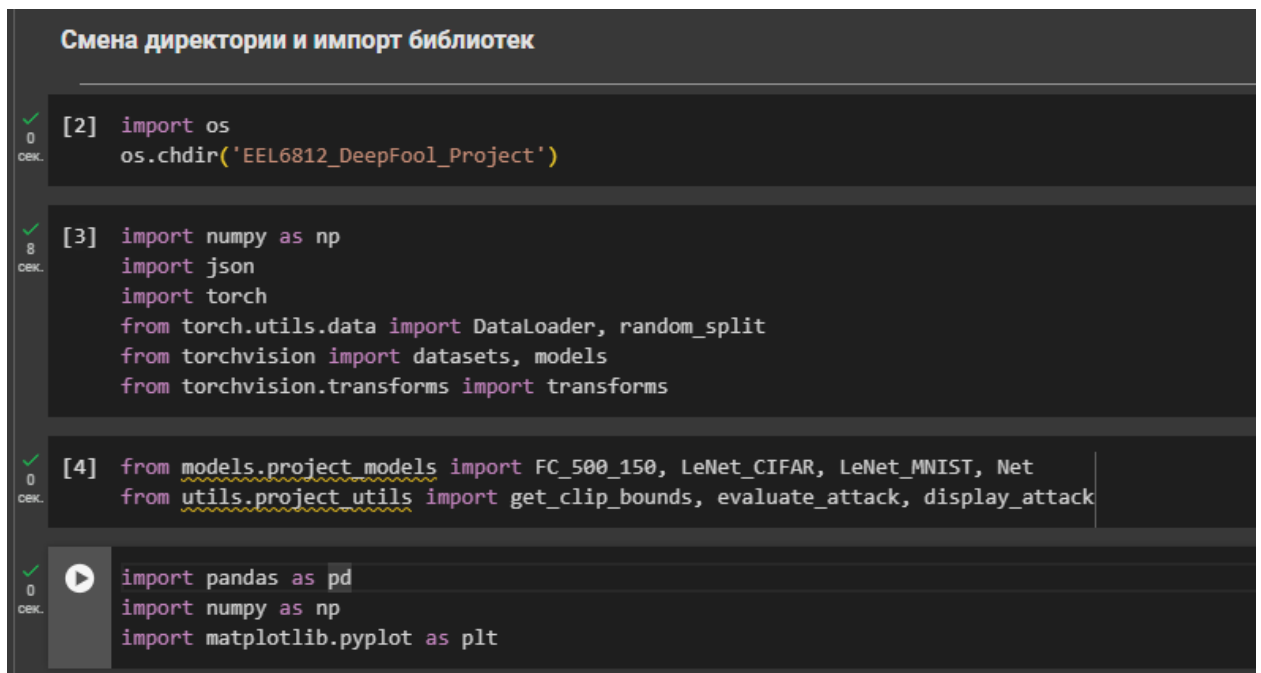


The screenshot shows a Jupyter Notebook titled "Карпейкин В.А. ББМО-02-23 1ЛАБ.ipynb". The interface includes a top bar with icons for file, edit, view, paste, environment, instruments, help, and saved changes. Below the top bar, there are tabs for "+ Код" and "+ Текст". The notebook content shows a single cell with the following code and output:

```
[1] !git clone https://github.com/ewatson2/EEL6812_DeepFool_Project.git
```

Cloning into 'EEL6812\_DeepFool\_Project'...  
remote: Enumerating objects: 96, done.  
remote: Counting objects: 100% (3/3), done.  
remote: Compressing objects: 100% (2/2), done.  
remote: Total 96 (delta 2), reused 1 (delta 1), pack-reused 93 (from 1)  
Receiving objects: 100% (96/96), 33.99 MiB | 13.85 MiB/s, done.  
Resolving deltas: 100% (27/27), done.

## Смена директории и импорт библиотек



The screenshot shows a Jupyter Notebook titled "Смена директории и импорт библиотек". The notebook content shows four cells with the following code:

```
[2] import os
os.chdir('EEL6812_DeepFool_Project')
```

```
[3] import numpy as np
import json
import torch
from torch.utils.data import DataLoader, random_split
from torchvision import datasets, models
from torchvision.transforms import transforms
```

```
[4] from models.project_models import FC_500_150, LeNet_CIFAR, LeNet_MNIST, Net
from utils.project_utils import get_clip_bounds, evaluate_attack, display_attack
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

## Установка случайного значения – номер в списке группы «10»

```
Установка случайного значения - номер в списке группы "10"
```

```
[7] rand_seed = 10
```

```
[8] np.random.seed(rand_seed)
    torch.manual_seed(rand_seed)

    use_cuda = torch.cuda.is_available()
    device = torch.device('cuda' if use_cuda else 'cpu')
```

## Загрузка датасета MNIST

```
Загрузка датасета MNIST
```

```
[9] mnist_mean = 0.5
    mnist_std = 0.5
    mnist_dim = 28
    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    mnist_min, mnist_max = get_clip_bounds(mnist_mean, mnist_std, mnist_dim)
    mnist_min = mnist_min.to(device)
    mnist_max = mnist_max.to(device)
    mnist_tf_train = transforms.Compose([
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        transforms.Normalize(mean=mnist_mean, std=mnist_std)
    ])
    mnist_tf = transforms.Compose([
        transforms.ToTensor(),
        transforms.Normalize(mean=mnist_mean, std=mnist_std)
    ])

    mnist_tf_inv = transforms.Compose([
        transforms.Normalize(
            mean=0.0,
            std=np.divide(1.0, mnist_std)),
        transforms.Normalize(
            mean=np.multiply(-1.0, mnist_std),
            std=1.0)])

    mnist_temp = datasets.MNIST(root='datasets/mnist', train=True, download=True, transform=mnist_tf_train)
    mnist_train, mnist_val = random_split(mnist_temp, [50000, 10000])
    mnist_test = datasets.MNIST(root='datasets/mnist', train=False, download=True, transform=mnist_tf)
```

Downloading <http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz>  
Failed to download (trying next):  
HTTP Error 403: Forbidden

Downloading <https://ossci-datasets.s3.amazonaws.com/mnist/train-images-idx3-ubyte.gz>  
Downloading <https://ossci-datasets.s3.amazonaws.com/mnist/train-images-idx3-ubyte.gz> to datasets/mnist/MNIST/raw/train-images-idx3-ubyte.gz  
100%|██████████| 9.91M/9.91M [00:01<00:00, 5.11MB/s]  
Extracting datasets/mnist/MNIST/raw/train-images-idx3-ubyte.gz to datasets/mnist/MNIST/raw

## Загрузка датасета CIFAR-10

## Загрузка датасета CIFAR-10

```
17 [1] cifar_mean = [0.491, 0.482, 0.447]
    cifar_std = [0.202, 0.199, 0.201]
    cifar_dim = 32
    cifar_min, cifar_max = get_clip_bounds(cifar_mean, cifar_std, cifar_dim)
    cifar_min = cifar_min.to(device)
    cifar_max = cifar_max.to(device)
    cifar_tf_train = transforms.Compose([
        transforms.RandomCrop(size=cifar_dim, padding=4),
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        transforms.Normalize(mean=cifar_mean, std=cifar_std)
    ])
    cifar_tf = transforms.Compose([
        transforms.ToTensor(),
        transforms.Normalize(mean=cifar_mean, std=cifar_std)
    ])

    cifar_tf_inv = transforms.Compose([
        transforms.Normalize(
            mean=[0.0, 0.0, 0.0],
            std=np.divide(1.0, cifar_std)),
        transforms.Normalize(
            mean=np.multiply(-1.0, cifar_mean),
            std=[1.0, 1.0, 1.0])]

    cifar_temp = datasets.CIFAR10(root='datasets/cifar-10', train=True, download=True, transform=cifar_tf_train)
    cifar_train, cifar_val = random_split(cifar_temp, [40000, 10000])
    cifar_test = datasets.CIFAR10(root='datasets/cifar-10', train=False, download=True, transform=cifar_tf)
    cifar_classes = ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck']

    Downloading https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz to datasets/cifar-10/cifar-10-python.tar.gz
    100%|██████████| 170M/170M [00:12<00:00, 13.1MB/s]
    Extracting datasets/cifar-10/cifar-10-python.tar.gz to datasets/cifar-10
    Files already downloaded and verified
```

## Настройка DataLoader

```
Настройка DataLoader

[11] batch_size = 64
    workers = 4
    mnist_loader_train = DataLoader(mnist_train, batch_size=batch_size, shuffle=True, num_workers=workers)
    mnist_loader_val = DataLoader(mnist_val, batch_size=batch_size, shuffle=False, num_workers=workers)
    mnist_loader_test = DataLoader(mnist_test, batch_size=batch_size, shuffle=False, num_workers=workers)
    cifar_loader_train = DataLoader(cifar_train, batch_size=batch_size, shuffle=True, num_workers=workers)
    cifar_loader_val = DataLoader(cifar_val, batch_size=batch_size, shuffle=False, num_workers=workers)
    cifar_loader_test = DataLoader(cifar_test, batch_size=batch_size, shuffle=False, num_workers=workers)

/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:617: UserWarning: This DataLoader will create 4 worker processes in total. Our suggested max number of worker in current system is 2, which is smaller than what ti
warnings.warn()
```

## FGSM атака

Стойкость к атаке моделей LeNet, FC на датасете MNIST и стойкость к атаке моделей Network-In-Network, LeNet на датасете CIFAR-10

# LeNet MNIST

$\text{fgsm\_eps} = 0.001$

```
<ipython-input-27-a421ced483ef>:2: FutureWarning: You are using
  model.load_state_dict(torch.load('weights/clean/mnist_lenet.pt
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
Точность до атаки: 98.34%
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 1.69%
FGSM Robustness : 8.06e-04
FGSM Time (All Images) : 1.05 s
FGSM Time (Per Image) : 104.57 us
```

$\text{fgsm\_eps} = 0.02$

```
<ipython-input-29-a421ced483ef>:2: FutureWarning: You are using
  model.load_state_dict(torch.load('weights/clean/mnist_lenet.pt
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
Точность до атаки: 98.34%
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 2.56%
FGSM Robustness : 1.59e-02
FGSM Time (All Images) : 0.95 s
FGSM Time (Per Image) : 95.34 us
```

$\text{fgsm\_eps} = 0.5$

```
<ipython-input-31-a421ced483ef>:2: FutureWarning: You are using
  model.load_state_dict(torch.load('weights/clean/mnist_lenet.pt
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
Точность до атаки: 98.34%
/usr/local/lib/python3.10/dist-packages/torch/utils/data/datalo
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 82.92%
FGSM Robustness : 3.83e-01
FGSM Time (All Images) : 1.00 s
FGSM Time (Per Image) : 99.73 us
```

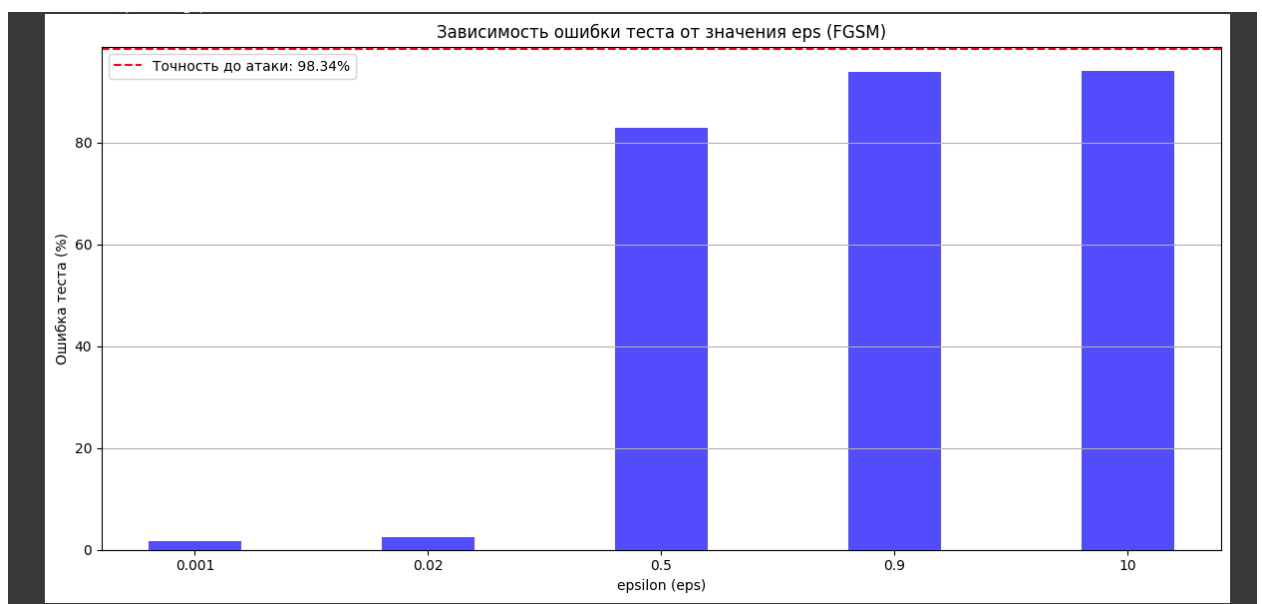
$\text{fgsm\_eps} = 0.9$

```
<ipython-input-33-a421ced483ef>:2: FutureWarning: You
  model.load_state_dict(torch.load('weights/clean/mnis
/usr/local/lib/python3.10/dist-packages/torch/utils/da
  warnings.warn(
Точность до атаки: 98.34%
/usr/local/lib/python3.10/dist-packages/torch/utils/da
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 93.80%
FGSM Robustness : 6.81e-01
FGSM Time (All Images) : 1.04 s
FGSM Time (Per Image) : 103.73 us
```

$\text{fgsm\_eps} = 10$

```
<ipython-input-35-a421ced483ef>:2: Future
  model.load_state_dict(torch.load('weig
/usr/local/lib/python3.10/dist-packages/t
  warnings.warn(
Точность до атаки: 98.34%
FGSM Test Error : 94.15%
FGSM Robustness : 1.46e+00
FGSM Time (All Images) : 1.41 s
FGSM Time (Per Image) : 141.28 us
```

## График



## FC MNIST

$\text{fgsm\_eps} = 0.001$

```
<ipython-input-37-86c4b3caf57a>:2: FutureWarning: You are using `torch.load`  
model.load_state_dict(torch.load('weights/  
/usr/local/lib/python3.10/dist-packages/torch/weights/weights.pth'))  
warnings.warn(  
Точность до атаки: 97.03%  
/usr/local/lib/python3.10/dist-packages/torch/weights/weights.pth'))  
warnings.warn(  
FGSM Batches Complete : (157 / 157)  
FGSM Test Error : 3.07%  
FGSM Robustness : 8.08e-04  
FGSM Time (All Images) : 0.67 s  
FGSM Time (Per Image) : 67.24 us
```

$\text{fgsm\_eps} = 0.02$

```
<ipython-input-39-86c4b3caf57a>:2: FutureWarning: You are using `torch.load`  
model.load_state_dict(torch.load('weights/clean/mnist_fc.pth'))  
/usr/local/lib/python3.10/dist-packages/torch/weights/weights.pth'))  
warnings.warn(  
Точность до атаки: 97.03%  
/usr/local/lib/python3.10/dist-packages/torch/weights/weights.pth'))  
warnings.warn(  
FGSM Batches Complete : (157 / 157)  
FGSM Test Error : 5.54%  
FGSM Robustness : 1.60e-02  
FGSM Time (All Images) : 0.64 s  
FGSM Time (Per Image) : 63.89 us
```

$\text{fgsm\_eps} = 0.5$

```
<ipython-input-41-86c4b3caf57a>:2: FutureWarning:
  model.load_state_dict(torch.load('weigh
/usr/local/lib/python3.10/dist-packages/t
  warnings.warn(
Точность до атаки: 97.03%
/usr/local/lib/python3.10/dist-packages/t
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 99.21%
FGSM Robustness : 3.86e-01
FGSM Time (All Images) : 0.63 s
FGSM Time (Per Image) : 63.05 us
```

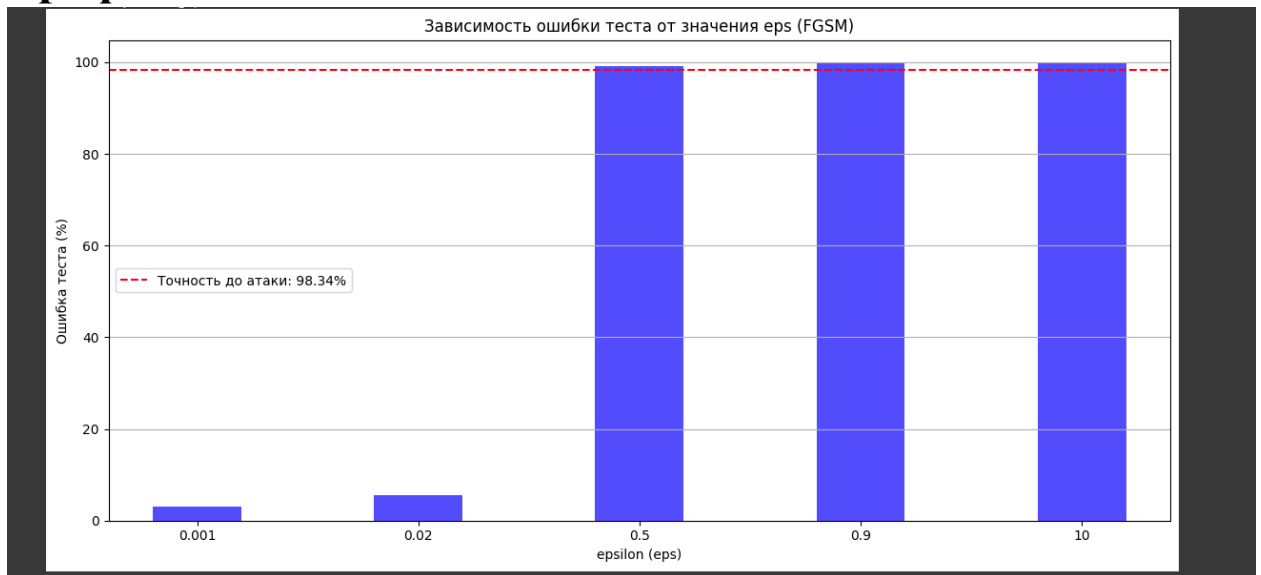
$\text{fgsm\_eps} = 0.9$

```
<ipython-input-43-86c4b3caf57a>:2: Futu
  model.load_state_dict(torch.load('wei
/usr/local/lib/python3.10/dist-packages
  warnings.warn(
Точность до атаки: 97.03%
/usr/local/lib/python3.10/dist-packages
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 99.87%
FGSM Robustness : 6.86e-01
FGSM Time (All Images) : 0.63 s
FGSM Time (Per Image) : 62.78 us
```

$\text{fgsm\_eps} = 10$

```
↻ <ipython-input-45-86c4b3caf57a>:2: Fut
  model.load_state_dict(torch.load('we
/usr/local/lib/python3.10/dist-package
  warnings.warn(
Точность до атаки: 97.03%
FGSM Test Error : 99.87%
FGSM Robustness : 1.47e+00
FGSM Time (All Images) : 0.85 s
FGSM Time (Per Image) : 84.99 us
```

## График



## Network-In-Network CIFAR-10

fgsm\_eps = 0.001

```
<ipython-input-47-412f7b94b9eb>:2: FutureWarning: You are u
  model.load_state_dict(torch.load('weights/clean/cifar_ni
/usr/local/lib/python3.10/dist-packages/torch/utils/data/d
  warnings.warn(
Точность до атаки: 90.72%
/usr/local/lib/python3.10/dist-packages/torch/utils/data/d
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 10.12%
FGSM Robustness : 8.92e-04
FGSM Time (All Images) : 1.29 s
FGSM Time (Per Image) : 129.21 us
```

fgsm\_eps = 0.02

```
<ipython-input-50-412f7b94b9eb>:2: FutureW
  model.load_state_dict(torch.load('weight
/usr/local/lib/python3.10/dist-packages/to
  warnings.warn(
Точность до атаки: 90.72%
/usr/local/lib/python3.10/dist-packages/to
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 30.76%
FGSM Robustness : 1.78e-02
FGSM Time (All Images) : 1.68 s
FGSM Time (Per Image) : 167.74 us
```



$\text{fgsm\_eps} = 0.5$

```
<ipython-input-53-412f7b94b9eb>:2: FutureWarning: You are using
model.load_state_dict(torch.load('weights/clean/cifar_
/usr/local/lib/python3.10/dist-packages/torch/utils/data
warnings.warn(
Точность до атаки: 90.72%
/usr/local/lib/python3.10/dist-packages/torch/utils/data
warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 82.67%
FGSM Robustness : 4.40e-01
FGSM Time (All Images) : 1.54 s
FGSM Time (Per Image) : 153.98 us
```

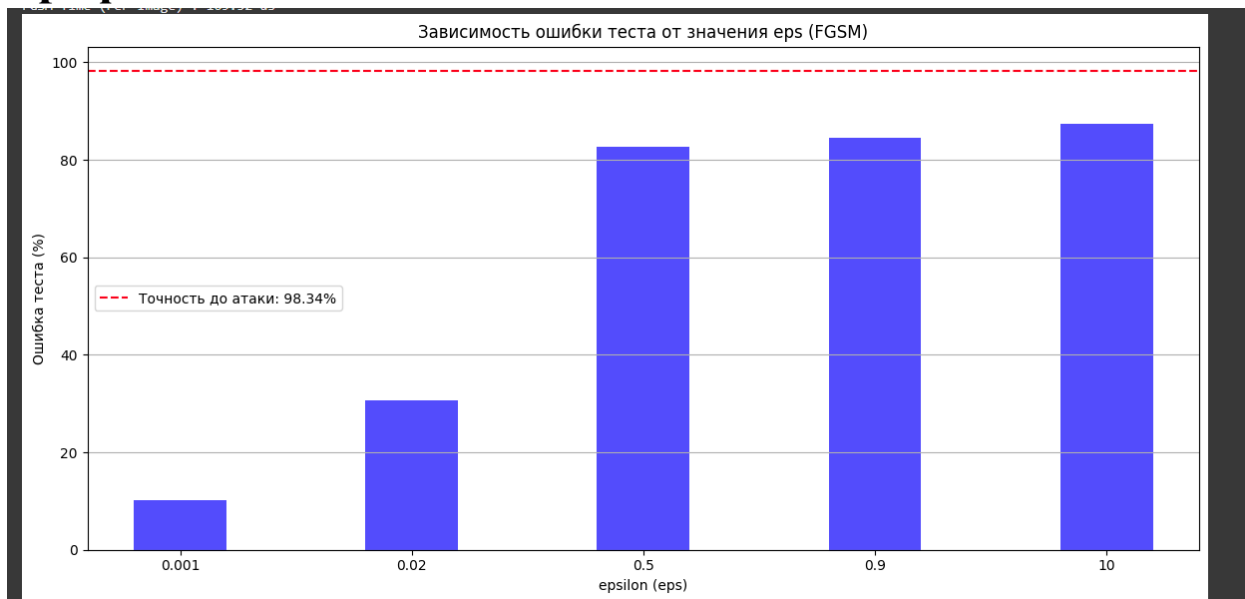
$\text{fgsm\_eps} = 0.9$

```
<ipython-input-56-412f7b94b9eb>:2: FutureWarning: You are using
model.load_state_dict(torch.load('weights/clean/cifar_
/usr/local/lib/python3.10/dist-packages/torch/utils/data
warnings.warn(
Точность до атаки: 90.72%
/usr/local/lib/python3.10/dist-packages/torch/utils/data
warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 84.62%
FGSM Robustness : 7.79e-01
FGSM Time (All Images) : 1.57 s
FGSM Time (Per Image) : 157.02 us
```

$\text{fgsm\_eps} = 10$

```
<ipython-input-59-412f7b94b9eb>:2: FutureWarning: model.load_state_dict(torch.load('weig
/usr/local/lib/python3.10/dist-packages/
warnings.warn(
Точность до атаки: 98.72%
FGSM Test Error : 87.50%
FGSM Robustness : 2.46e+00
FGSM Time (All Images) : 1.70 s
FGSM Time (Per Image) : 169.52 us
```

## График



## LeNet CIFAR-10

$\text{fgsm\_eps} = 0.001$

```
<ipython-input-48-fe2d187d7de7>:2: FutureWarning
  model.load_state_dict(torch.load('weights/clea
/usr/local/lib/python3.10/dist-packages/torch/ut
  warnings.warn(
Точность до атаки: 78.66%
/usr/local/lib/python3.10/dist-packages/torch/ut
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 22.72%
FGSM Robustness : 8.92e-04
FGSM Time (All Images) : 1.41 s
FGSM Time (Per Image) : 140.76 us
```

$\text{fgsm\_eps} = 0.02$

```
<ipython-input-51-fe2d187d7de7>:2: FutureWar
  model.load_state_dict(torch.load('weights/
/usr/local/lib/python3.10/dist-packages/torc
  warnings.warn(
Точность до атаки: 78.66%
/usr/local/lib/python3.10/dist-packages/torc
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 47.76%
FGSM Robustness : 1.78e-02
FGSM Time (All Images) : 1.29 s
FGSM Time (Per Image) : 128.99 us
```

$\text{fgsm\_eps} = 0.5$

```
<ipython-input-54-fe2d187d7de7>:2: FutureWa
  model.load_state_dict(torch.load('weights
/usr/local/lib/python3.10/dist-packages/tor
  warnings.warn(
Точность до атаки: 78.66%
/usr/local/lib/python3.10/dist-packages/tor
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 95.17%
FGSM Robustness : 4.40e-01
FGSM Time (All Images) : 1.26 s
FGSM Time (Per Image) : 125.84 us
```

$\text{fgsm\_eps} = 0.9$

```

<ipython-input-57-fe2d187d7de7>:2: FutureWarning:
  model.load_state_dict(torch.load('weights/clean/
/usr/local/lib/python3.10/dist-packages/torch/util
  warnings.warn(
Точность до атаки: 78.66%
/usr/local/lib/python3.10/dist-packages/torch/util
  warnings.warn(
FGSM Batches Complete : (157 / 157)
FGSM Test Error : 92.04%
FGSM Robustness : 7.80e-01
FGSM Time (All Images) : 1.33 s
FGSM Time (Per Image) : 133.41 us

```

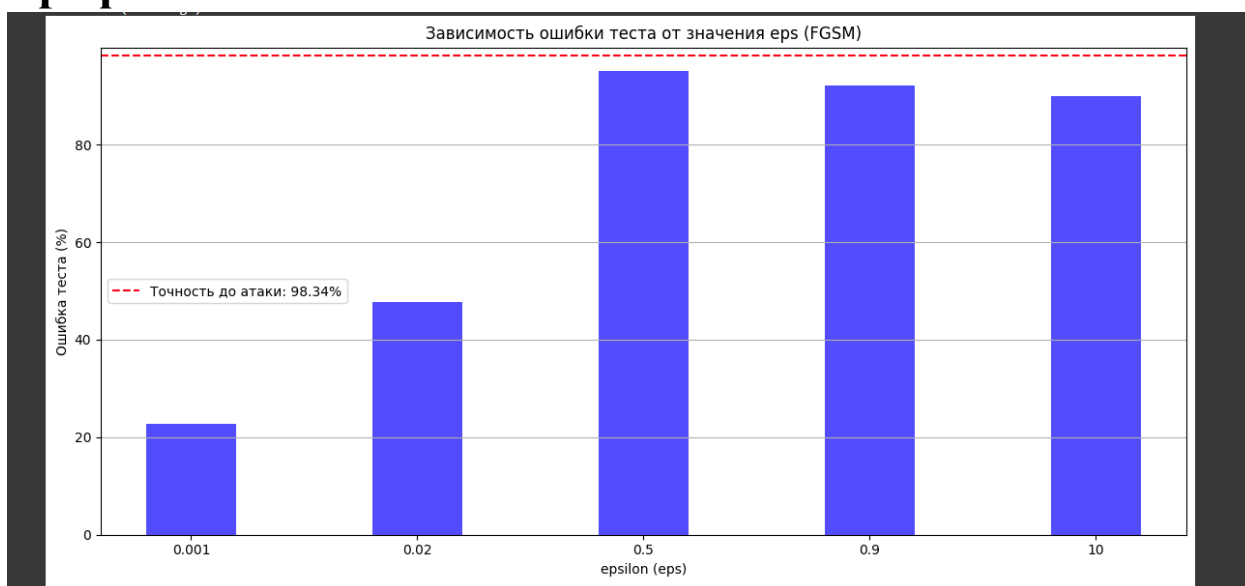
fgsm\_eps = 10

```

<ipython-input-60-fe2d187d7de7>:2: Futu
  model.load_state_dict(torch.load('wei
/usr/local/lib/python3.10/dist-packages
  warnings.warn(
Точность до атаки: 78.66%
FGSM Test Error : 89.90%
FGSM Robustness : 2.47e+00
FGSM Time (All Images) : 1.25 s
FGSM Time (Per Image) : 124.57 us

```

## График



# DeepFool атака

## Стойкость к атаке моделей LeNet, FC на датасете MNIST и стойкость к атаке моделей Network-In-Network, LeNet на датасете CIFAR-10

### LeNet MNIST

```
Стойкость к атакам модели LeNet на датасете MNIST

[18] model = LeNet_MNIST().to(device)
model.load_state_dict(torch.load('weights/clean/mnist_lenet.pth'))

evaluate_clean(model, mnist_loader_test, device)

evaluate_attack('mnist_lenet_deepfool.csv', 'results',
               device, model, mnist_loader_test,
               mnist_min, mnist_max, deep_args, is_fgsm=False)

if device.type == 'cuda':
    torch.cuda.empty_cache()

<ipython-input-18-9a4fabdb4dc1>:2: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default
model.load_state_dict(torch.load('weights/clean/mnist_lenet.pth'))
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:617: UserWarning: This DataLoader will create 4 worke
warnings.warn(
Точность до атаки: 98.34%
DeepFool Test Error : 98.74%
DeepFool Robustness : 9.64e-02
DeepFool Time (All Images) : 193.32 s
DeepFool Time (Per Image) : 19.33 ms
```

### FC MNIST

```
Стойкость к атакам модели FC на датасете MNIST

[19] model = FC_500_150().to(device)
model.load_state_dict(torch.load('weights/clean/mnist_fc.pth'))

evaluate_clean(model, mnist_loader_test, device)

evaluate_attack('mnist_fc_deepfool.csv', 'results',
               device, model, mnist_loader_test,
               mnist_min, mnist_max, deep_args, is_fgsm=False)

if device.type == 'cuda':
    torch.cuda.empty_cache()

<ipython-input-19-f4287413ae4e>:2: FutureWarning: You are using `torch.load` with `
model.load_state_dict(torch.load('weights/clean/mnist_fc.pth'))
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:617: UserWarning: This DataLoader will create 4 worke
warnings.warn(
Точность до атаки: 97.03%
DeepFool Test Error : 97.92%
DeepFool Robustness : 6.78e-02
DeepFool Time (All Images) : 141.81 s
DeepFool Time (Per Image) : 14.18 ms
```

## Network-In-Network CIFAR-10

```
Стойкость к атакам модели Network-In-Network на датасете CIFAR-10

3 сек.
model = Net().to(device)
model.load_state_dict(torch.load('weights/clean/cifar_nin.pth'))

evaluate_clean(model, cifar_loader_test, device)

evaluate_attack('cifar_nin_deepfool.csv', 'results',
               device, model, cifar_loader_test,
               cifar_min, cifar_max, deep_args, is_fgsm=False)

if device.type == 'cuda':
    torch.cuda.empty_cache()

<ipython-input-20-d39c82e071ac>:2: FutureWarning: You are using `torch.load` with `weights`
model.load_state_dict(torch.load('weights/clean/cifar_nin.pth'))
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:617: UserWarning: The
warnings.warn(
Точность до атаки: 90.72%
DeepFool Test Error : 93.76%
DeepFool Robustness : 2.12e-02
DeepFool Time (All Images) : 185.12 s
DeepFool Time (Per Image) : 18.51 ms
```

## LeNet CIFAR-10

```
Стойкость к атакам модели LeNet на датасете CIFAR-10

3 сек.
[21] model = LeNet_CIFAR().to(device)
model.load_state_dict(torch.load('weights/clean/cifar_lenet.pth'))

evaluate_clean(model, cifar_loader_test, device)

evaluate_attack('cifar_lenet_deepfool.csv', 'results',
               device, model, cifar_loader_test,
               cifar_min, cifar_max, deep_args, is_fgsm=False)

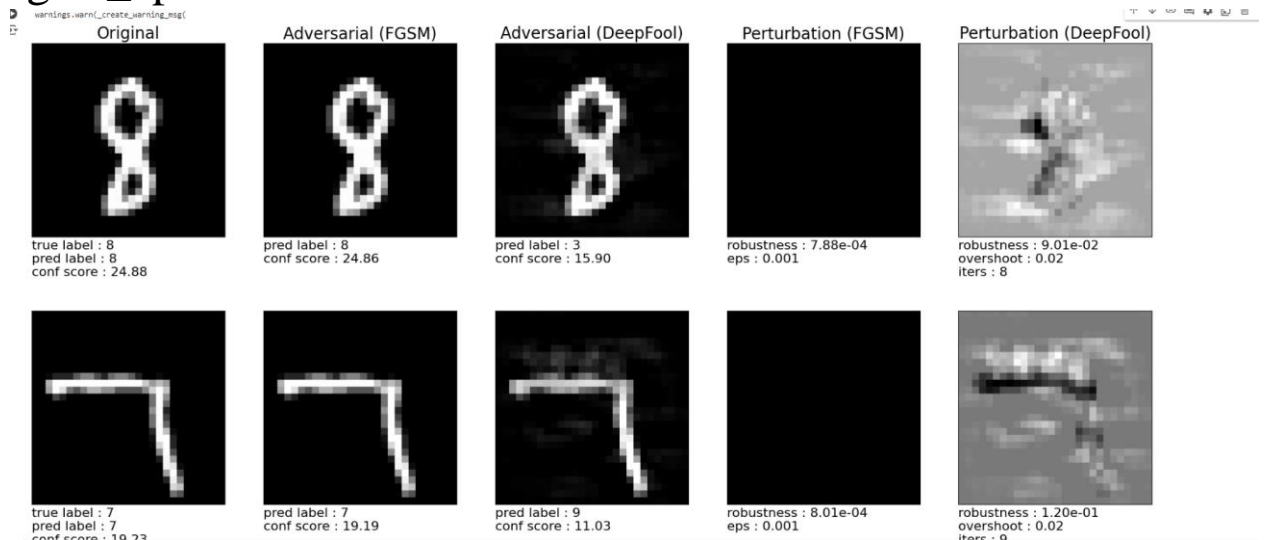
if device.type == 'cuda':
    torch.cuda.empty_cache()

<ipython-input-21-71a3964ca979>:2: FutureWarning: You are using `torch.load` with `weights`
model.load_state_dict(torch.load('weights/clean/cifar_lenet.pth'))
/usr/local/lib/python3.10/dist-packages/torch/utils/data/dataloader.py:617: UserWarning: The
warnings.warn(
Точность до атаки: 78.66%
DeepFool Test Error : 87.81%
DeepFool Robustness : 1.78e-02
DeepFool Time (All Images) : 73.27 s
DeepFool Time (Per Image) : 7.33 ms
```

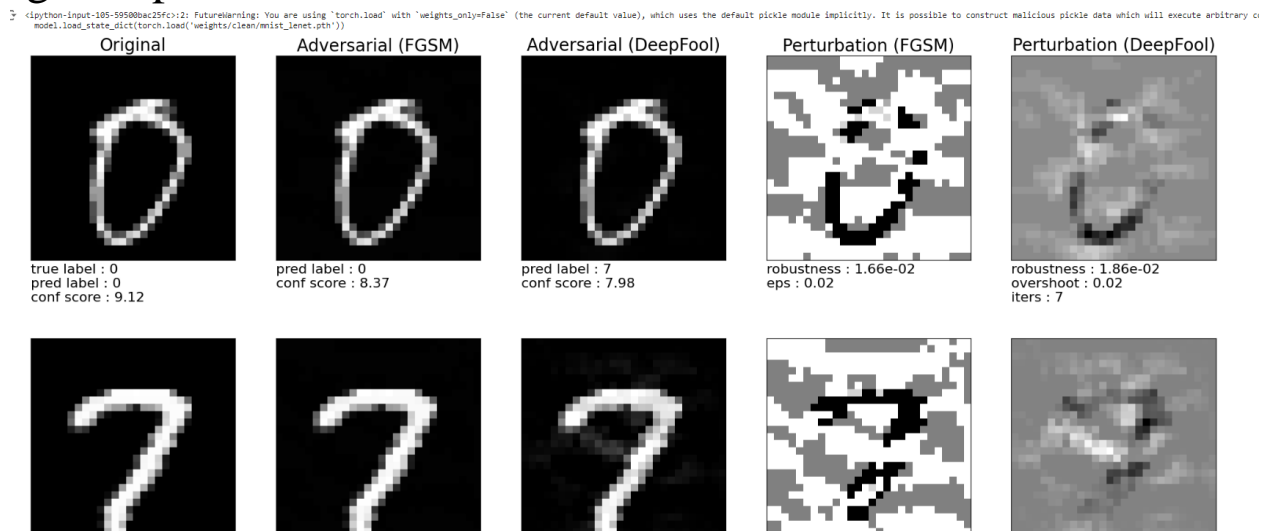
## Визуальное представление

## LeNet MNIST

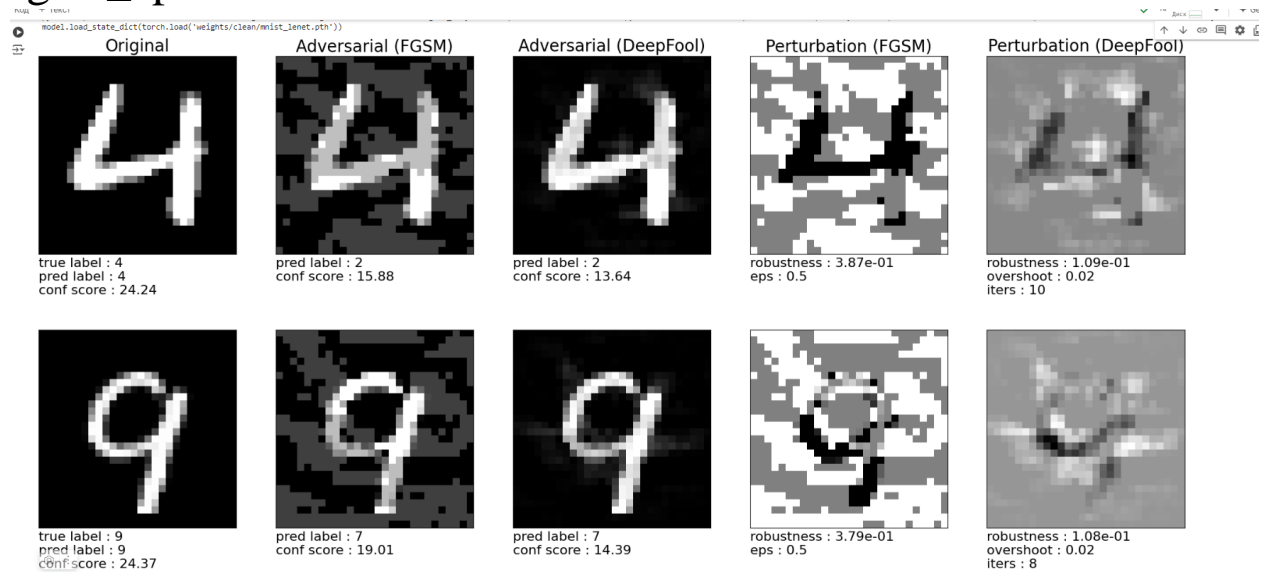
fgsm\_eps = 0.001



fgsm\_eps = 0.02



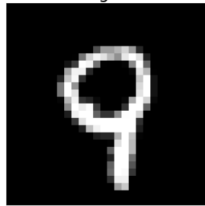
fgsm\_eps = 0.5



fgsm\_eps = 0.9

model.load\_state\_dict(torch.load('weights/clean/mnist\_lenet.pth'))

Original



true label : 9  
pred label : 9  
conf score : 22.86

Adversarial (FGSM)



pred label : 8  
conf score : 16.38

Adversarial (DeepFool)



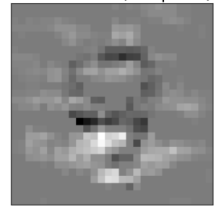
pred label : 8  
conf score : 15.82

Perturbation (FGSM)

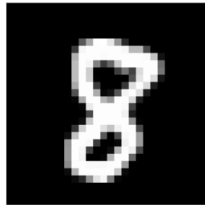


robustness : 6.72e-01  
eps : 0.9

Perturbation (DeepFool)



robustness : 5.36e-02  
overshoot : 0.02  
iters : 9



true label : 8  
pred label : 8  
conf score : 30.76



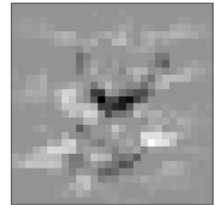
pred label : 3  
conf score : 22.04



pred label : 2  
conf score : 16.71



robustness : 6.49e-01  
eps : 0.9

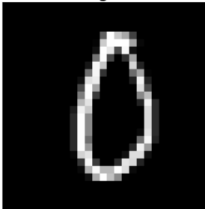


robustness : 1.34e-01  
overshoot : 0.02  
iters : 10

fgsm\_eps = 10

model.load\_state\_dict(torch.load('weights/clean/mnist\_lenet.pth'))

Original



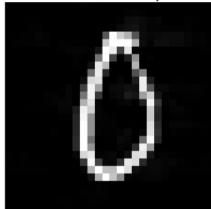
true label : 0  
pred label : 0  
conf score : 12.98

Adversarial (FGSM)



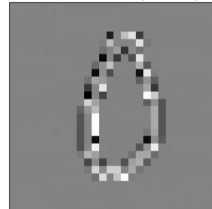
pred label : 5  
conf score : 26.00

Adversarial (DeepFool)



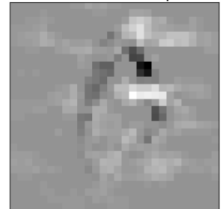
pred label : 6  
conf score : 10.35

Perturbation (FGSM)



robustness : 1.47e+00  
eps : 10

Perturbation (DeepFool)



robustness : 3.99e-02  
overshoot : 0.02  
iters : 7



true label : 3  
pred label : 3  
conf score : 22.86



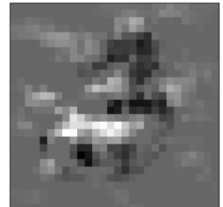
pred label : 2  
conf score : 23.12



pred label : 8  
conf score : 11.80



robustness : 1.42e+00  
eps : 10

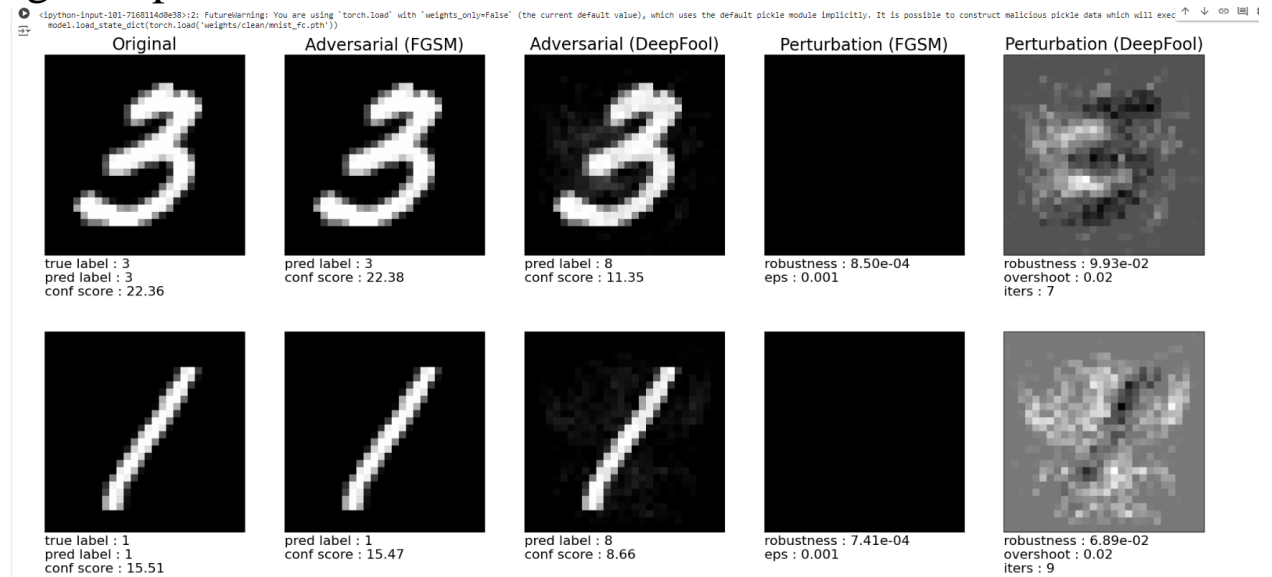


robustness : 1.60e-01  
overshoot : 0.02

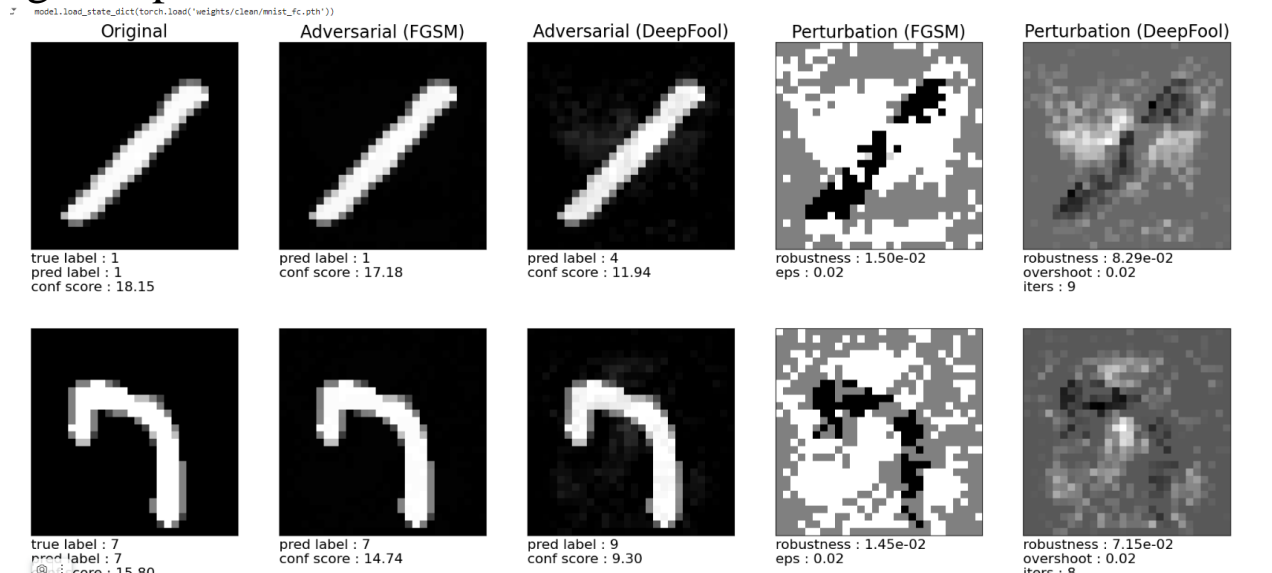
FC MNIST



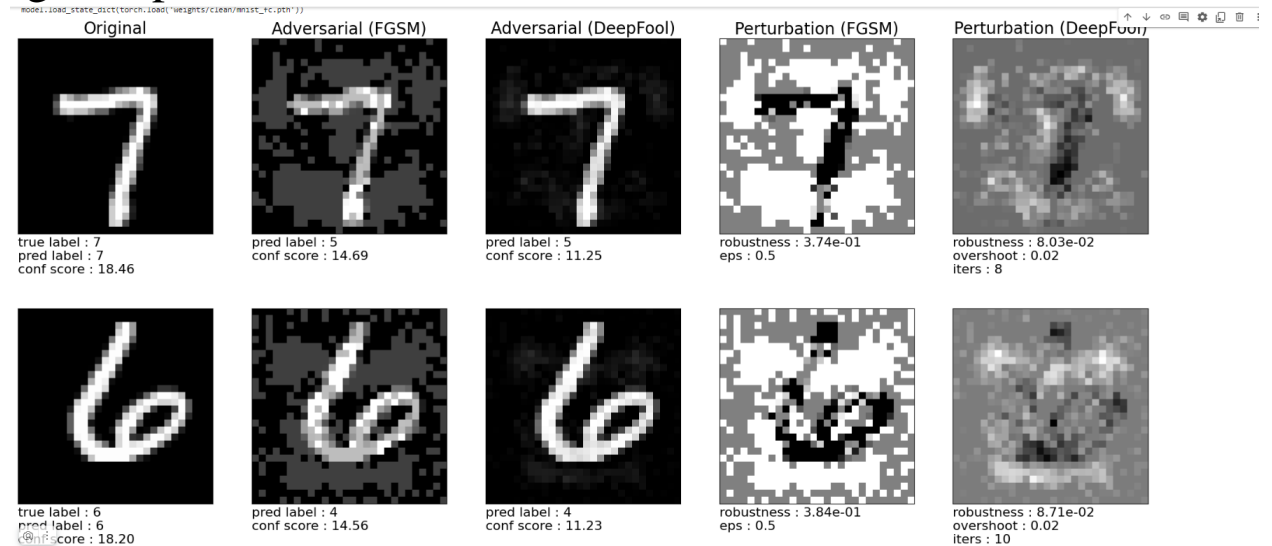
fgsm\_eps = 0.001



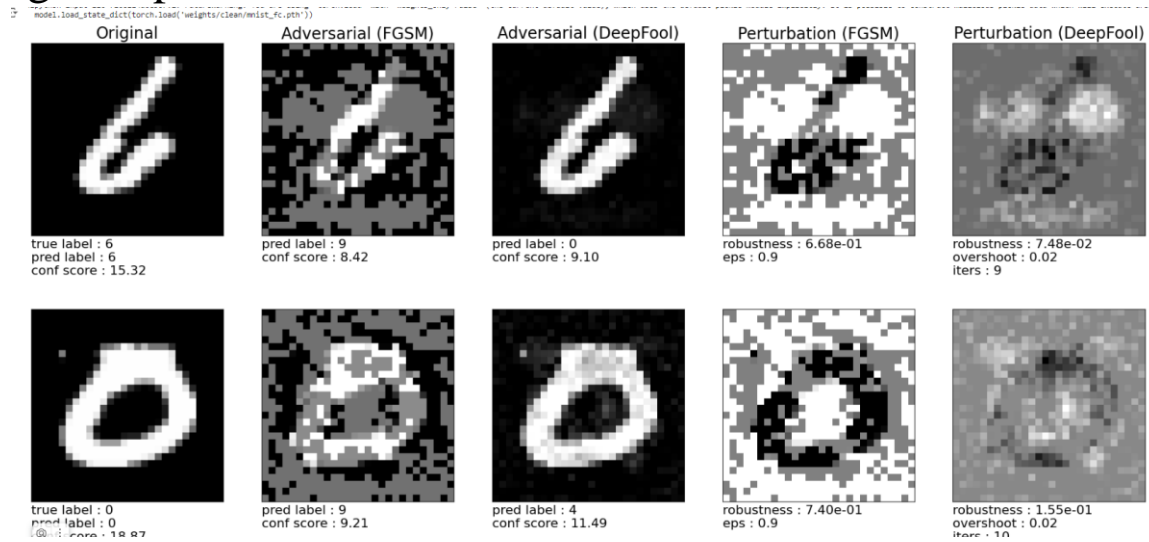
fgsm\_eps = 0.02



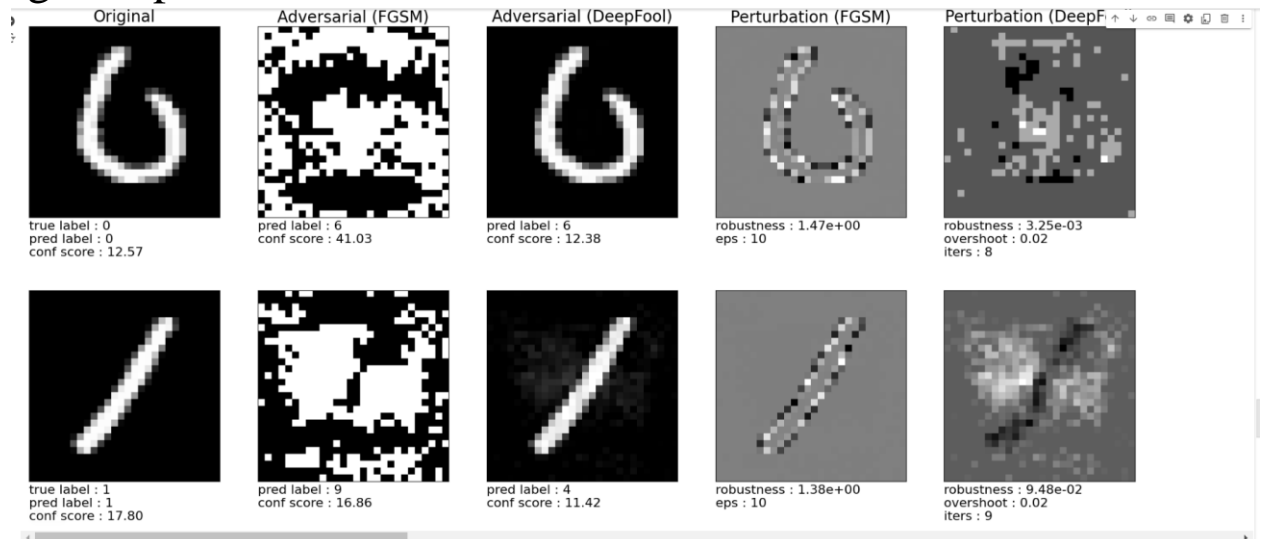
fgsm\_eps = 0.5



fgsm\_eps = 0.9



fgsm\_eps = 10



**Network-In-Network CIFAR-10**

fgsm\_eps = 0.001

`model.load_state_dict(torch.load('weights/clean/cifar_100.pth'))`

Original



true label : horse  
pred label : horse  
conf score : 32.37

Adversarial (FGSM)



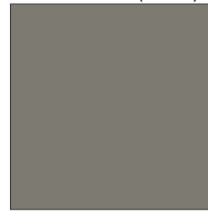
pred label : horse  
conf score : 32.11

Adversarial (DeepFool)



pred label : bird  
conf score : 21.72

Perturbation (FGSM)



robustness : 1.04e-03  
eps : 0.001

Perturbation (DeepFool)



robustness : 3.61e-02  
overshoot : 0.02  
iters : 3



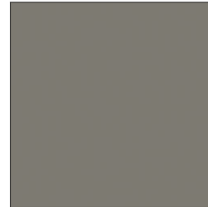
true label : truck  
pred label : truck  
conf score : 44.69



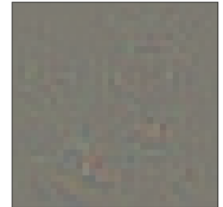
pred label : truck  
conf score : 44.96



pred label : ship  
conf score : 22.74



robustness : 9.35e-04  
eps : 0.001



robustness : 5.28e-02  
overshoot : 0.02  
iters : 4

fgsm\_eps = 0.02

Original



true label : horse  
pred label : horse  
conf score : 32.37

Adversarial (FGSM)



pred label : horse  
conf score : 32.11

Adversarial (DeepFool)



pred label : bird  
conf score : 21.72

Perturbation (FGSM)



robustness : 1.04e-03  
eps : 0.001

Perturbation (DeepFool)



robustness : 3.61e-02  
overshoot : 0.02  
iters : 3



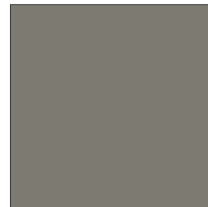
true label : truck  
pred label : truck  
conf score : 44.69



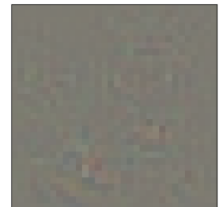
pred label : truck  
conf score : 44.96



pred label : ship  
conf score : 22.74




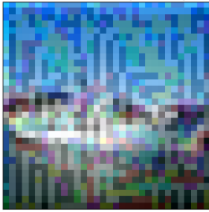

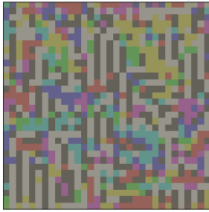

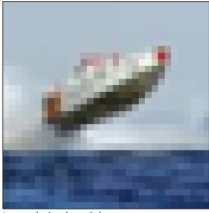
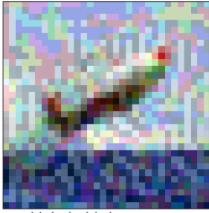

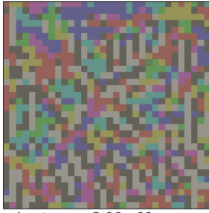
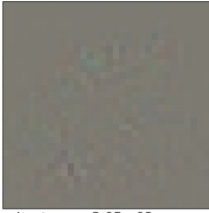
robustness : 9.35e-04  
eps : 0.001



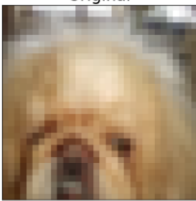
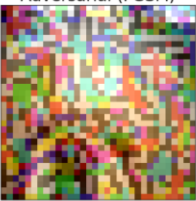
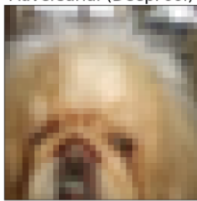
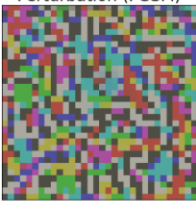

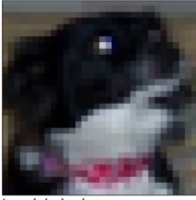
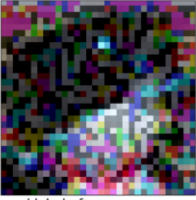
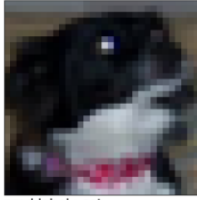
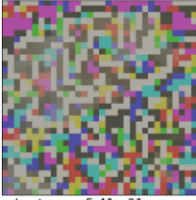

robustness : 5.28e-02  
overshoot : 0.02  
iters : 4

fgsm\_eps = 0.5





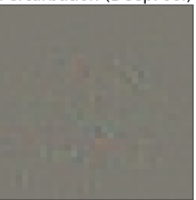




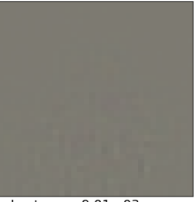
`<ipython-input-112-e97f89c1935b>12: FutureWarning: You are using 'torch.load' with 'weights_only=False' (the current default value), which uses the default pickle module implicitly. It is possible to construct malicious pickle data which will execute arbitrary code during unpickling. It is recommended to use 'torch.load(..., weights_only=True)' to disable this functionality.`

Original	Adversarial (FGSM)	Adversarial (DeepFool)	Perturbation (FGSM)	Perturbation (DeepFool)
				
true label : ship pred label : ship conf score : 35.45	pred label : ship conf score : 15.81	pred label : airplane conf score : 19.36	robustness : 4.27e-01 eps : 0.5	robustness : 4.66e-02 overshoot : 0.02 iters : 3
				
true label : ship pred label : ship conf score : 31.40	pred label : bird conf score : 16.82	pred label : airplane conf score : 23.94	robustness : 3.90e-01 eps : 0.5	robustness : 3.05e-02 overshoot : 0.02 iters : 3

fgsm\_eps = 0.9

Original	Adversarial (FGSM)	Adversarial (DeepFool)	Perturbation (FGSM)	Perturbation (DeepFool)
				
true label : dog pred label : dog conf score : 30.20	pred label : frog conf score : 20.11	pred label : horse conf score : 22.16	robustness : 9.05e-01 eps : 0.9	robustness : 2.10e-02 overshoot : 0.02 iters : 2
				
true label : dog pred label : dog conf score : 24.91	pred label : frog conf score : 22.97	pred label : cat conf score : 22.94	robustness : 5.41e-01 eps : 0.9	robustness : 6.13e-03 overshoot : 0.02 iters : 2




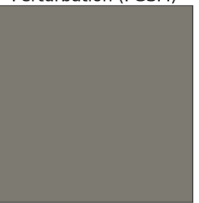
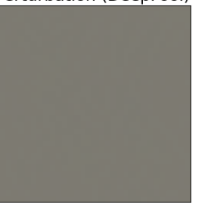



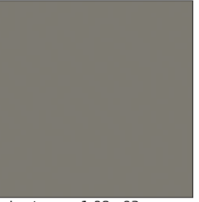
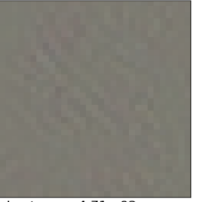
fgsm\_eps = 10

Original	Adversarial (FGSM)	Adversarial (DeepFool)	Perturbation (FGSM)	Perturbation (DeepFool)
				
true label : automobile pred label : automobile conf score : 59.58	pred label : automobile conf score : 24.03	pred label : truck conf score : 44.23	robustness : 2.54e+00 eps : 10	robustness : 3.57e-02 overshoot : 0.02 iters : 2
				
true label : ship pred label : airplane	pred label : frog conf score : 17.52	pred label : deer conf score : 13.40	robustness : 1.61e+00 eps : 10	robustness : 9.01e-03 overshoot : 0.02




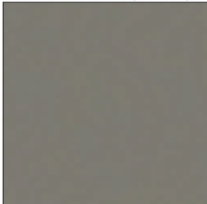




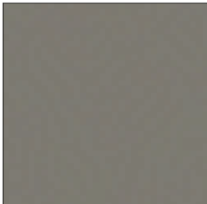

# LeNet CIFAR-10

fgsm\_eps = 0.001

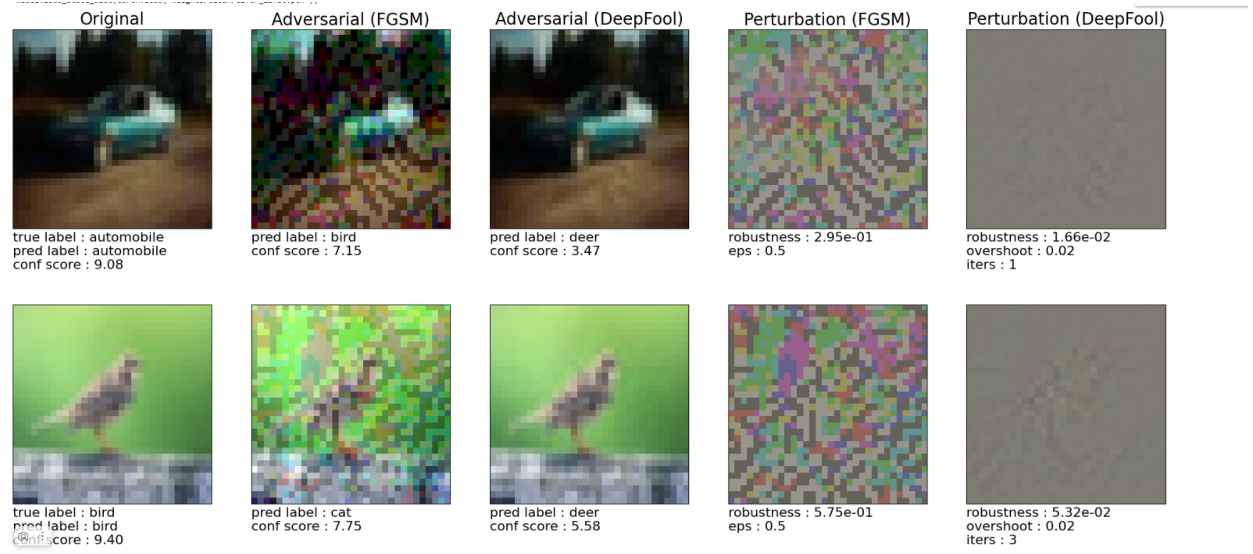
`> python-input-103-c252624b6c5>2: FutureWarning: You are using "torch.load" with "weights_only=False" (the current default value), which uses the default pickle module implicitly. It is possible to construct malicious pickle data which will execute arbitrary code during unpickling. It is recommended to use "torch.load(..., weights_only=True)" to avoid this warning and any potential security issues.`

Original	Adversarial (FGSM)	Adversarial (DeepFool)	Perturbation (FGSM)	Perturbation (DeepFool)
				
true label : bird pred label : dog conf score : 3.99	pred label : dog conf score : 4.03	pred label : bird conf score : 3.66	robustness : 1.06e-03 eps : 0.001	robustness : 4.13e-03 overshoot : 0.02 iters : 2
				
true label : frog pred label : frog conf score : 10.45	pred label : frog conf score : 10.33	pred label : bird conf score : 5.94	robustness : 1.08e-03 eps : 0.001	robustness : 4.71e-02 overshoot : 0.02 iters : 2

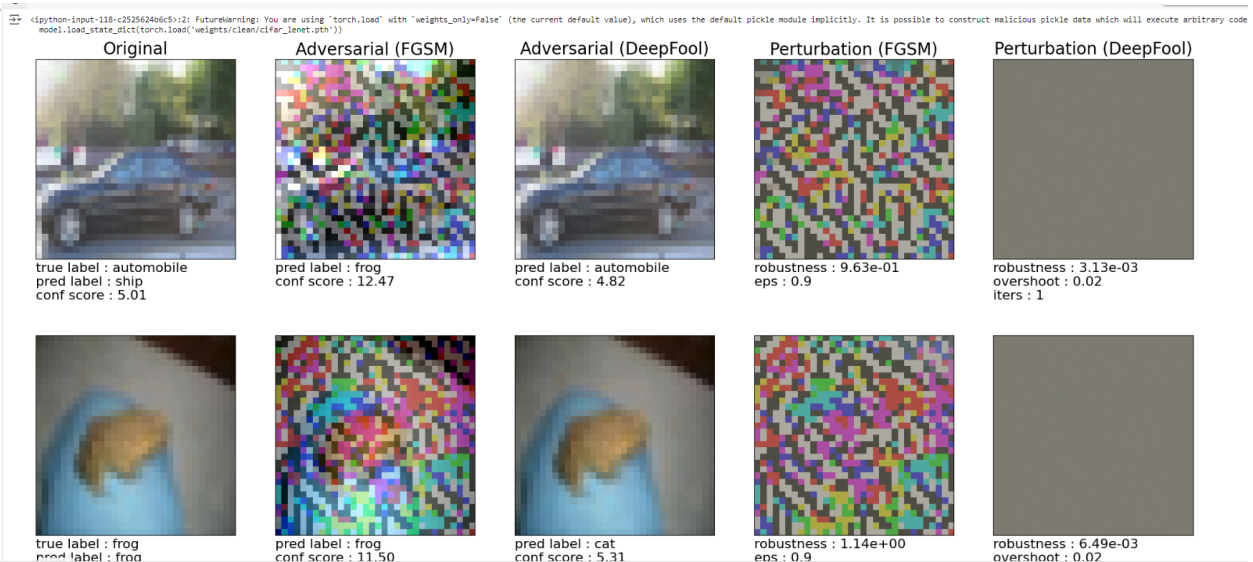
fgsm\_eps = 0.02

Original	Adversarial (FGSM)	Adversarial (DeepFool)	Perturbation (FGSM)	Perturbation (DeepFool)
				
true label : bird pred label : bird conf score : 5.31	pred label : cat conf score : 4.73	pred label : cat conf score : 4.42	robustness : 3.31e-02 eps : 0.02	robustness : 7.97e-03 overshoot : 0.02 iters : 2
				
true label : frog pred label : frog conf score : 14.51	pred label : frog conf score : 11.21	pred label : horse conf score : 6.08	robustness : 1.35e-02 eps : 0.02	robustness : 2.78e-02 overshoot : 0.02 iters : 4

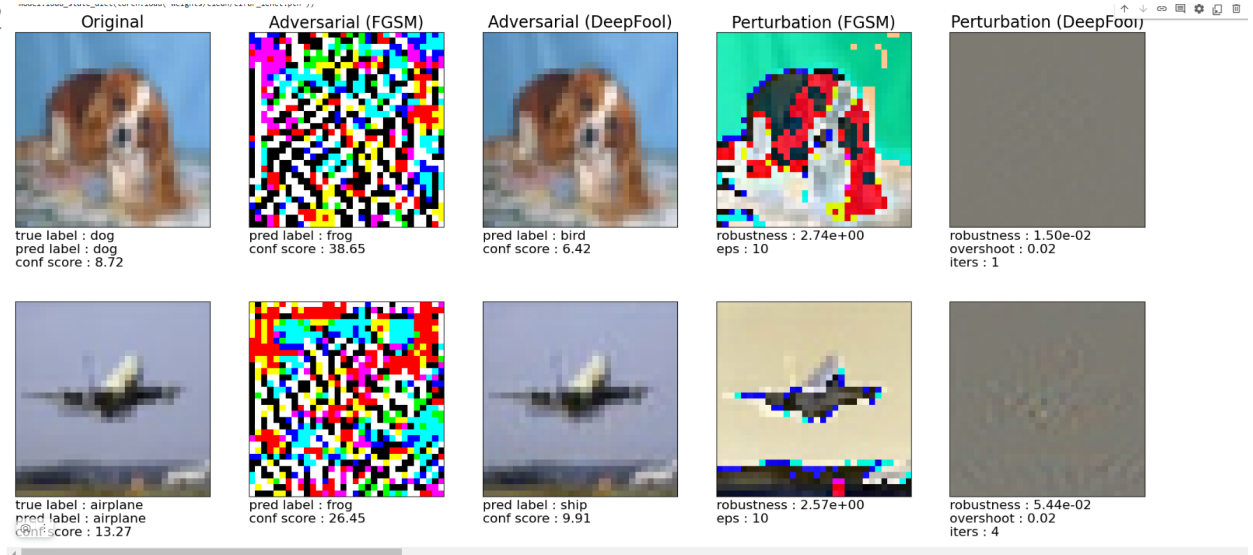
fgsm\_eps = 0.5



fgsm\_eps = 0.9



fgsm\_eps = 10



## **Заключение**

Когда `fgsm_eps` увеличивается, сети становятся уязвимее к атакам. Значительно уязвимее они становятся со значения `fgsm_eps = 0.5`