**专业课程设计1 智能算法与模型测试**

**班级：211413 姓名：王子淳 学号：21376425**

# 实验原理

## 对抗攻击

对抗攻击是一种通过向模型输入数据添加细微扰动，旨在误导模型从而得到错误分类结果的技术。攻击者利用对抗扰动ϵ，将原始样本x转换为对抗样本x′=x+ϵx' = x + \epsilonx′=x+ϵ。对抗样本与原始样本在视觉上差异极小，但会导致模型分类错误。该技术被广泛应用于深度学习模型的安全性评估，例如通过生成伪造的对抗样本，测试分类器对这些伪造数据的鲁棒性。

本实验中使用了多种常见的对抗攻击方法，例如FGSM（Fast Gradient Sign Method）、PGD（Projected Gradient Descent）和DeepFool等。通过这些攻击方法，实验生成了一系列对抗性样本，并通过计算模型的干净正确率和对抗正确率，来评估模型在对抗扰动下的表现。

## 模糊测试

模糊测试是一种自动化测试技术，通过输入随机或变异的数据，以揭示软件中的潜在漏洞和缺陷。实验使用了覆盖率引导的模糊测试方法，通过神经网络的覆盖率信息（如神经元覆盖率、激活阈值等），指导生成更具代表性和多样性的测试样本。通过对模型进行不同标准（如NC、KMNC、SNAC、NLC等）的模糊测试，实验评估了这些标准下神经网络的行为差异，并对生成的样本进行了分析。

实验通过两种不同的模式进行测试：

1、随机模糊测试：通过随机变异输入数据，测试模型的鲁棒性。

2、覆盖引导的模糊测试：基于神经网络的覆盖率引导生成变异样本，测试模型对覆盖率的敏感性。

# 实验流程

（操作步骤，包括调参思路）

## 对抗攻击

先运行指令：

python fool.py -m resnet50 -mod random -a FGSM -b 16 -e 0.1

在该行命令中，我们可修改的模型或者参数如下：

- Model (-m)：

resnet50、mobilenet\_v2、vgg16\_bn.

- Attack (-a)：

FGM、FGSM、MIFGSM；L1PGD、L2PGD、LinfPGD；L2DeepFoolAttack、LinfDeepFoolAttack...

- Batch\_size (-b):

int <= 20

- Epsilon (-e)：

{int/float}

该条命令是在随机模式下，对一批随机选取的16张图像进行对抗攻击，用于测试模型对大批量输入的鲁棒性。

然后根据生成的log文件里面的结果，对指令中的可改模型或参数进行适当的调整，以提高模型的鲁棒性。

## 模糊测试

先进性随机模糊测试，对应的命令如下：

python fuzz\_rand.py -m resnet50 -c NC -y 0.75

然后再进行覆盖引导的模糊测试，对应的指令如下：

python fuzz.py -m resnet50 -c NC -y 0.75

其中，可更改的模型以及参数如下：

- Model (-m)：

resnet50、mobilenet\_v2、vgg16\_bn.

- Criterion (-c)：

NC、KMNC、SNAC、NLC.

- Hyper (-y):

{int/float} or None

最后，对生成的log文件里面的记录进行分析，适当调整模型的选择和参数，使得鲁棒性增加。

# 实验对象

## 第一个实验，（对抗攻击）中，可选的模型有如下几种：

#### **ResNet50**

ResNet50是残差网络（Residual Network）的一种，具有50层深度。其主要创新点是引入了“残差连接”，通过让网络学习残差函数来避免梯度消失问题，从而能够有效地训练深度更大的网络。

#### **MobileNetV2**

MobileNetV2是一种轻量级卷积神经网络，专为移动设备和嵌入式设备设计，重点在于减少计算复杂度和参数量。其核心创新是“倒置残差结构”（Inverted Residuals）和“深度可分离卷积”（Depthwise Separable Convolutions）。

#### **VGG16\_BN**

VGG16是一种经典的卷积神经网络模型，具有16层深度，以其简单且统一的卷积层和池化层设计闻名。VGG16\_BN是其带有批归一化（Batch Normalization）版本，能够加速训练过程并改善模型的泛化能力。

## 第一个实验，（攻击方法）有如下几种：

#### **FGM**

FGM是对抗攻击的一种基本形式，它利用损失函数对输入的梯度生成对抗样本。该方法通过在梯度的方向上施加扰动，使模型产生错误分类。

#### **FGSM**

FGSM是FGM的改进版本，它通过计算损失函数对输入的梯度符号来生成对抗样本。这种方法通过在每个像素上施加一个固定幅度的扰动，从而让模型输出错误的分类结果。

#### **MIFGSM**

MIFGSM是在FGSM基础上的迭代版本，增加了动量项从而避免陷入局部最优。每次迭代都会计算梯度并施加扰动，使攻击效果更强。

#### **PGD**

PGD是一种基于梯度下降的迭代攻击方法，通过在每一步迭代中施加一个小的扰动并将其投影到给定的扰动空间（如L1、L2或Linf范数）内，使对抗扰动控制在规定范围内。PGD被认为是最强的白盒攻击之一。

#### **DeepFool**

DeepFool是一种针对深度神经网络设计的最优攻击方法。它通过寻找最小的扰动来改变模型的分类结果。DeepFool算法能够逐步逼近分类边界，最终生成对抗样本。

## 第二个实验，（标准）有如下几种：

需要说明的是，实验二对应的可选模型和实验一可选模型是一样的，所以此处就不再赘述模型。而是详细分析这几种不一样的“标准”。

**Criterion** 是模糊测试中的评估标准，用于判断神经网络的测试覆盖率。通过不同的标准，模糊测试可以更系统地探索神经网络的行为和潜在漏洞。

1. **模糊测试标准（Criterion）：**

**NC（Neuron Coverage，神经元覆盖率）**：

**定义**：统计神经网络中激活的神经元的数量。即每个神经元是否被激活过，覆盖的神经元越多，测试的深度越大。

**特点**：这是最基础的覆盖率标准，评估模型对输入数据的反应是否足够广泛。

**KMNC（K-Multisection Neuron Coverage，K-多节神经元覆盖）**：

**定义**：将神经元的激活值划分为K个区间，统计每个区间内被覆盖的神经元数目。

**特点**：通过分段激活状态，能够更加详细地评估神经网络的神经元覆盖情况。它是NC的扩展，检测更加精细。

**SNAC（Strong Neuron Activation Coverage，强神经元激活覆盖）**：

**定义**：统计神经元在高激活状态下的覆盖率，重点分析神经元在高激活值范围内的表现。

**特点**：专注于神经网络在极端情况下的反应能力，检测神经元是否在较高的激活状态下也能被充分覆盖。

**NLC（Neural Layer Coverage，神经元层覆盖）**：

**定义**：统计每一层神经元的输出覆盖情况，评估每层神经元的激活分布。

**特点**：从网络的层级结构出发，分析每一层的神经元覆盖情况，能够发现特定层的弱点或漏洞。

# 数据集

python fool.py -m resnet50 -mod random -a FGSM -b 16 -e 0.1

这条指令是在**随机模式**下运行对抗攻击，使用ResNet50模型，通过FGSM攻击方法对从CIFAR10数据集中随机选取的16张图像进行攻击，生成对抗样本。epsilon控制扰动的大小为0.1。

python fool.py -m resnet50 -mod given -a FGSM -i ./datasets/CIFAR10/test/airplane/image31.png -e 0.1

这条指令是在**给定模式**下运行对抗攻击，使用ResNet50模型，针对CIFAR10数据集中指定的单张图像（“飞机”图像）进行FGSM攻击，生成对抗样本，扰动幅度控制为0.1。

# 实验结果

## 对抗攻击

### 基本结果演示

按照讲义上的示例进行操作，对应的结果如下：

----------------------------------------  
clean accuracy:  100.0 %  
----------------------------------------  
robust accuracy:  
Linf norm <= 0.1   : 18.8 %  
----------------------------------------  
robust accuracy for perturbations with  
Linf norm <= 0.1   : 18.8 %  
perturbation sizes:  
[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002  
     0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002  
     0.10000002 0.10000002 0.10000002 0.10000002]SSDS

### Epsilon参数调整

下面几次，依次通过调整Epsilon的大小，来观察Robust Accuracy（鲁棒准确率）的大小变化情况。

对应的Epsilon的大小依次如下：0.1、0.05、0.01、0.005、0.002

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

**robust accuracy:**

**Linf norm <= 0.1 : 18.8 %**

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 18.8 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.05 : 25.0 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.05 : 25.0 %

perturbation sizes:

[0.05000001 0.05000001 0.05000001 0.05000001 0.05000001 0.05000001

0.05000001 0.05000001 0.05000001 0.05000001 0.05000001 0.05000001

0.05000001 0.05000001 0.05000001 0.05000001]

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.01 : 56.2 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.01 : 56.2 %

perturbation sizes:

[0.01000002 0.01000002 0.01000002 0.01000002 0.01000002 0.01000002

0.01000002 0.01000002 0.01000002 0.01000002 0.01000002 0.01000002

0.01000002 0.01000002 0.01000002 0.01000002]

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.005 : 56.2 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.005 : 56.2 %

perturbation sizes:

[0.00500003 0.00500003 0.00500003 0.00500003 0.00500003 0.00500003

0.00500003 0.00500003 0.00500003 0.00500003 0.00500003 0.00500003

0.00500003 0.00500003 0.00500003 0.00500003]

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.002 : 87.5 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.002 : 87.5 %

perturbation sizes:

[0.002 0.002 0.002 0.002 0.002 0.002 0.002 0.002 0.002 0.002 0.002 0.002

0.002 0.002 0.002 0.002]

----------------------------分界线-----------------------------

### 模型调整

Epsilon都是0.1

**当采用的是resnet50模型时，结果如下：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

**robust accuracy:**

**Linf norm <= 0.1 : 18.8 %**

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 18.8 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

**当采用的是mobilenet\_v2模型时，结果如下：**

----------------------------------------

clean accuracy: 93.8 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 0.0 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 0.0 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

**当采用的是vgg16\_bn模型时，结果如下：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 25.0 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 25.0 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

### 攻击算法调整

**CIFAR10-vgg16\_bn-FGM：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 93.8 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 93.8 %

perturbation sizes:

[0.01034397 0.01777251 0.01009119 0.0106132 0.0226216 0.01729318

0.0127268 0.01072434 0.01016185 0.01060668 0.0149473 0.01139575

0.00913954 0.0132215 0.01179254 0.01130641]

**CIFAR10-vgg16\_bn-FGSM：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 25.0 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 25.0 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

**CIFAR10-vgg16\_bn-L1PGD：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 100.0 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 100.0 %

perturbation sizes:

[0.00021452 0.00038101 0.00023729 0.00022194 0.0005089 0.00039554

0.00025344 0.00023535 0.00020787 0.00024739 0.00034842 0.00022954

0.00017494 0.00029236 0.00024617 0.00022066]

**CIFAR10-vgg16\_bn-MIFGSM：**

----------------------------------------

clean accuracy: 100.0 %

----------------------------------------

robust accuracy:

Linf norm <= 0.1 : 12.5 %

----------------------------------------

robust accuracy for perturbations with

Linf norm <= 0.1 : 12.5 %

perturbation sizes:

[0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002 0.10000002 0.10000002

0.10000002 0.10000002 0.10000002 0.10000002]

## 模糊测试

### 基本结果演示

**随机模糊测试：**

**rand-CIFAR10-resnet50-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.357320, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 10.105751s, Epoch: 213, Current coverage: 0.357320, Delta coverage:0.000000, AE: 36, Delta batch: 214

**覆盖引导的模糊测试：**

**CIFAR10-resnet50-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.357320, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 5.309890s, Epoch: 50, Current coverage: 0.375273, Delta coverage:0.017953, AE: 19, Delta batch: 50

Delta time: 11.697994s, Epoch: 100, Current coverage: 0.404366, Delta coverage:0.047046, AE: 37, Delta batch: 100

Delta time: 11.697994s, Epoch: 100, Current coverage: 0.404441, Delta coverage:0.047121, AE: 37, Delta batch: 101

### 覆盖引导的模糊测试相比随机测试的效果

通过对比上面黄色区域的数据大小，很明显覆盖引导的模糊测试的测试效果比随机测试的测试效果好。

### 覆盖标准（及其超参数）调整

注：都是以vgg16\_bn模型为基础进行的

**CIFAR10-vgg16\_bn-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.288508, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 4.397770s, Epoch: 50, Current coverage: 0.321664, Delta coverage:0.033156, AE: 16, Delta batch: 50

Delta time: 7.521806s, Epoch: 83, Current coverage: 0.332770, Delta coverage:0.044262, AE: 29, Delta batch: 83

**CIFAR10-vgg16\_bn-NLC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 89429.156250, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 20.492254s, Epoch: 24, Current coverage: 90186.335938, Delta coverage:757.179688, AE: 7, Delta batch: 24

**CIFAR10-vgg16\_bn-SNAC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.000000, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 2.983459s, Epoch: 50, Current coverage: 0.095606, Delta coverage:0.095606, AE: 13, Delta batch: 50

Delta time: 6.044161s, Epoch: 100, Current coverage: 0.166908, Delta coverage:0.166908, AE: 25, Delta batch: 100

Delta time: 7.263787s, Epoch: 112, Current coverage: 0.186222, Delta coverage:0.186222, AE: 27, Delta batch: 113

### 模型调整

注：都是以NC作为标准的。

**CIFAR10-mobilenet\_v2-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.479843, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 9.750292s, Epoch: 47, Current coverage: 0.485644, Delta coverage:0.005801, AE: 13, Delta batch: 45

**CIFAR10-resnet50-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.357320, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 5.309890s, Epoch: 50, Current coverage: 0.375273, Delta coverage:0.017953, AE: 19, Delta batch: 50

Delta time: 11.697994s, Epoch: 100, Current coverage: 0.404366, Delta coverage:0.047046, AE: 37, Delta batch: 100

Delta time: 11.697994s, Epoch: 100, Current coverage: 0.404441, Delta coverage:0.047121, AE: 37, Delta batch: 101

**CIFAR10-vgg16\_bn-NC：**

Delta time: 0.000000s, Epoch: 0, Current coverage: 0.288508, Delta coverage:0.000000, AE: 0, Delta batch: 0

Delta time: 4.397770s, Epoch: 50, Current coverage: 0.321664, Delta coverage:0.033156, AE: 16, Delta batch: 50

Delta time: 7.521806s, Epoch: 83, Current coverage: 0.332770, Delta coverage:0.044262, AE: 29, Delta batch: 83

# 结果分析

## 对抗攻击

通过上面epsilon大小的调整对于最终实验结果的影响，可以知道总体趋势是随着epsilon的减小，最终模型的鲁棒性逐渐变强。

在epsilon固定的前提下，vgg16\_bn模型的效果优于resnet50的效果优于mobilenet\_v2模型的效果。

对于**vgg16\_bn**模型而言，其对于**L1PGD**攻击的鲁棒性高于对于**FGM**攻击的鲁棒性高于对于**FGSM**攻击的鲁棒性高于对于**MIFGSM**攻击的鲁棒性。

## 模糊测试

对于覆盖标准而言，**NLC**的效果好于**NC**好于**SNAC**。

对于模型而言，**mobilenet\_v2**的效果好于**resnet50**好于**vgg16\_bn**。

# 总结

在本次实验中，我们通过对抗攻击和模糊测试两种方式，评估了不同深度学习模型的鲁棒性和安全性。实验的主要目的是测试模型在面对对抗性样本和随机变异输入时的表现，分析模型的弱点和潜在的改进方向。

模型的鲁棒性：我们发现模型的鲁棒性在不同攻击方式和扰动强度下表现各异。深度更大的VGG16\_BN模型在对抗攻击中表现出较强的鲁棒性，而轻量级的MobileNetV2则在覆盖率测试中具有更好的表现。这说明在面对不同的攻击场景时，模型的架构和复杂度会显著影响其防御能力。

模糊测试的效果：覆盖引导的模糊测试比随机测试更有效，可以更快、更深入地探索模型的潜在漏洞。对于实际应用场景中的神经网络安全测试，覆盖引导的测试策略更具实用价值。

**思路：**

在实验过程中，我引入了多种不同的攻击方式和覆盖标准，并尝试通过调参（如epsilon大小和攻击方法）来优化模型的表现。这种调参方法帮助我更深入地理解不同模型在面对对抗扰动和模糊测试时的差异性。

另外，我通过对比不同模型和不同标准下的表现，总结了各类模型在实际应用中的优缺点，为后续改进模型鲁棒性提供了参考。