数据集：Cifar10（完整数据集每类6000张，一共10个类别）

攻击模型：ResNet20



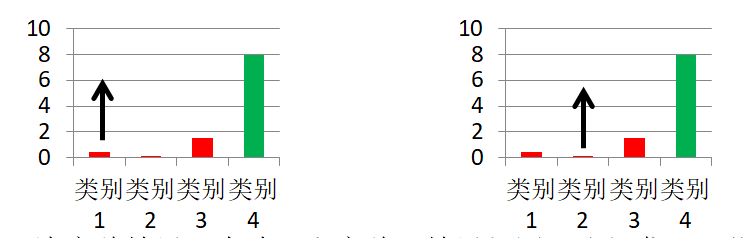
## 1.1 对不同概率的类进行提升概率测试

对图片经过网络后，输出10个类出现的概率进行排序后。（概率最大的类为正确的类）分别对第二、第三、第五、第八的非正确类进行反向传播提升该类的概率，如图4所示。然后再用ResNet20测试，预测正确的比例如表1所示。表中数值越低，表明算法效果越好。

**表 1 对不同出现概率的类进行概率提升**

| 总扰动大小 | 每次扰动大小 | I-FGSM | SIB-FGSM | TIB-FGSM | FIB-FGSM | EIB-FGSM |
| --- | --- | --- | --- | --- | --- | --- |
| 1/255 | 1/255 | 0.78 | 0.782 | 0.818 | 0.836 | 0.840 |
| 2/255 | 1/255 | 0.696 | 0.724 | 0.786 | 0.822 | 0.840 |
| 2/255 | 0.78 | 0.782 | 0.818 | 0.836 | 0.840 |
| 3/255 | 1/255 | 0.602 | 0.654 | 0.758 | 0.854 | 0.882 |
| 2/255 | 0.69 | 0.732 | 0.822 | 0.874 | 0.900 |
| 3/255 | 0.78 | 0.782 | 0.818 | 0.836 | 0.84 |
| 4/255 | 1/255 | 0.594 | 0.656 | 0.762 | 0.856 | 0.906 |
| 2/255 | 0.69 | 0.732 | 0.822 | 0.874 | 0.900 |
| 3/255 | 0.704 | 0.754 | 0.842 | 0.892 | 0.918 |
| 4/255 | 0.78 | 0.782 | 0.818 | 0.836 | 0.840 |
| 5/255 | 1/255 | 0.282 | 0.298 | 0.378 | 0.442 | 0.496 |
| 2/255 | 0.288 | 0.294 | 0.384 | 0.462 | 0.546 |
| 3/255 | 0.34 | 0.352 | 0.44 | 0.526 | 0.572 |
| 4/255 | 0.334 | 0.346 | 0.442 | 0.55 | 0.624 |
| 5/255 | 0.578 | 0.538 | 0.602 | 0.638 | 0.642 |
| 6/255 | 1/255 | 0.27 | 0.296 | 0.382 | 0.488 | 0.61 |
| 2/255 | 0.288 | 0.294 | 0.382 | 0.466 | 0.542 |
| 3/255 | 0.34 | 0.352 | 0.44 | 0.526 | 0.572 |
| 4/255 | 0.334 | 0.346 | 0.442 | 0.55 | 0.624 |
| 5/255 | 0.34 | 0.352 | 0.452 | 0.596 | 0.678 |
| 6/255 | 0.578 | 0.538 | 0.602 | 0.638 | 0.642 |
| 7/255 | 1/255 | 0.184 | 0.218 | 0.276 | 0.376 | 0.486 |
| 2/255 | 0.270 | 0.300 | 0.422 | 0.538 | 0.728 |
| 3/255 | 0.278 | 0.302 | 0.428 | 0.554 | 0.722 |
| 4/255 | 0.334 | 0.346 | 0.442 | 0.55 | 0.624 |
| 5/255 | 0.34 | 0.352 | 0.452 | 0.596 | 0.678 |
| 6/255 | 0.344 | 0.37 | 0.498 | 0.64 | 0.758 |
| 7/255 | 0.578 | 0.538 | 0.602 | 0.638 | 0.642 |
| 8/255 | 1/255 | 0.178 | 0.216 | 0.278 | 0.42 | 0.59 |
| 2/255 | 0.27 | 0.296 | 0.422 | 0.534 | 0.728 |
| 3/255 | 0.278 | 0.302 | 0.43 | 0.554 | 0.724 |
| 4/255 | 0.334 | 0.346 | 0.442 | 0.55 | 0.624 |
| 5/255 | 0.34 | 0.352 | 0.452 | 0.596 | 0.678 |
| 6/255 | 0.344 | 0.37 | 0.498 | 0.64 | 0.758 |
| 7/255 | 0.364 | 0.386 | 0.53 | 0.668 | 0.808 |
| 8/255 | 0.578 | 0.538 | 0.602 | 0.638 | 0.642 |
| 16/255 | 1/255 | 0.022 | 0.022 | 0.046 | 0.078 | 0.134 |
| 2/255 | 0.028 | 0.04 | 0.072 | 0.108 | 0.204 |
| 4/255 | 0.022 | 0.036 | 0.062 | 0.096 | 0.194 |
| 8/255 | 0.098 | 0.092 | 0.132 | 0.22 | 0.294 |
| 12/255 | 0.094 | 0.106 | 0.162 | 0.256 | 0.366 |
| 16/255 | 0.382 | 0.318 | 0.38 | 0.43 | 0.446 |
| 32/255 | 2/255 | 0.008 | 0.008 | 0.01 | 0.008 | 0.02 |
| 4/255 | 0.01 | 0.01 | 0.008 | 0.01 | 0.03 |
| 8/255 | 0.01 | 0.012 | 0.014 | 0.03 | 0.058 |
| 12/255 | 0.014 | 0.02 | 0.046 | 0.05 | 0.102 |
| 16/255 | 0.024 | 0.032 | 0.056 | 0.078 | 0.13 |
| 24/255 | 0.022 | 0.036 | 0.058 | 0.136 | 0.19 |
| 32/255 | 0.278 | 0.226 | 0.236 | 0.246 | 0.284 |

I-FGSM ，原来的算法。SIB-FGSM 挑选第二大概率的类，TIB-FGSM 第三大概率的类，FIB-FGSM第五大概率的类， EIB-FGSM第八大概率的类。缩写解释，如SIB-the second is the best。表中黄色数据是比I-FGSM效果好的数据。

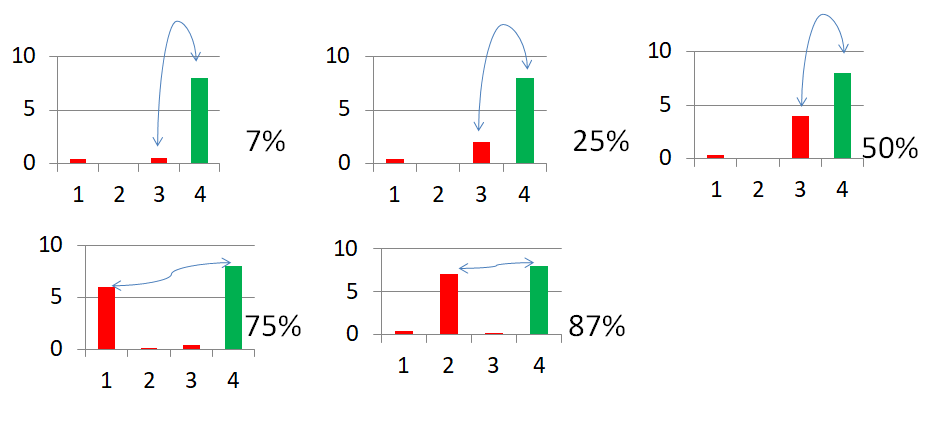


**图 1 提升非正确类的概率**

结论：概率出现越大的非正确类，提升效果越好。也就是说提升除了正确类的第二概率对应的类效果比其他类的效果好。但是，效果并不如I-FGSM算法好，较为接近。接下来对，非正确类第二大概率类进行划分，按照第二概率类占第一大概率类的百分比对整个数据集进行划分。

## 1.2 对样本进行划分测试，寻找有利于SIB-FGSM的样本空间

按照第二大概率类占比第一大概率类的占比来划分，如图2所示。每个10%一个间隔。将图片划分成10份。分别用着两种算法进行测试



**图 2 第二大概率类出现概率与正确类的占比**

**表 2 50%-60%区间表**

| 总扰动大小 | 每次扰动大小 | I-FGSM | SIB-FGSM |
| --- | --- | --- | --- |
| 1/255 | 1/255 | 0.542 | 0.540 |
| 2/255 | 1/255 | 0.402 | 0.442 |
| 2/255 | 0.542 | 0.540 |
| 3/255 | 1/255 | 0.291 | 0.371 |
| 2/255 | 0.400 | 0.451 |
| 3/255 | 0.542 | 0.540 |
| 4/255 | 1/255 | 0.282 | 0.364 |
| 2/255 | 0.400 | 0.451 |
| 3/255 | 0.407 | 0.469 |
| 4/255 | 0.542 | 0.540 |
| 5/255 | 1/255 | 0.049 | 0.073 |
| 2/255 | 0.049 | 0.076 |
| 3/255 | 0.078 | 0.109 |
| 4/255 | 0.073 | 0.109 |
| 5/255 | 0.231 | 0.236 |
| 6/255 | 1/255 | 0.044 | 0.076 |
| 2/255 | 0.049 | 0.076 |
| 3/255 | 0.078 | 0.109 |
| 4/255 | 0.073 | 0.109 |
| 5/255 | 0.076 | 0.124 |
| 6/255 | 0.231 | 0.236 |
| 7/255 | 1/255 | 0.029 | 0.049 |
| 2/255 | 0.047 | 0.076 |
| 3/255 | 0.053 | 0.080 |
| 4/255 | 0.073 | 0.109 |
| 5/255 | 0.076 | 0.124 |
| 6/255 | 0.078 | 0.127 |
| 7/255 | 0.231 | 0.236 |
| 8/255 | 1/255 | 0.029 | 0.049 |
| 2/255 | 0.047 | 0.076 |
| 3/255 | 0.053 | 0.078 |
| 4/255 | 0.073 | 0.109 |
| 5/255 | 0.076 | 0.124 |
| 6/255 | 0.078 | 0.127 |
| 7/255 | 0.080 | 0.131 |
| 8/255 | 0.231 | 0.236 |
| 16/255 | 1/255 | 0.002 | 0.007 |
| 2/255 | 0.004 | 0.009 |
| 4/255 | 0.002 | 0.009 |
| 8/255 | 0.011 | 0.020 |
| 12/255 | 0.011 | 0.022 |
| 16/255 | 0.102 | 0.104 |
| 32/255 | 2/255 | 0.000 | 0.000 |
| 4/255 | 0.000 | 0.000 |
| 8/255 | 0.000 | 0.000 |
| 12/255 | 0.000 | 0.002 |
| 16/255 | 0.000 | 0.011 |
| 24/255 | 0.000 | 0.018 |
| 32/255 | 0.071 | 0.069 |

**表 3 20%-30表**

|  |  |  |  |
| --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM | SIB-FGSM |
| 1/255 | 1/255 | 0.877 | 0.873 |
| 2/255 | 1/255 | 0.806 | 0.817 |
| 2/255 | 0.877 | 0.873 |
| 3/255 | 1/255 | 0.707 | 0.764 |
| 2/255 | 0.806 | 0.826 |
| 3/255 | 0.877 | 0.873 |
| 4/255 | 1/255 | 0.697 | 0.758 |
| 2/255 | 0.806 | 0.826 |
| 3/255 | 0.816 | 0.843 |
| 4/255 | 0.877 | 0.873 |
| 5/255 | 1/255 | 0.300 | 0.325 |
| 2/255 | 0.300 | 0.334 |
| 3/255 | 0.380 | 0.388 |
| 4/255 | 0.370 | 0.385 |
| 5/255 | 0.638 | 0.603 |
| 6/255 | 1/255 | 0.282 | 0.324 |
| 2/255 | 0.300 | 0.335 |
| 3/255 | 0.380 | 0.388 |
| 4/255 | 0.370 | 0.385 |
| 5/255 | 0.370 | 0.396 |
| 6/255 | 0.638 | 0.603 |
| 7/255 | 1/255 | 0.180 | 0.243 |
| 2/255 | 0.276 | 0.326 |
| 3/255 | 0.281 | 0.337 |
| 4/255 | 0.370 | 0.385 |
| 5/255 | 0.370 | 0.396 |
| 6/255 | 0.383 | 0.416 |
| 7/255 | 0.638 | 0.603 |
| 8/255 | 1/255 | 0.177 | 0.243 |
| 2/255 | 0.276 | 0.328 |
| 3/255 | 0.280 | 0.336 |
| 4/255 | 0.370 | 0.385 |
| 5/255 | 0.370 | 0.396 |
| 6/255 | 0.383 | 0.416 |
| 7/255 | 0.396 | 0.443 |
| 8/255 | 0.638 | 0.603 |
| 16/255 | 1/255 | 0.021 | 0.030 |
| 2/255 | 0.031 | 0.043 |
| 4/255 | 0.031 | 0.042 |
| 8/255 | 0.099 | 0.104 |
| 12/255 | 0.096 | 0.115 |
| 16/255 | 0.429 | 0.359 |
| 32/255 | 2/255 | 0.001 | 0.003 |
| 4/255 | 0.002 | 0.004 |
| 8/255 | 0.003 | 0.005 |
| 12/255 | 0.006 | 0.017 |
| 16/255 | 0.022 | 0.029 |
| 24/255 | 0.016 | 0.045 |
| 32/255 | 0.278 | 0.205 |

**表 4 0-10%表**

|  |  |  |  |
| --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM | SIB-FGSM |
| 1/255 | 1/255 | 0.979 | 0.971 |
| 2/255 | 1/255 | 0.950 | 0.956 |
| 2/255 | 0.979 | 0.971 |
| 3/255 | 1/255 | 0.912 | 0.943 |
| 2/255 | 0.950 | 0.966 |
| 3/255 | 0.979 | 0.971 |
| 4/255 | 1/255 | 0.908 | 0.947 |
| 2/255 | 0.950 | 0.966 |
| 3/255 | 0.950 | 0.968 |
| 4/255 | 0.979 | 0.971 |
| 5/255 | 1/255 | 0.618 | 0.599 |
| 2/255 | 0.618 | 0.609 |
| 3/255 | 0.725 | 0.676 |
| 4/255 | 0.721 | 0.670 |
| 5/255 | 0.916 | 0.868 |
| 6/255 | 1/255 | 0.592 | 0.609 |
| 2/255 | 0.618 | 0.609 |
| 3/255 | 0.725 | 0.676 |
| 4/255 | 0.721 | 0.670 |
| 5/255 | 0.712 | 0.681 |
| 6/255 | 0.916 | 0.868 |
| 7/255 | 1/255 | 0.416 | 0.481 |
| 2/255 | 0.584 | 0.609 |
| 3/255 | 0.582 | 0.607 |
| 4/255 | 0.721 | 0.670 |
| 5/255 | 0.712 | 0.681 |
| 6/255 | 0.716 | 0.689 |
| 7/255 | 0.916 | 0.868 |
| 8/255 | 1/255 | 0.403 | 0.487 |
| 2/255 | 0.584 | 0.611 |
| 3/255 | 0.584 | 0.609 |
| 4/255 | 0.721 | 0.670 |
| 5/255 | 0.712 | 0.681 |
| 6/255 | 0.716 | 0.689 |
| 7/255 | 0.735 | 0.725 |
| 8/255 | 0.916 | 0.868 |
| 16/255 | 1/255 | 0.065 | 0.088 |
| 2/255 | 0.088 | 0.109 |
| 4/255 | 0.082 | 0.109 |
| 8/255 | 0.290 | 0.246 |
| 12/255 | 0.271 | 0.284 |
| 16/255 | 0.758 | 0.618 |
| 32/255 | 2/255 | 0.015 | 0.019 |
| 4/255 | 0.019 | 0.021 |
| 8/255 | 0.029 | 0.034 |
| 12/255 | 0.034 | 0.048 |
| 16/255 | 0.109 | 0.074 |
| 24/255 | 0.078 | 0.101 |
| 32/255 | 0.590 | 0.416 |

结论：从表2至表4中可以看到随着第二大概率的类占比第一大概率的越大，I-FGSM算法性能下降很快，SIB-FGSM性能下降的没有那么快。效果稍好一些。此时对I-FGSM算法的优化可以为，先对整个输入的样本空间进行划分，然后对0-10%这部分占比的空间应用SIB-FGSM算法，可以达到对SIB-FGSM算法的性能优化。

res-20 占比

[476, 1089, 1145, 996, 672, 450, 240, 128, 47, 37] 0.1

res-44

[1076, 1460, 1157, 818, 422, 229, 81, 27, 5, 5] 0.21

res110

[[1341, 1670, 1152, 671, 287, 120, 30, 8, 1, 0] 0.26

res1202

[1411, 1704, 1156, 636, 248, 101, 22, 2, 0, 0] 0.28

可以看出网络模型的复杂度越高，第一类占比越多。话句话说，正确类的概率值越高，其他类的概率就越低。同样第二大概率的值也越低。

经过思考，我发现我们的算法有一定的缺点，在迭代次数为n次过程中我们发现随着每次迭代，有可能刚开始我们选择的第二大的概率类，随着迭代的过程渐渐成了，非第二大概率的类，基于此我们要在每次迭代的过程中不断去寻找第二大概率的类，然后在提升重新选择后的第二大的概率类的概率。不断的在每次的迭代的过程中去寻找第二大概率的类。以下我们将这个算法命名为DSIB-FGSM。

## 1.3动态选取第二大概率的类进行提升概率DSIB-FGSM

**表 5 动态提升第二大概率类**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM | SIB-FGSM | DSIB-FGSM |
| 1/255 | 1/255 | 0.78 | 0.782 | 0.782 |
| 2/255 | 1/255 | 0.696 | 0.724 | 0.692 |
| 2/255 | 0.78 | 0.782 | 0.782 |
| 3/255 | 1/255 | 0.602 | 0.654 | 0.594 |
| 2/255 | 0.69 | 0.732 | 0.688 |
| 3/255 | 0.78 | 0.782 | 0.782 |
| 4/255 | 1/255 | 0.594 | 0.656 | 0.594 |
| 2/255 | 0.69 | 0.732 | 0.688 |
| 3/255 | 0.704 | 0.754 | 0.712 |
| 4/255 | 0.78 | 0.782 | 0.782 |
| 5/255 | 1/255 | 0.282 | 0.298 | 0.262 |
| 2/255 | 0.288 | 0.294 | 0.258 |
| 3/255 | 0.34 | 0.352 | 0.320 |
| 4/255 | 0.334 | 0.346 | 0.312 |
| 5/255 | 0.578 | 0.538 | 0.538 |
| 6/255 | 1/255 | 0.27 | 0.296 | 0.252 |
| 2/255 | 0.288 | 0.294 | 0.258 |
| 3/255 | 0.34 | 0.352 | 0.320 |
| 4/255 | 0.334 | 0.346 | 0.312 |
| 5/255 | 0.34 | 0.352 | 0.318 |
| 6/255 | 0.578 | 0.538 | 0.538 |
| 7/255 | 1/255 | 0.184 | 0.218 | 0.168 |
| 2/255 | 0.27 | 0.3 | 0.250 |
| 3/255 | 0.278 | 0.302 | 0.260 |
| 4/255 | 0.334 | 0.346 | 0.312 |
| 5/255 | 0.34 | 0.352 | 0.318 |
| 6/255 | 0.344 | 0.37 | 0.326 |
| 7/255 | 0.578 | 0.538 | 0.538 |
| 8/255 | 1/255 | 0.178 | 0.216 | 0.158 |
| 2/255 | 0.27 | 0.296 | 0.248 |
| 3/255 | 0.278 | 0.302 | 0.260 |
| 4/255 | 0.334 | 0.346 | 0.312 |
| 5/255 | 0.34 | 0.352 | 0.318 |
| 6/255 | 0.344 | 0.37 | 0.326 |
| 7/255 | 0.364 | 0.386 | 0.338 |
| 8/255 | 0.578 | 0.538 | 0.538 |
| 16/255 | 1/255 | 0.022 | 0.022 | 0.018 |
| 2/255 | 0.028 | 0.04 | 0.022 |
| 4/255 | 0.022 | 0.036 | 0.022 |
| 8/255 | 0.098 | 0.092 | 0.066 |
| 12/255 | 0.094 | 0.106 | 0.066 |
| 16/255 | 0.382 | 0.318 | 0.318 |
| 32/255 | 2/255 | 0.008 | 0.008 | 0.004 |
| 4/255 | 0.01 | 0.01 | 0.006 |
| 8/255 | 0.01 | 0.012 | 0.008 |
| 12/255 | 0.014 | 0.02 | 0.008 |
| 16/255 | 0.024 | 0.032 | 0.018 |
| 24/255 | 0.022 | 0.036 | 0.018 |
| 32/255 | 0.278 | 0.226 | 0.226 |

结论，从表5可以看出此时的实验效果已经比I-FGSM算法效果要好。此时我们的数据是每类50张图片，我们将数据扩大到每类500张图片。放大实验，单个类500张图片

**表 6 动态提升第二大概率类放大实验**

|  |  |  |  |
| --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM-500 | SSIB-500 |
| 1/255 | 1/255 | 0.775 | 0.770 |
| 2/255 | 1/255 | 0.692 | 0.687 |
| 2/255 | 0.775 | 0.770 |
| 3/255 | 1/255 | 0.597 | 0.595 |
| 2/255 | 0.690 | 0.686 |
| 3/255 | 0.775 | 0.770 |
| 4/255 | 1/255 | 0.589 | 0.590 |
| 2/255 | 0.690 | 0.686 |
| 3/255 | 0.703 | 0.704 |
| 4/255 | 0.775 | 0.770 |
| 5/255 | 1/255 | 0.266 | 0.242 |
| 2/255 | 0.269 | 0.244 |
| 3/255 | 0.335 | 0.302 |
| 4/255 | 0.326 | 0.297 |
| 5/255 | 0.551 | 0.524 |
| 6/255 | 1/255 | 0.258 | 0.235 |
| 2/255 | 0.269 | 0.244 |
| 3/255 | 0.335 | 0.302 |
| 4/255 | 0.326 | 0.297 |
| 5/255 | 0.327 | 0.300 |
| 6/255 | 0.551 | 0.524 |
| 7/255 | 1/255 | 0.172 | 0.162 |
| 2/255 | 0.256 | 0.233 |
| 3/255 | 0.258 | 0.236 |
| 4/255 | 0.326 | 0.297 |
| 5/255 | 0.327 | 0.300 |
| 6/255 | 0.334 | 0.309 |
| 7/255 | 0.551 | 0.524 |
| 8/255 | 1/255 | 0.168 | 0.158 |
| 2/255 | 0.256 | 0.233 |
| 3/255 | 0.258 | 0.236 |
| 4/255 | 0.326 | 0.297 |
| 5/255 | 0.327 | 0.300 |
| 6/255 | 0.334 | 0.309 |
| 7/255 | 0.346 | 0.322 |
| 8/255 | 0.551 | 0.524 |
| 16/255 | 1/255 | 0.022 | 0.020 |
| 2/255 | 0.031 | 0.026 |
| 4/255 | 0.029 | 0.024 |
| 8/255 | 0.103 | 0.073 |
| 12/255 | 0.098 | 0.073 |
| 16/255 | 0.376 | 0.321 |
| 32/255 | 2/255 | 0.003 | 0.002 |
| 4/255 | 0.004 | 0.003 |
| 8/255 | 0.006 | 0.005 |
| 12/255 | 0.010 | 0.007 |
| 16/255 | 0.030 | 0.016 |
| 24/255 | 0.025 | 0.014 |
| 32/255 | 0.267 | 0.201 |

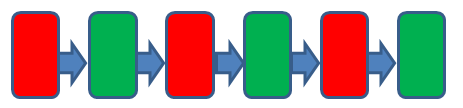
结论：从上表可以看出DSIB-FGSM比I-FGSM算法效果要好。

## 1.4 融合两个损失函数。

基于总扰动是6/255，单次扰动是1/255，单个分类500张图片。

### 1.4.1交替使用两个损失函数（I-FGSM和DSIB-FGSM两个损失函数）

图3可以理解为，压低正确类出现的概率和提升错误类出现的概率两个损失函数交替使用。结果数据如表7所示。



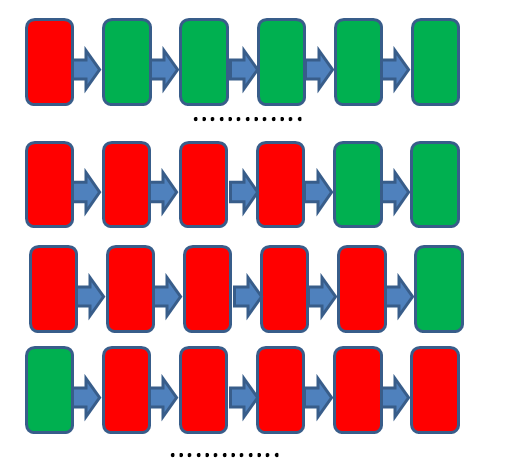
**图 3 交替使用两个损失函数**

**表 7 在总扰动6/255下交替使用两个损失函数**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM-500 | DSIB-500 | 交替损失函数 |
| 6/255 | 1/255 | 0.258 | 0.235 | 0.240 |
| 2/255 | 0.269 | 0.244 | 0.247 |
| 3/255 | 0.335 | 0.302 | 0.309 |
| 4/255 | 0.326 | 0.297 | 0.302 |
| 5/255 | 0.327 | 0.300 | 0.305 |
| 6/255 | 0.551 | 0.524 | 0.524 |

### 1.4.2按一定比例分别迭代两个损失函数

对两个损失函数按照一定比例进行融合，如图4所示。



**图 4 按照一定比例融合两个损失函数**

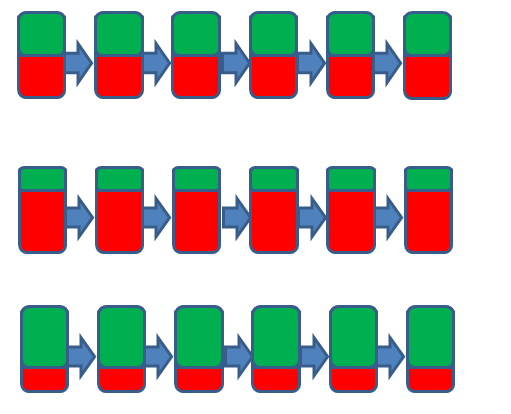
在表8中，S代表DSIB-FGSM个数，i代表I-FGSM的个数。

**表 8 两个损失函数不同比例融合**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | 1s-5i | 2s-4i | 3s-3i | 4s-2i | 5s-1i | 5i-1s | 4i-2s | 3i-3s | 2i-4s | 1i-5s |
| 6/255 | 1/255 | 0.240 | 0.234 | 0.235 | 0.235 | 0.235 | 0.258 | 0.258 | 0.258 | 0.256 | 0.252 |
| 2/255 | 0.247 | 0.244 | 0.244 | 0.244 | 0.244 | 0.269 | 0.269 | 0.268 | 0.268 | 0.265 |
| 3/255 | 0.309 | 0.302 | 0.302 | 0.302 | 0.302 | 0.335 | 0.335 | 0.327 | 0.335 | 0.327 |
| 4/255 | 0.302 | 0.297 | 0.297 | 0.297 | 0.297 | 0.326 | 0.326 | 0.321 | 0.326 | 0.321 |
| 5/255 | 0.305 | 0.300 | 0.300 | 0.300 | 0.300 | 0.327 | 0.327 | 0.323 | 0.327 | 0.323 |
| 6/255 | 0.524 | 0.524 | 0.524 | 0.524 | 0.524 | 0.551 | 0.551 | 0.551 | 0.551 | 0.551 |

### 1.4.3单次迭代中融合两个损失函数

与上一小节不同的是，本次融合是在一次迭代的过程中，按照一定比例融合两个损失函数为一个新的损失函数，然后在单次迭代中用润和后的损失函数。如图5所示的几种融合方式。



**图 5 纵向融合损失函数**

**表 9 单次迭代融合两个损失函数**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 总扰动大小 | 每次扰动大小 | I-FGSM-500 | DSIB-500 | 奇偶 | 1：1 | 1：3 | 3：1 | 1s：9i | 0.01s:0.99i |
| 6/255 | 1/255 | 0.258 | 0.235 | 0.240 | 0.235 | 0.235 | 0.235 | 0.235 | 0.235 |
| 2/255 | 0.269 | 0.244 | 0.247 | 0.244 | 0.244 | 0.244 | 0.244 | 0.245 |
| 3/255 | 0.335 | 0.302 | 0.309 | 0.302 | 0.302 | 0.302 | 0.302 | 0.302 |
| 4/255 | 0.326 | 0.297 | 0.302 | 0.297 | 0.297 | 0.297 | 0.298 | 0.299 |
| 5/255 | 0.327 | 0.300 | 0.305 | 0.300 | 0.300 | 0.300 | 0.300 | 0.301 |
| 6/255 | 0.551 | 0.524 | 0.524 | 0.524 | 0.524 | 0.524 | 0.524 | 0.523 |

## 1.5 将算法扩大到整个数据集DSIB-FGSM算法

由于融合两个损失函数的版本效果不太明显，最后在整个数据集上验证的时候选择了动态提升第二大类概率的DSIB-FGSM算法进行了整个数据集的实验。实验结果表10所示。其数据的相对规律和在500张图片上的测试结果一致。

**表 10 DSIB-FGSM在整个数据集上的测试结果**

| 总扰动-单次扰动 | 4-1 | 6-1 | 8-1 | 9-1 | 12-1 |
| --- | --- | --- | --- | --- | --- |
| DSIB-FGSM | 0.593 | 0.236 | 0.155 | 0.072 | 0.047 |
| I-FGSM | 0.595 | 0.258 | 0.166 | 0.086 | 0.056 |

## 1.6 小结

优化后的I-FGSM，DSIB-FGSM可用如下公式表达。

（1）

注：1.X输入图片

2.

3.总扰动大小

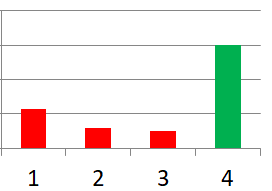
4.动态寻找最大概率的非正确的类

5.（X,y）交叉熵损失函数

6.函数是对超对总扰动范围的数值或者255进行剪裁

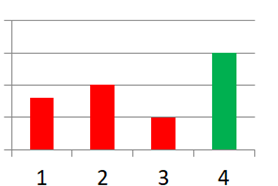
采用DSIB-FGSM后算法的效果提升范围在1%-3%左右。可以对算法进行两个损失函数的融合可以达到更好一些的效果，对此我们只做了一部分的探讨和参数选择。对于性能提升的可能的解释，DSIB-FGSM损失函数对达到让图片分类错误这一目的的描述更为精确，另外一个解释是对于图片的分类问题，提升一个类的概率，比降低一个类正确的概率要“容易”。对于描述更精确这个解释，我们可以想象以下这个场景。

开始的时候正确类的概率。如图6所示



**图 6 未经过算法干扰的概率分布图**

经过I-FGSM算法干扰后。类4的损失被分配到其他三个类上。有这种可能这种损失被均匀的分配到了三个类上，导致类4的概率依然很高，从而导致没有成功让图片被错误分类。



**图 7 经过I-FGSM**算法后的正确类的分布

DSIB-FGSM算法没有选择“被动的等待错误分类概率提高”，而是主动的提高最有可能被提高概率类，来代替之前的正确类。

# 总结

本文将DSIB-FGSM和Tai Chi-FGSM算法平均性能提升2%（在不同总扰动和单次扰动下，提升空间在0.1%到3%之间），在Cifar-10数据集基础上建立了基于优化后I-FGSM算法的ResNes20模型。同时，还需要在更多的数据中去验证。

因为时间的关系，我们还有很多工作没有做。例如Tai Chi版本的调参，模型和数据集的复杂度的变换，以及其他基于I-FGSM的算法，BIM，PGD的改进。通过构建了对应的I-FGSM模型，对I-FGSM算法的主要改进是对损失函数进行改进，比如静态提升第二大概率类的概率，动态提升第二大概率类以及融合两种损失函数三种尝试。I-FGSM算法虽然有较好的识别效果，但是还有进一步改进的空间，来还可以迭代不同损失函数的方法去训练神经网络，后续将寻找更高效的算法来进行优化。