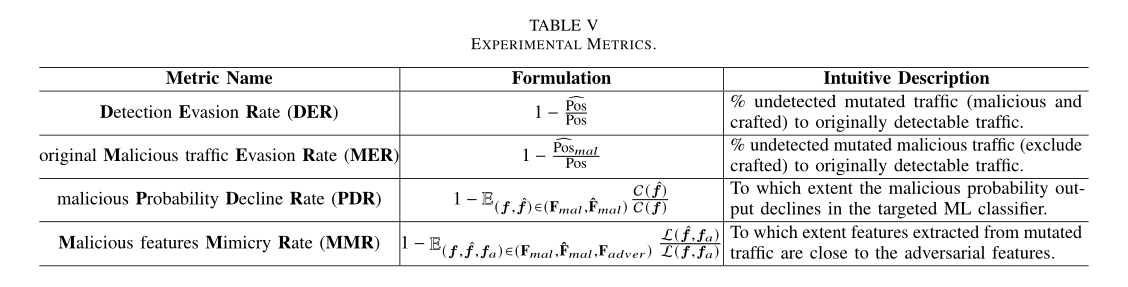
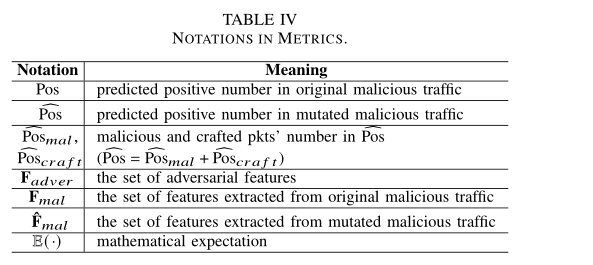
## 总目标

###### NIDS配置：FE（bytes encoding）+ ML（textCNN）

## **PGA（FE knowledge==100%）**

### 2.1实验目标

在完全了解bytes encoding的情况下，对textCNN进行攻击，以PDR/MMR为评价指标。



假正例率 FPR = FP / （TN + FP）

表示的，预测为良性流量例但真实情况为恶性的比率。

PDR=1-（改造后恶意包个数/原恶意包个数）

### 2.1实验步骤

* + 1. 准备流量数据：准备train和test数据集。采用mawilab数据集进行测试。
* 改造原始textCNN字节编码文件init.py

|  |
| --- |
| Python extractor.py -i 0.pcap -o 0.npy |

Python extractor.py -i mawilab\_20\_30w.pcap -o mawilab\_20\_30w.npy

* + 1. 修改textCNN，把它作为目标NIDS中的ML。
* 改造原始textCNN字节编码文件mavilab.py

Train:

|  |
| --- |
| Python model.py -M train |

Test:

|  |
| --- |
| Python model.py -M test |

* + 1. 归一化

使用传统方法和使用Kitsune Normalizer均无用。突然想到不需要归一化，因为字节编码里面得到的都是0，1，似乎没有太大必要归一化，对代码进行改动，隐去归一化部分。



* + 1. 生成对抗性特征

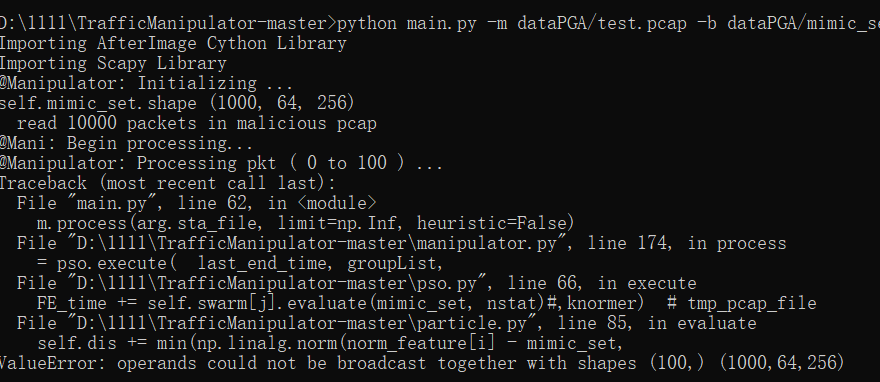
数据准备：

test.pcap ：10000条数据，就是测试集

mimic\_set.npy：从这10000条里面选1000条良性（先转npy再截取）

|  |
| --- |
| python main.py -m dataPGA/t1000.pcap -b dataPGA/mimic\_set.npy |

python main.py -m 2.pcap -b dataPGA/mimic\_set.npy



原因：维度的原因。

具体分析代码，可以看到报错信息中的norm\_feature的shape是(100,100)表示100个包里面的100个向量，即论文中的算法每次处理100个包。Mimic\_set是我们要模仿的流量，他是（1000,64,256）维度的。这两个量之间是无法相减的，原因在于字节编码生成的维度和Manipulator.py文件中使用到的afterImage提取器提取后的特征维度不能相容。

|  |
| --- |
| # Create global feature extractor          init\_scapy\_in = rdpcap(init\_pcap\_file)          self.global\_FE = Kitsune(init\_scapy\_in, np.Inf) |

如果FGA中要改动GAN-PSO中使用的afterImage提取器提取特征进行迭代，是一件很麻烦的事情。

PSO算法学习：

1粒子群算法简介

粒子群算法(Particle Swarm Optimization,简称PSO)是1995年Eberhart博士和Kennedy博士一起提出的。粒子群算法是通过模拟鸟群捕食行为设计的一种群智能算法。区域内有大大小小不同的食物源，鸟群的任务是找到最大的食物源（全局最优解），鸟群的任务是找到这个食物源。鸟群在整个搜寻的过程中，通过相互传递各自位置的信息，让其他的鸟知道食物源的位置最终，整个鸟群都能聚集在食物源周围，即我们所说的找到了最优解，问题收敛。学者受自然界的启发开发了诸多类似智能算法，如蚁群算法、布谷鸟搜索算法、鱼群算法、捕猎算法等等。

2算法原理

这是一个根据鸟群觅食行为衍生出启发式的算法，现在有一群鸟，他们一起出发觅食，目标就是找到可行域食物最丰富的位置。小鸟们在同一个微信群聊中，可以不断共享自己找到的事物最丰富的地方。策略如下：

　　1． 每只鸟随机找一个地方，按照一个随机的方向出发。

　　2． 每飞一分钟后， 每只鸟将自己找到的最优地点以及事物存量共享到群里，然后计算出群体找到的最优位置。

　　3． 每只鸟回顾自己的路径，综合考虑自己走过的最优的位置和群体最优位置决定下一步的方向。

　　4． 如果大家都到了同一个地方附近，就停止寻找，否则重复2 ，3 步。

整个群体的位置更新如下图，每一个红点即一个粒子： （图片来自 scikit-opt）

具体见[(148条消息) 粒子群算法原理\_二郎银的博客-CSDN博客\_粒子群算法原理](https://blog.csdn.net/weixin_43705953/article/details/111510906)

|  |
| --- |
| pklList 恶意流量  grp\_size=100 每次处理100个包  pso\_iter（迭代次数）,pso\_num(粒子个数), pso\_size（组数） = 3, 6, 3  max\_cft\_pkt 与一个原始数据包聚合的最大手工数据包数量。是精心编制的数据包数与原始数据包数的比率  max\_time\_extend 变异流量中每两个变异数据包的间隔时间不超过max\_time\_extend 倍原始间隔时间。是变异流量与原始流量之间经过的时间比率  min\_time\_extend:如上，接最小  `grp\_size`：每次处理变异的网络数据包数量（注意它与上面的 `grp\_size` 不同）  proto\_max\_lmt： v：最大协议层数，与TCP/UDP IP有关 |

|  |
| --- |
| proto\_max\_lmt = []  # maximum protocol layer number（最大协议层数）      for i in range(grp\_size):          if groupList[i].haslayer(TCP) or groupList[i].haslayer(                  UDP) or groupList[i].haslayer(ICMP):              proto\_max\_lmt.append(3.)          elif groupList[i].haslayer(IP) or groupList[i].haslayer(                  IPv6) or groupList[i].haslayer(ARP):              proto\_max\_lmt.append(2.)          elif groupList[i].haslayer(Ether):              proto\_max\_lmt.append(1.)          else:              proto\_max\_lmt.append(0.) |

|  |
| --- |
| class Unit:      def \_\_init\_\_(self, grp\_size, max\_cft\_pkt):          #生成一个100\*2矩阵          self.mal = np.zeros((grp\_size, 2))          # print(grp\_size,max\_cft\_pkt)          #生成一个100\*5\*3矩阵          self.craft = np.zeros((grp\_size, max\_cft\_pkt, 3)) |

PSO算法

1. 基本参数设置：

max\_iter, particle\_num, local\_grp\_size = 3, 6, 3

STA\_best\_X,STA\_best\_feature, STA\_best\_pktList, STA\_gbl\_dis, STA\_avg\_dis,STA\_best\_all\_feature,fe\_time

X

|  |
| --- |
| self.mal = np.zeros((grp\_size, 2))  X.mal[i][0]=groupList[i].time + ics\_time 修改后的时间  X.mal[i][1] 变异次数 |

|  |
| --- |
| self.craft = np.zeros((grp\_size, max\_cft\_pkt, 3))  X.craft[nxt\_mal\_no][cft\_no][0] = X.mal[nxt\_mal\_no][0] - slot\_time  X.craft[nxt\_mal\_no][cft\_no][1] = random.choice([1., 2., 3.])  X.craft[nxt\_mal\_no][cft\_no][2] = random.uniform(0, mtu)  nxt\_mal\_no表示第几个包，cft\_no表示变异次数 [0]是超过时间片的时间，[1]是协议的层级，[2]是最大传输单元 |

proto\_max\_lmt

每个包的协议层级

Class PSO

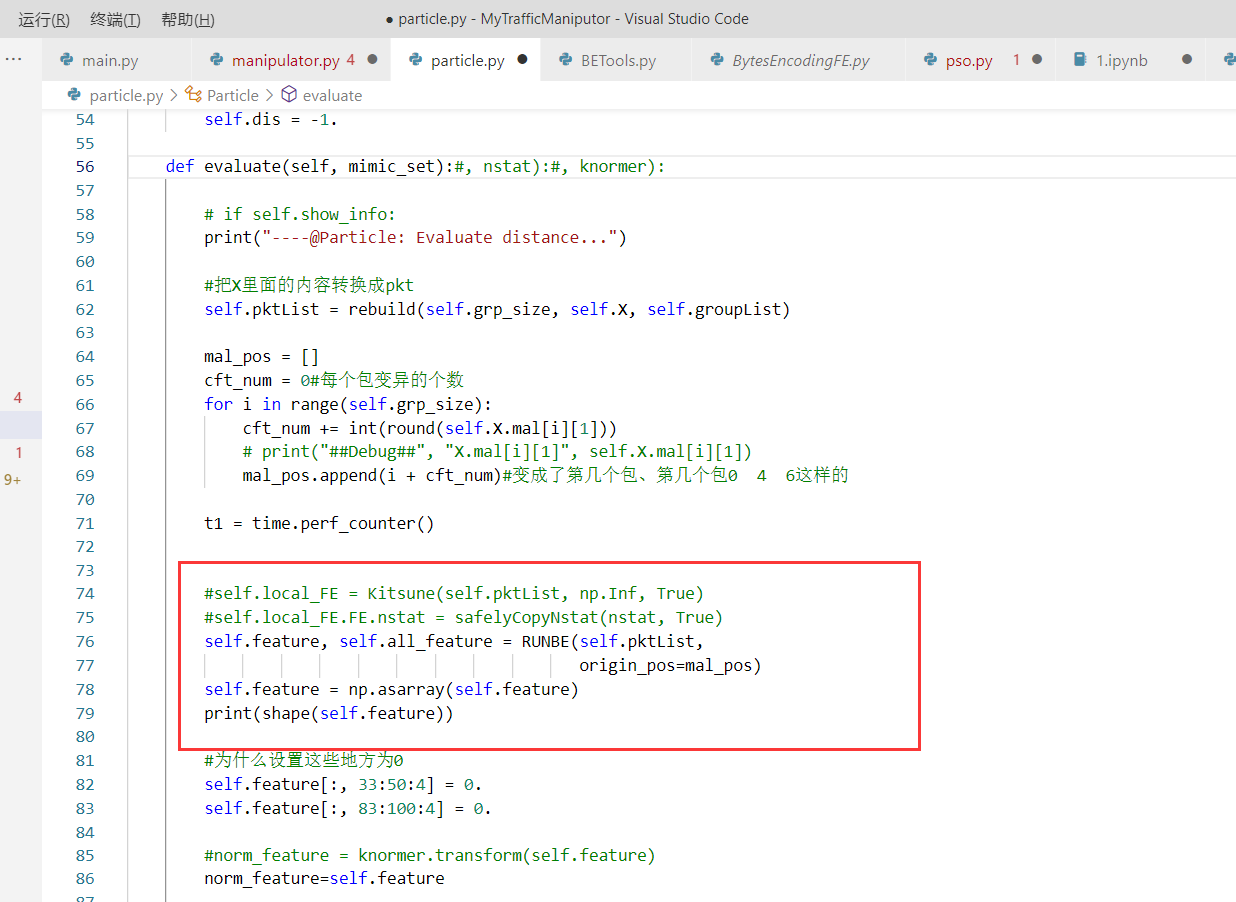
|  |
| --- |
| self.max\_iter = max\_iter#最大迭代次数          self.particle\_num = particle\_num#粒子个数          self.grp\_size = grp\_size#每组大小          self.grp\_best\_dis = [-1.] \* (self.particle\_num // self.grp\_size)          self.grp\_best\_index = [-1] \* (self.particle\_num // self.grp\_size)          self.global\_best\_dis = -1.          self.global\_best\_index = -1          self.global\_best\_pktlist = None            self.swarm = []          self.STA\_glb\_dis\_list = []          self.STA\_avg\_dis\_list = [] |

Class Particle

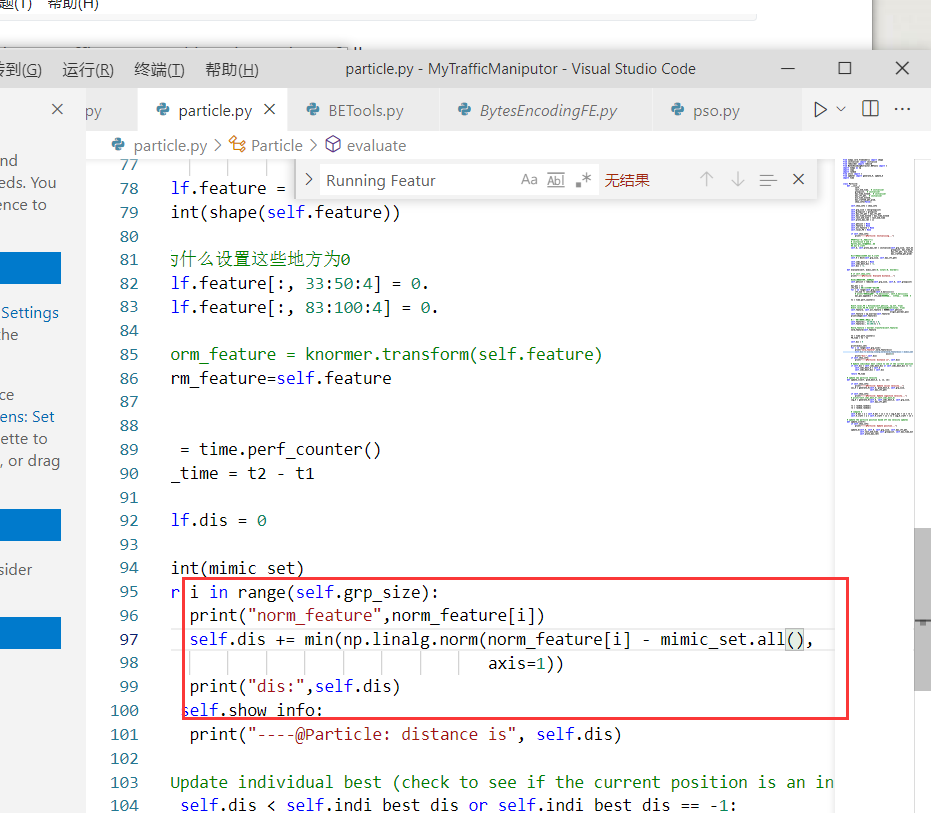
|  |
| --- |
| self.show\_info = show\_info          self.grp\_size = len(groupList)          self.groupList = groupList          self.max\_cft\_pkt = max\_cft\_pkt          self.max\_time\_extend = max\_time\_extend          self.last\_end\_time = last\_end\_time          self.proto\_max\_lmt = []          self.pktList = None          self.feature = None          self.all\_feature = None          self.local\_FE = None          if self.show\_info:              print("----@Particle: Initializing...")          #PSO算法中X和V的初始值          # initialize X and V          #proto\_max\_lmt最大协议层          #X.mal X.craft          self.X, self.proto\_max\_lmt = initialize(self.grp\_size, last\_end\_time,                                                  groupList, max\_time\_extend,                                                  max\_cft\_pkt, min\_time\_extend,                                                  max\_crafted\_pkt\_prob)          #初始化两个矩阵V.mal V.craft          self.V = Unit(self.grp\_size, self.max\_cft\_pkt)          self.indi\_best\_X = None          self.indi\_best\_dis = -1.          self.dis = -1. |
|  |

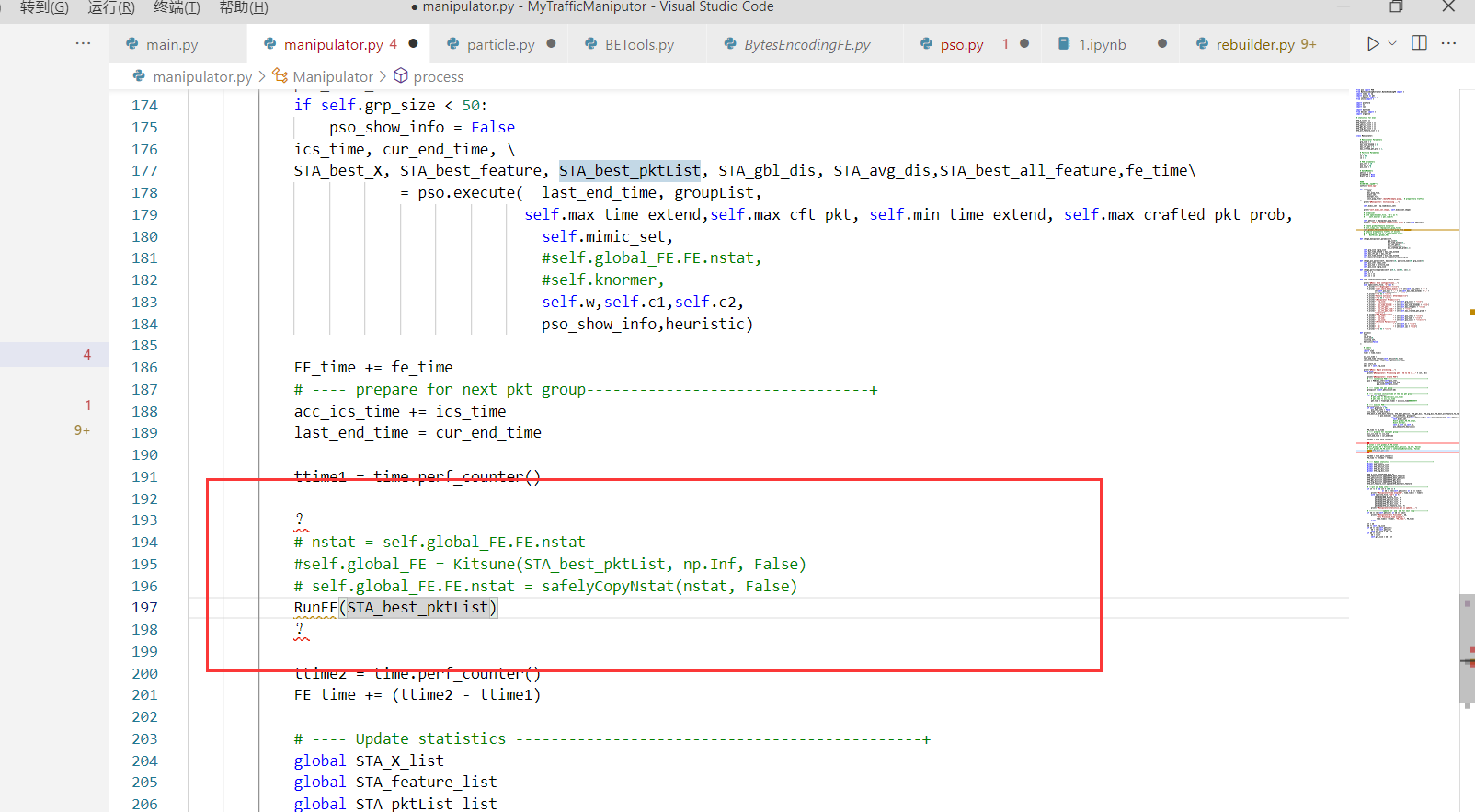
这里要改

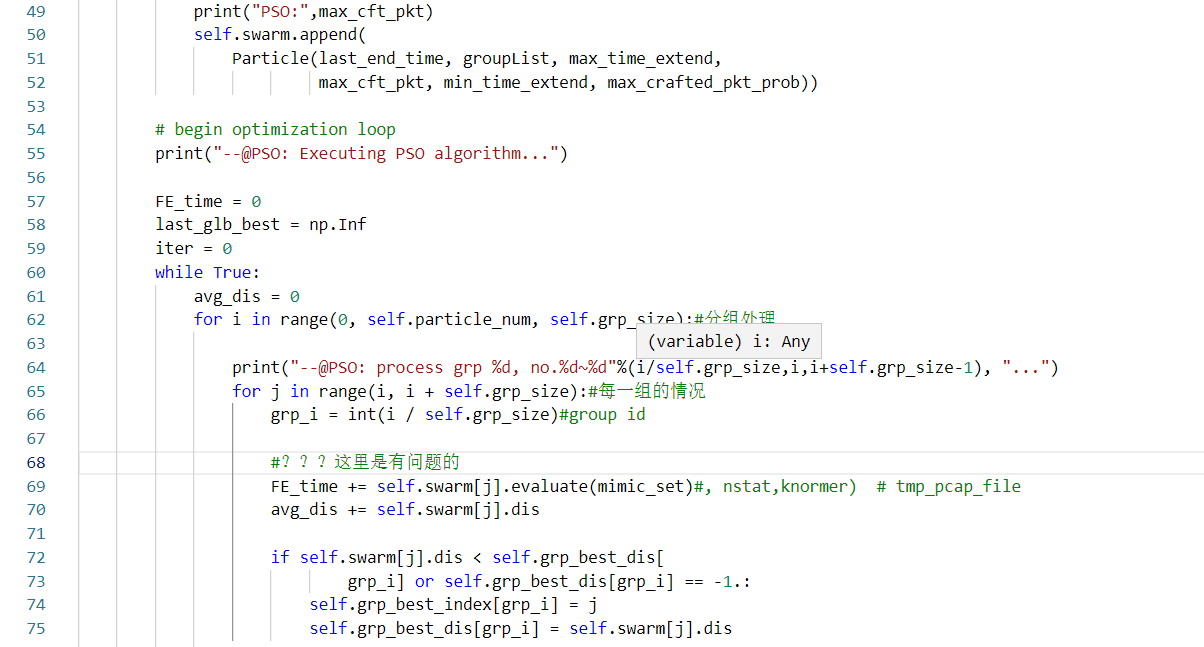
|  |
| --- |
| def rebuild(      grp\_size,      X,      groupList,      #tmp\_pcap\_file  ):      newList = []      for i in range(grp\_size):          for j in range(int(round(X.mal[i][1]))):              pkt = copy.deepcopy(groupList[i])              if round(X.craft[i][j][1]) == 1:                  if groupList[i].haslayer(Ether):                      pkt[Ether].remove\_payload()                  else:                      raise RuntimeError("Error in rebuilder!")              elif round(X.craft[i][j][1]) == 2:                  if groupList[i].haslayer(IP):                      pkt[IP].remove\_payload()                  elif groupList[i].haslayer(IPv6):                      pkt[IPv6].remove\_payload()                  elif groupList[i].haslayer(ARP):                      pkt[ARP].remove\_payload()                  else:                      raise RuntimeError("Error in rebuilder!")              elif round(X.craft[i][j][1]) == 3:                  if groupList[i].haslayer(ICMP):                      pkt[ICMP].remove\_payload()                  elif groupList[i].haslayer(TCP):                      pkt[TCP].remove\_payload()                  elif groupList[i].haslayer(UDP):                      pkt[UDP].remove\_payload()                  else:                      raise RuntimeError("Error in rebuilder!")              else:                  raise RuntimeError("Error in rebuilder!")              pkt.add\_payload(random\_bytes(int(round(X.craft[i][j][2]))))              pkt.time = X.mal[i][0] - X.craft[i][j][0]              newList.append(pkt)          mal\_pkt = copy.deepcopy(groupList[i])          mal\_pkt.time = X.mal[i][0]          newList.append(mal\_pkt)      # wrpcap(tmp\_pcap\_file, newList)      return newList |



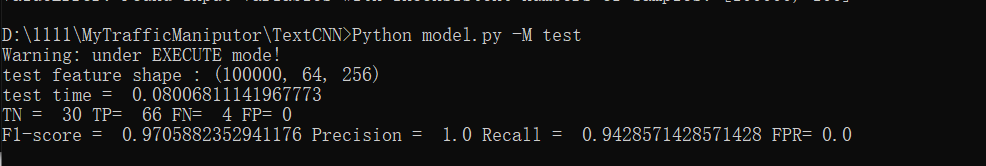
|  |
| --- |
| D:\1111\MyTrafficManiputor>python main.py -m dataPGA/test2.pcap -b dataPGA/mimic\_set.npy -i dataPGA/init.pcap  @Manipulator: Initializing ...  read 100 packets in malicious pcap  feature vectors shape : (1, 64, 256)  save successfully  @Mani: Save configurations...  <class 'int'> <class 'int'>  @Mani: Begin processing...  @Manipulator: Processing pkt ( 0 to 100 ) ...  @Manipulator: Create PSO  6 3  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 485.8573669026338  6 3  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  --@PSO:Step 1 Finished...Global best value: 485.8573669026338  6 3  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  --@PSO:Step 2 Finished...Global best value: 485.8573669026338  @Manipulator:All Finished! 100 Pkts Processed,Time elapsed: 354.7077798843384 FE\_time: 1.3028541999999383 |

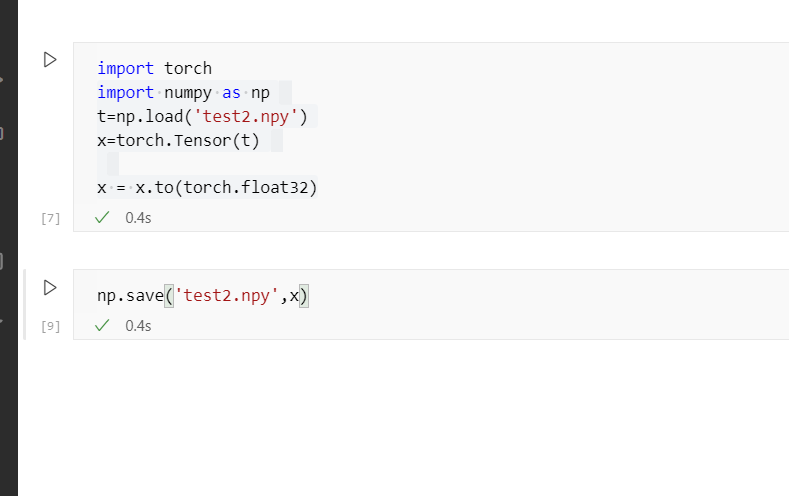






实验结果：





RuntimeError: expected scalar type Double but found Float

原因：tensor的数据类型不正确

解决： 将数据类型转为float32

并不是因为他expected Double你就要弄成Double

查了一些博文说

tensor.to(torch.float32)

但其实这里面的tensor指的是tensor类型的数据

在你自己的代码上他可能是x可能是data

这个要你自己多尝试

我的是x.to(torch.float32)

torch里有一些Bug

还有包括target后面要接.long()

分享出来希望帮到大家

————————————————

版权声明：本文为CSDN博主「不许人间见秃头」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

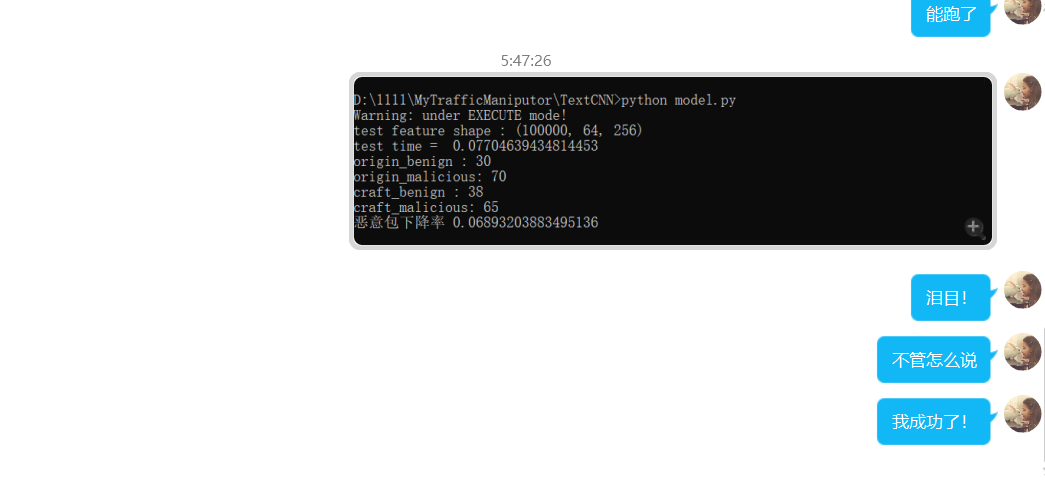
原文链接：<https://blog.csdn.net/ttdxtt/article/details/112347283>

我也用numpy转了，真尼玛坑人，两个不一样，tensor的数据格式，长的和numpy不一样

|  |
| --- |
| @Manipulator: Initializing ...  read 100 packets in malicious pcap  feature vectors shape : (1, 64, 256)  save successfully  @Mani: Save configurations...  <class 'int'> <class 'int'>  @Mani: Begin processing...  @Manipulator: Processing pkt ( 0 to 100 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 485.8573669026338  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  --@PSO:Step 1 Finished...Global best value: 485.8573669026338  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  --@PSO:Step 2 Finished...Global best value: 485.8573669026338  STA\_all\_feature\_list.shape: (103, 64, 256)  @Manipulator:All Finished! 100 Pkts Processed,Time elapsed: 105.79559588432312 FE\_time: 1.2491151000000262 |

泪目！！！

老子成功了！



|  |
| --- |
| --@PSO:Step 2 Finished...Global best value: 657.3402129797578  STA\_all\_feature\_list.shape: (10055, 64, 256)  @Manipulator:Time elapsed: 14053.927922964096  @Manipulator:statistics.pkl is updated...  @Manipulator:All Finished! 10000 Pkts Processed,Time elapsed: 14061.498632669449 FE\_time: 138.91846340000095 |

为什么用10000个效果不好呢

|  |
| --- |
| Warning: under EXECUTE mode!  test feature shape : (100000, 64, 256)  test time = 0.9862239360809326  origin\_benign : 4806  origin\_malicious: 5194  craft\_benign : 4814  craft\_malicious: 5241  恶意包下降率 -0.0018332173048235534 |

有个严重的问题，在于我怎么迭代都没有用

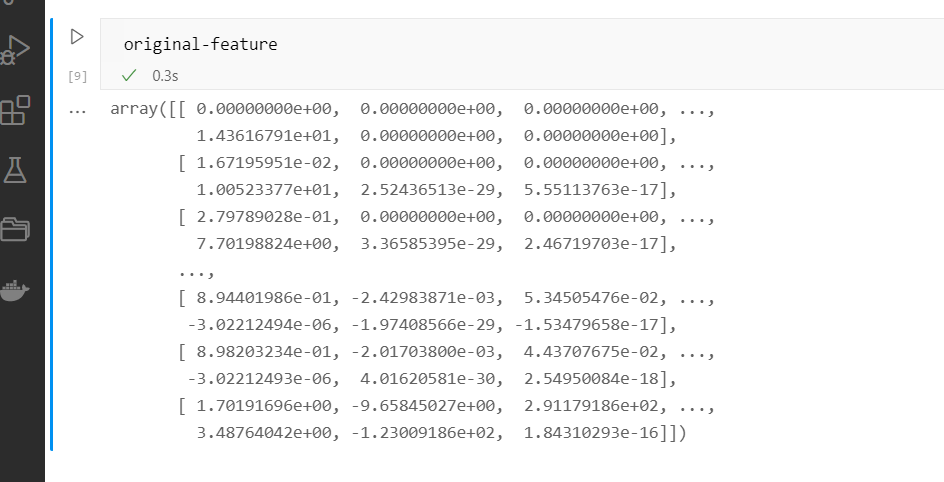
Global best value: 286.4195683208163不会改变

|  |
| --- |
| ----@Particle: Evaluate distance...  变异后流量的长度 106  原始流量长度 100  --@PSO:Step 3 Finished...Global best value: 231.52981927084232  STA\_all\_feature\_list.shape: (1019, 64, 256)  @Manipulator:Time elapsed: 3004.4800028800964  @Manipulator:statistics.pkl is updated...  @Manipulator:All Finished! 1000 Pkts Processed,Time elapsed: 3004.9838354587555 FE\_time: 26.390950299998956 |

|  |
| --- |
| Warning: under EXECUTE mode!  test feature shape : (100000, 64, 256)  test time = 0.18503451347351074  origin\_benign : 562  origin\_malicious: 438  craft\_benign : 573  craft\_malicious: 446  PDR -0.0182648401826484 |

为什么效果不好

效果不好的原因：



而我的全是0，就是除了新增的包，原始包内容是不变的？

为什么呢

 ics\_time,

cur\_end\_time, \

STA\_best\_X,

 STA\_best\_feature, STA\_best\_pktList, STA\_gbl\_dis, STA\_avg\_dis,STA\_best\_all\_feature,fe\_time\

|  |
| --- |
| D:\1111\MyTrafficManiputor>python main.py -m dataPGA/t1000.pcap -b dataPGA/mimic\_set.npy  @Manipulator: Initializing ...  read 1000 packets in malicious pcap  @Mani: Save configurations...  <class 'int'> <class 'int'>  @Mani: Begin processing...  @Manipulator: Processing pkt ( 0 to 100 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 485.8573669026338  STA\_feature\_list.shape: (100, 64, 256)  @Manipulator: Processing pkt ( 100 to 200 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 106  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 593.8675307430922  STA\_feature\_list.shape: (200, 64, 256)  @Manipulator: Processing pkt ( 200 to 300 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 108  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 47.36084085451695  STA\_feature\_list.shape: (300, 64, 256)  @Manipulator: Processing pkt ( 300 to 400 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 106  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 108  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 197.82631779265958  STA\_feature\_list.shape: (400, 64, 256)  @Manipulator: Processing pkt ( 400 to 500 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 108  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 123.25356683631604  STA\_feature\_list.shape: (500, 64, 256)  @Manipulator: Processing pkt ( 500 to 600 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 108  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 283.9442696869444  STA\_feature\_list.shape: (600, 64, 256)  @Manipulator: Processing pkt ( 600 to 700 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 286.4195683208163  STA\_feature\_list.shape: (700, 64, 256)  @Manipulator: Processing pkt ( 700 to 800 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 106  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 107  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 108  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 357.00435757113564  STA\_feature\_list.shape: (800, 64, 256)  @Manipulator: Processing pkt ( 800 to 900 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 351.6456913004583  STA\_feature\_list.shape: (900, 64, 256)  @Manipulator: Processing pkt ( 900 to 1000 ) ...  @Manipulator: Create PSO  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 103  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 102  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 100  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 105  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 101  原始流量长度 100  ----@Particle: Evaluate distance...  变异后流量的长度 104  原始流量长度 100  --@PSO:Step 0 Finished...Global best value: 231.52981927084232  STA\_feature\_list.shape: (1000, 64, 256)  @Manipulator:Time elapsed: 806.9346022605896  @Manipulator:statistics.pkl is updated...  @Manipulator:All Finished! 1000 Pkts Processed,Time elapsed: 807.45871925354 FE\_time: 8.423769400000257 |

攻击结果不怎么好，发现对于原来的包没办法降，原因应该是和字节编码不看重mtu\time\协议有关系，不太好攻击

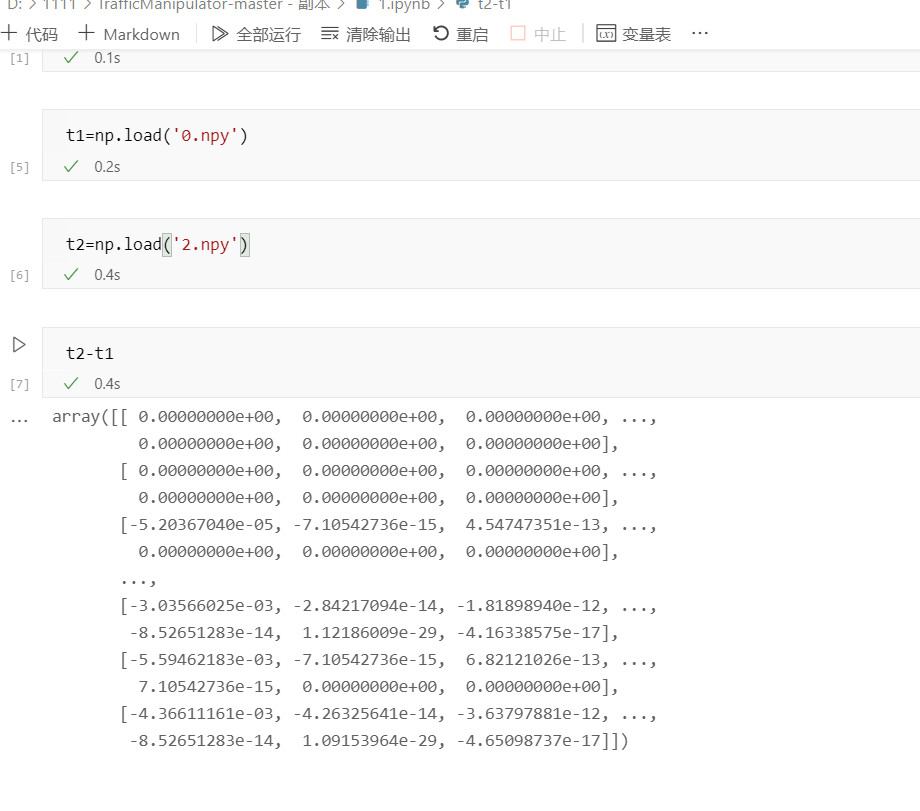
现在猜想，那三个量的变化对字节编码来说不那么明显

那么现在来看别的数据集

是不是也这样呢？

确实，为了验证猜想，我用afterImage提取了一下：

发现



还是有区别的

但是用字节编码，啥区别也没有了

结论是两个一样

dst : DestMACField = '64:f6:9d:19:6a:52' ('None')

src : SourceMACField = '88:e0:f3:7a:66:f0' ('None')

type : XShortEnumField = 2048 ('36864')

--

version : BitField (4 bits) = 4 ('4')

ihl : BitField (4 bits) = 5 ('None')

tos : XByteField = 0 ('0')

len : ShortField = 32 ('None')

id : ShortField = 64226 ('1')

flags : FlagsField = <Flag 2 (DF)> ('<Flag 0 ()>')

frag : BitField (13 bits) = 0 ('0')

ttl : ByteField = 59 ('64')

proto : ByteEnumField = 1 ('0')

chksum : XShortField = 28337 ('None')

src : SourceIPField = '203.179.179.237' ('None')

dst : DestIPField = '108.195.233.228' ('None')

options : PacketListField = [] ('[]')

--

type : ByteEnumField = 8 ('8')

code : MultiEnumField (Depends on 8) = 0 ('0')

chksum : XShortField = 16576 ('None')

id : XShortField (Cond) = 41079 ('0')

seq : XShortField (Cond) = 4568 ('0')

ts\_ori : ICMPTimeStampField (Cond) = None ('50788887')

ts\_rx : ICMPTimeStampField (Cond) = None ('50788887')

ts\_tx : ICMPTimeStampField (Cond) = None ('50788887')

gw : IPField (Cond) = None ("'0.0.0.0'")

ptr : ByteField (Cond) = None ('0')

reserved : ByteField (Cond) = None ('0')

length : ByteField (Cond) = None ('0')

addr\_mask : IPField (Cond) = None ("'0.0.0.0'")

nexthopmtu : ShortField (Cond) = None ('0')

unused : MultipleTypeField (ShortField, IntField, StrFixedLenField) = b'' ("b''")