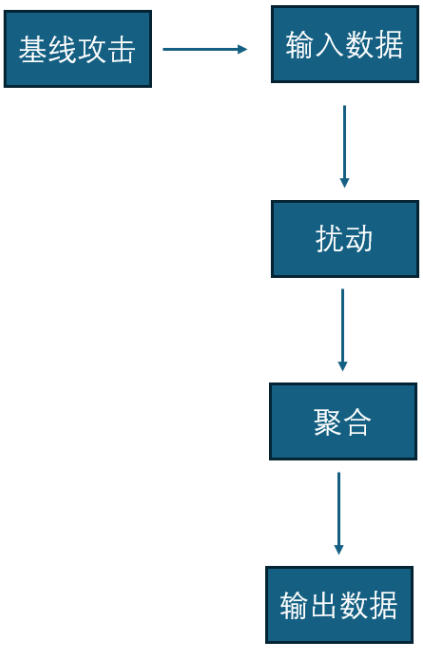
1. 复现 pirveNUD 协议
2. 复现 AHEAD 协议
3. 实现攻击方式

基线攻击



RPA 的攻击方式：直接随机生成范围内的数值

```
# 每一个假用户随机选择一个值攻击，假用户和假值为一对一关系，person值为假用户数
def attack_RPA(person):
    f = open(data_path_old)
    data_old = []
    for line in f:
        data_old.append(line.strip())
    f.close()
    f = open(data_path_new, "a")
    for m in range(person):
        num = choice(data_old)
        f.write(str(num)+'\n')
    f.close()
    print("RPA攻击成功，已经注入"+str(person)+"个假用户")
```

总的来说就是，找到用户库，获取用户值，得到合理的用户域范围，再随机注入数据

RIA 的部分代码:

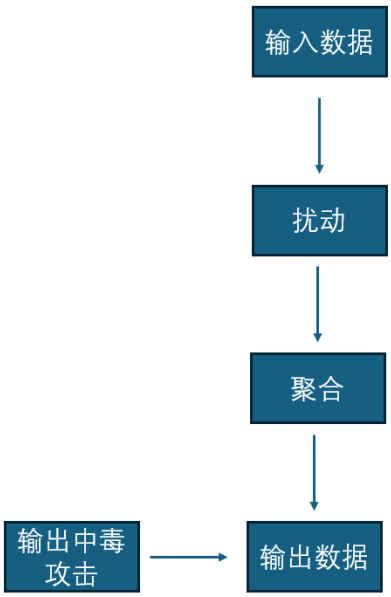
```
# 将用户域分组，并重复和攻击一组的数据，person为假用户值，分组的条件和思想可以自己看着办，可以研究什么时候是最有效的
1个用法
def attack_RIA(person):
    f = open(data_path_old)
    data_old = []
    for line in f:
        data_old.append(line.strip())
    f.close()

    # 总步长
    long = check_data(data_path_old)
    # 组数
    group_num = 4

    # print(long)
    # 随机选择组
    group_index = random.randint( a: 1,  b: 4)
    # print(group_index)
```

总的来说就是比 RIA 多了，一个分组的思想，但是用户组是没有分组的，所以需要自行分组，分组的范围也可以由攻击者自行裁定

输出中毒攻击（OPA）:



部分代码:

```python
errList = np.zeros(len(query_interval_table))

# 求MSE
var_list_consistency = []
for i, query_interval in enumerate(query_interval_table):
    d1_left = int(query_interval[0])
    d1_right = int(query_interval[1])
    real_frequency_value = real_frequency[i]
    estimated_frequency_value = ahead_tree_answer_query(ahead_tree, query_interval, domain_size)

    # 注入假用户
    if fake_user > 0:
        estimated_frequency_value = estimated_frequency_value + 0.1
        fake_user = fake_user - 1

    errList[i] = real_frequency_value - estimated_frequency_value
    print('answer index {}-th quey'.format(i))
    print("real_frequency_value: ", real_frequency_value)
    print("estimated_frequency_value: ", estimated_frequency_value)

    var_consistency = math.pow(real_frequency_value - estimated_frequency_value, _y: 2)
    var_list_consistency.append(var_consistency)

f = np.array(var_list_consistency)

# 求均值
print('mean_var_consistency:', np.mean(f))

# MSEDict['rand'].append(errormetric.MSE_metric(errList))
```

对输出后的数据进行数据中毒攻击，攻击的程度可以自行定夺