数据安全 -- 交互式发布DP方案评估

> **学号：2212452**
> **姓名：孟启轩**
> **专业：计算机科学与技术**

一、实验要求

参考教材实验5.1，对交互式发布方案进行DP方案设计，指定隐私预算为0.1，支持查询次数为20次，对DP发布后的结果进行评估说明隐私保护的效果。

二、实验内容

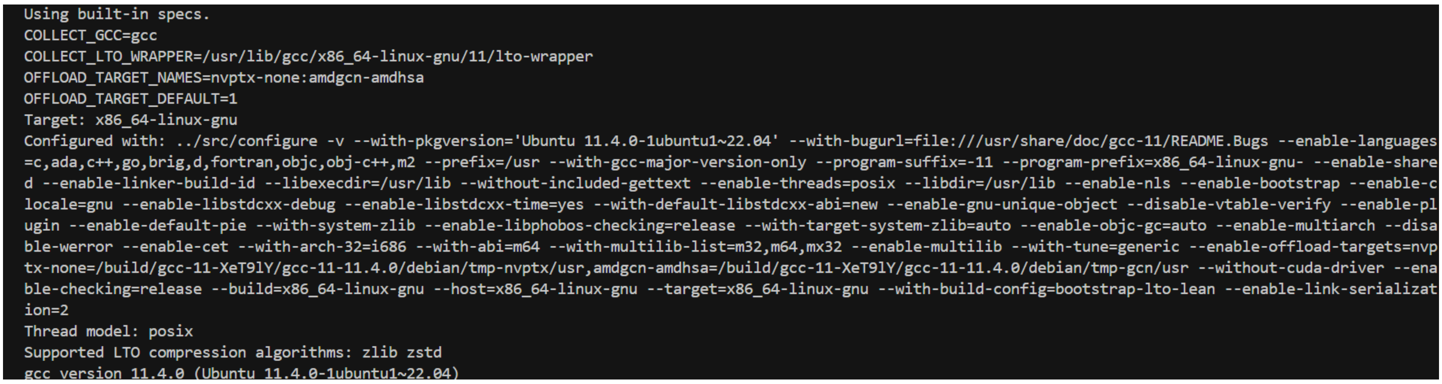
对一个数据集 `zoo.csv` 进行统计查询，该数据集描述了一个动物园喂食的场景，第一列中数据为动物名称，第二列中数据为动物每天消耗的胡萝卜数量。查询定义为`每日进食超过55根胡萝卜的动物数量`。根据所给的代码编译出简单的差分隐私拉普拉斯机制噪音产生和加噪程序和直方图加噪发布程序，使用给定的数据集，通过变换输入的隐私预算来观察不同隐私预算下的噪音规模和对数据的影响。

三、实验过程

1、前期准备

(一)安装必要的解释器

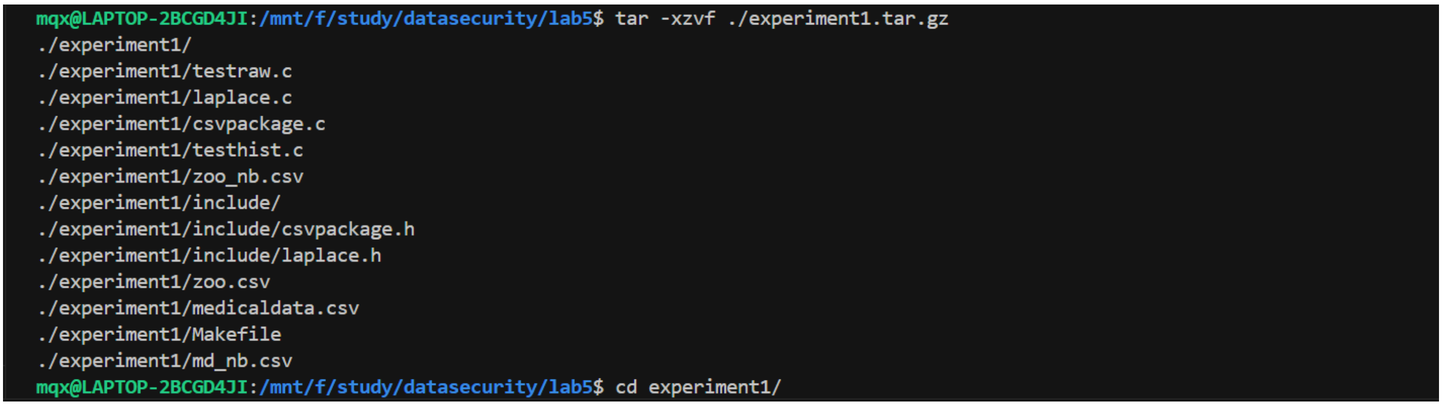
我之前已经安装过，这里直接执行指令`gcc -v`，查看gcc是否安装成功：



```
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/11/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none:amdgcn-amdhsa
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ../src/configure -v --with-pkgversion='Ubuntu 11.4.0-1ubuntu1~22.04' --with-bugurl=file:///usr/share/doc/gcc-11/README.Bugs --enable-languages
=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --prefix=/usr --with-gcc-major-version-only --program-suffix=-11 --program-prefix=x86_64-linux-gnu- --enable-share
d --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --enable-bootstrap --enable-c
locale=gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --with-default-libstdcxx-abi=new --enable-gnu-unique-object --disable-vtable-verify --enable-pl
ugin --enable-default-pie --with-system-zlib --enable-libphobos-checking=release --with-target-system-zlib=auto --enable-objc-gc=auto --enable-multiarch --disa
ble-werror --enable-cet --with-arch-32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvp
tx-none=/build/gcc-11-XeT9lY/gcc-11-11.4.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-11-XeT9lY/gcc-11-11.4.0/debian/tmp-gcn/usr --without-cuda-driver --ena
ble-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu --with-build-config=bootstrap-lto-lean --enable-link-serializat
ion=2
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 11.4.0 (Ubuntu 11.4.0-1ubuntu1~22.04)
```

(二)解压文件

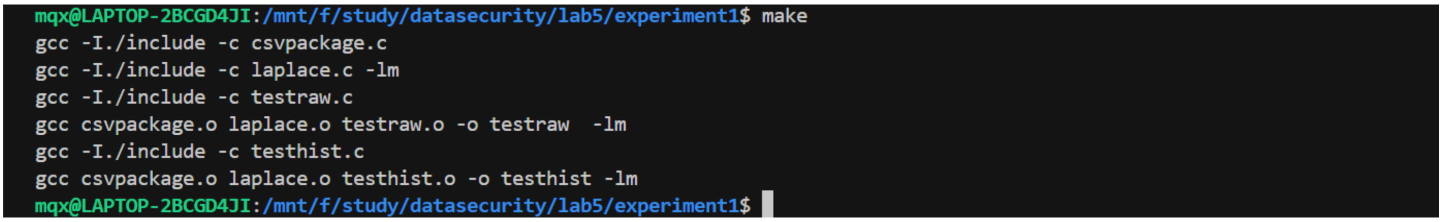
使用`tar -xzvf ./experiment1.tar.gz`指令解压提供的test.tar.gz压缩包：



```
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5$ tar -xzvf ./experiment1.tar.gz
./experiment1/
./experiment1/testraw.c
./experiment1/laplace.c
./experiment1/csvpackage.c
./experiment1/testhist.c
./experiment1/zoo_nb.csv
./experiment1/include/
./experiment1/include/csvpackage.h
./experiment1/include/laplace.h
./experiment1/zoo.csv
./experiment1/medicaldata.csv
./experiment1/Makefile
./experiment1/md_nb.csv
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5$ cd experiment1/
```

(三)编译

进入实验文件夹，执行make进行编译：



```
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5/experiment1$ make
gcc -I./include -c csvpackage.c
gcc -I./include -c laplace.c -lm
gcc -I./include -c testraw.c
gcc csvpackage.o laplace.o testraw.o -o testraw  -lm
gcc -I./include -c testhist.c
gcc csvpackage.o laplace.o testhist.o -o testhist -lm
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5/experiment1$
```

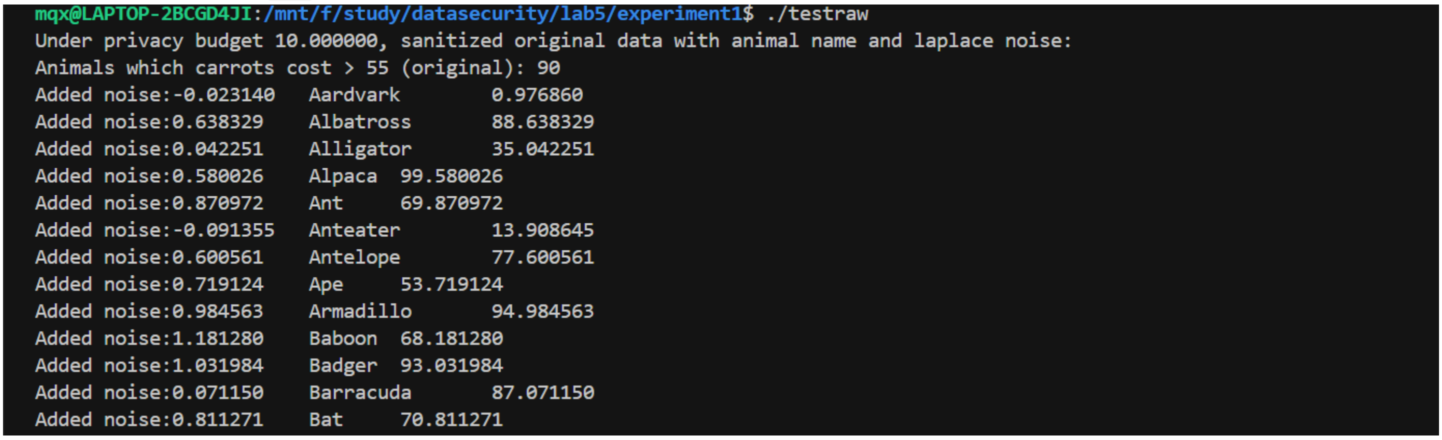
2、程序运行

根据实验指导书，用隐私预算`10`和`0.1`来分别运行程序`testraw`和`testhist`。

(一)testraw

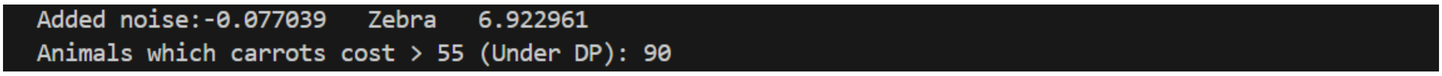
> 运行testraw程序，观察生成的噪音和加噪后的数据与原始数据的差别。

运行testraw，投入`10`的隐私预算：



```
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5/experiment1$ ./testraw
Under privacy budget 10.000000, sanitized original data with animal name and laplace noise:
Animals which carrots cost > 55 (original): 90
Added noise:-0.023140   Aardvark        0.976860
Added noise:0.638329    Albatross       88.638329
Added noise:0.042251    Alligator       35.042251
Added noise:0.580026    Alpaca  99.580026
Added noise:0.870972    Ant     69.870972
Added noise:-0.091355   Anteater        13.908645
Added noise:0.600561    Antelope        77.600561
Added noise:0.719124    Ape     53.719124
Added noise:0.984563    Armadillo       94.984563
Added noise:1.181280    Baboon  68.181280
Added noise:1.031984    Badger  93.031984
Added noise:0.071150    Barracuda       87.071150
Added noise:0.811271    Bat     70.811271
```

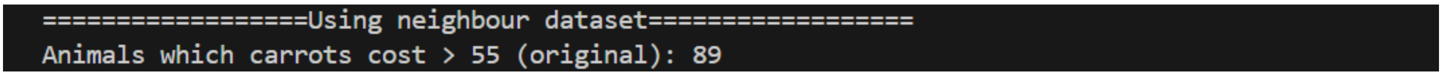
在较大的隐私预算下，添加的噪音主要集中在1左右。对于特定查询`每日进食大于55根胡萝卜的动物个数`，在该预算设置下，加噪前后的查询结果一致，表明数据的可用性良好：



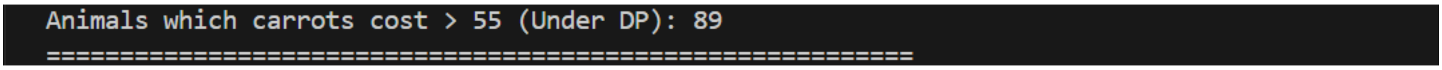
```
Added noise:-0.077039   Zebra   6.922961
Animals which carrots cost > 55 (Under DP): 90
```

然而，通过观察相邻数据集的处理情况，我们可以发现，即使在加噪后，数据集对该查询的响应仍然与数据集的变化保持一致，均为`89`。这表明`Dugeng`离开数据集引起的差异仍然被体现出来，说明该方

法不能有效抵御对该查询的差分攻击。

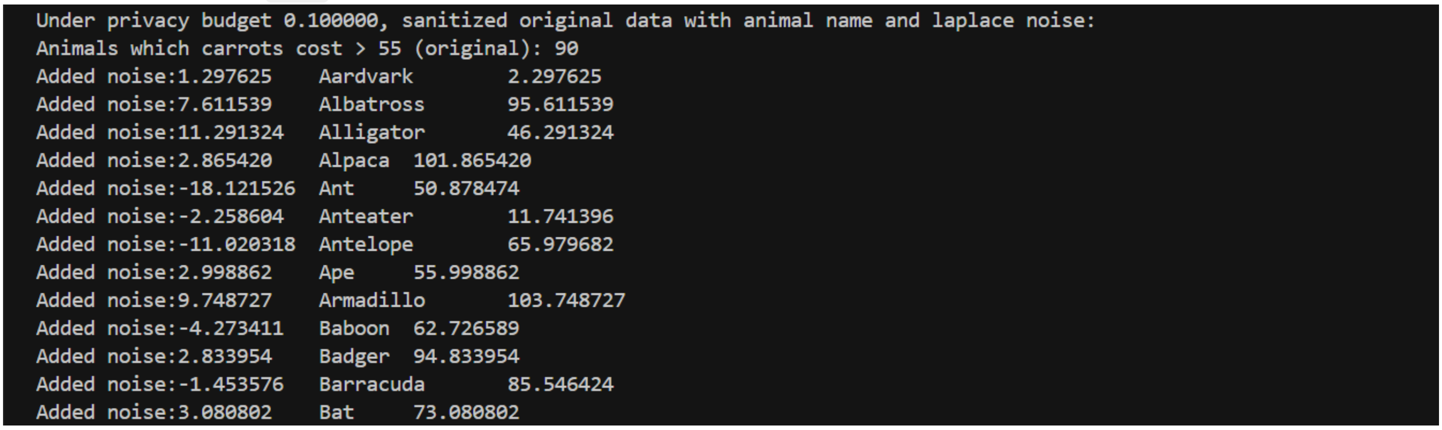


```
=================Using neighbour dataset=================
Animals which carrots cost > 55 (original): 89
```



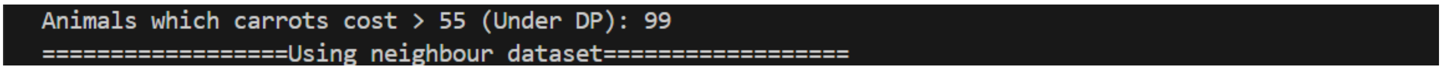
```
Animals which carrots cost > 55 (Under DP): 89
==========================================================
```

运行testraw，投入`0.1`的隐私预算：



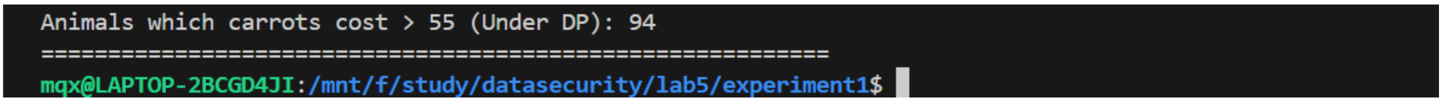
```
Under privacy budget 0.100000, sanitized original data with animal name and laplace noise:
Animals which carrots cost > 55 (original): 90
Added noise:1.297625    Aardvark        2.297625
Added noise:7.611539    Albatross       95.611539
Added noise:11.291324   Alligator       46.291324
Added noise:2.865420    Alpaca  101.865420
Added noise:-18.121526  Ant     50.878474
Added noise:-2.258604   Anteater        11.741396
Added noise:-11.020318  Antelope        65.979682
Added noise:2.998862    Ape     55.998862
Added noise:9.748727    Armadillo       103.748727
Added noise:-4.273411   Baboon  62.726589
Added noise:2.833954    Badger  94.833954
Added noise:-1.453576   Barracuda       85.546424
Added noise:3.080802    Bat     73.080802
```

可以看到，在该预算下，生成的拉普拉斯噪音显著增大，这导致加噪后的查询结果也受到了较大的影响：



```
Animals which carrots cost > 55 (Under DP): 99
=================Using neighbour dataset=================
```

但是，观察对相邻数据集进行加噪的结果，可以发现尽管相邻数据集的直接查询结果受到了`Dugeng`项移除的影响，但加噪后的相邻数据集查询结果与加噪前相比发生了显著变化，不再能反映出`Dugeng`项移除的影响。



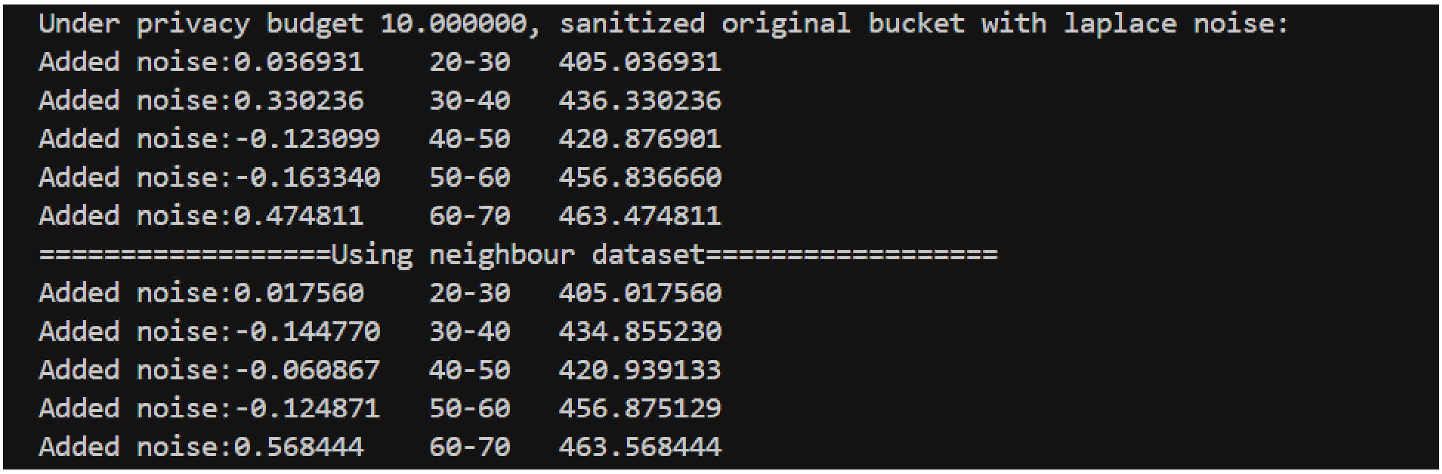
```
Animals which carrots cost > 55 (Under DP): 94
==========================================================
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5/experiment1$
```

可以看到投入较少的隐私预算时，虽然数据的可用性降低了，但是能够更好地抵御差分攻击的影响，在使用`0.1`作为隐私预算时，可以抵御差分攻击影响。

(二)testhist

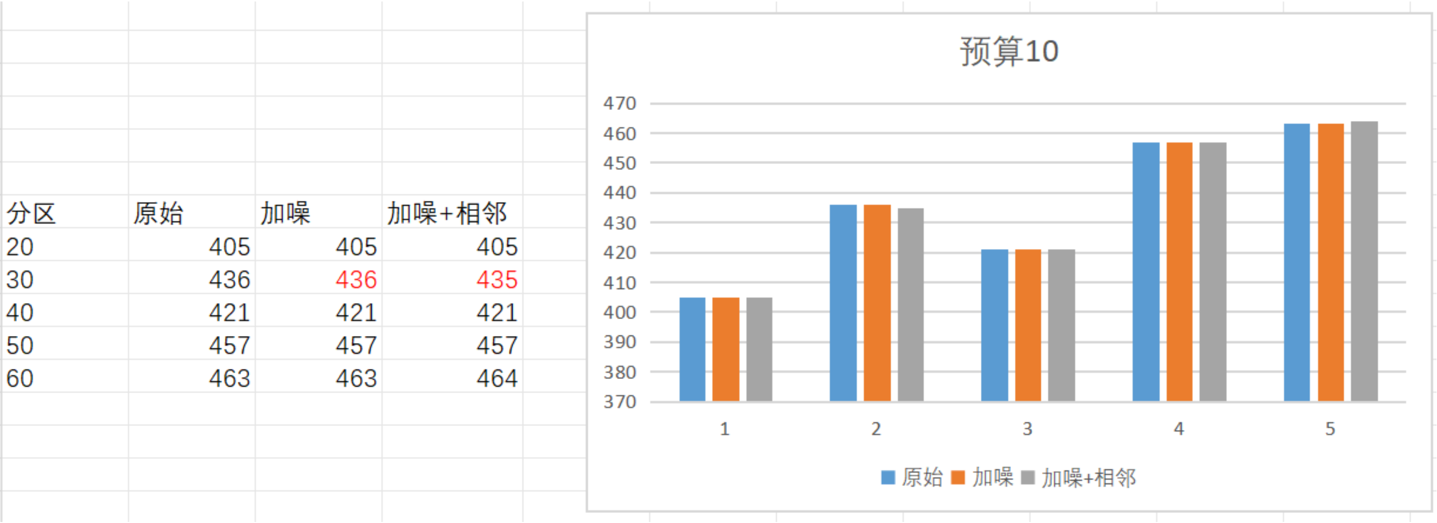
`testhist`程序提供了另一种差分隐私发布方法的演示，即差分隐私的直方图发布。在该方法下，加噪的对象是对数据进行分桶统计后的计数值，而非数据本身。

运行testhist程序，投入`10`的隐私预算：



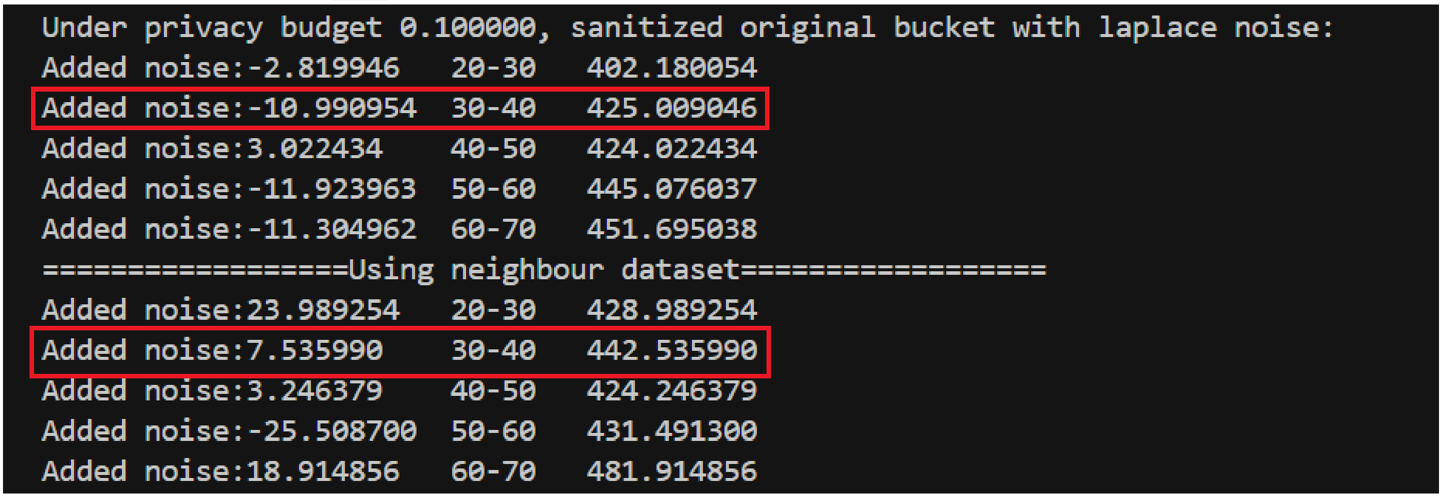
```
Under privacy budget 10.000000, sanitized original bucket with laplace noise:
Added noise:0.036931     20-30    405.036931
Added noise:0.330236     30-40    436.330236
Added noise:-0.123099    40-50    420.876901
Added noise:-0.163340    50-60    456.836660
Added noise:0.474811     60-70    463.474811
==================Using neighbour dataset==================
Added noise:0.017560     20-30    405.017560
Added noise:-0.144770    30-40    434.855230
Added noise:-0.060867    40-50    420.939133
Added noise:-0.124871    50-60    456.875129
Added noise:0.568444     60-70    463.568444
```

可以看到，当隐私预算为`10`时，由于加入的噪音量较小，相邻数据集的变化仍然能被体现出来。使用预算为`10`的差分隐私算法生成的数据和直方图如下：



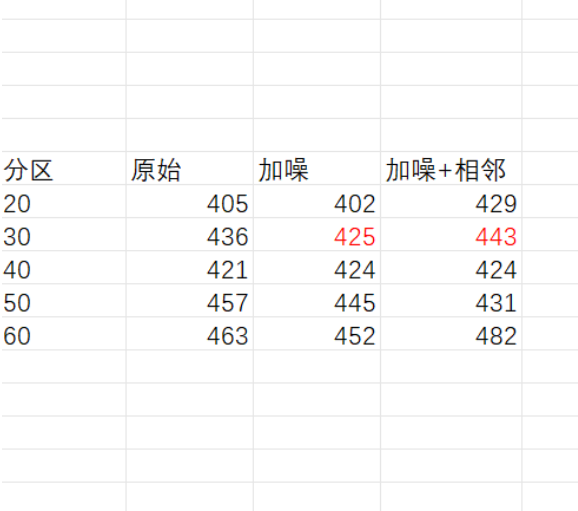
分区	原始	加噪	加噪+相邻
20	405	405	405
30	436	436	435
40	421	421	421
50	457	457	457
60	463	463	464

运行testhist程序，投入`0.1`的隐私预算：

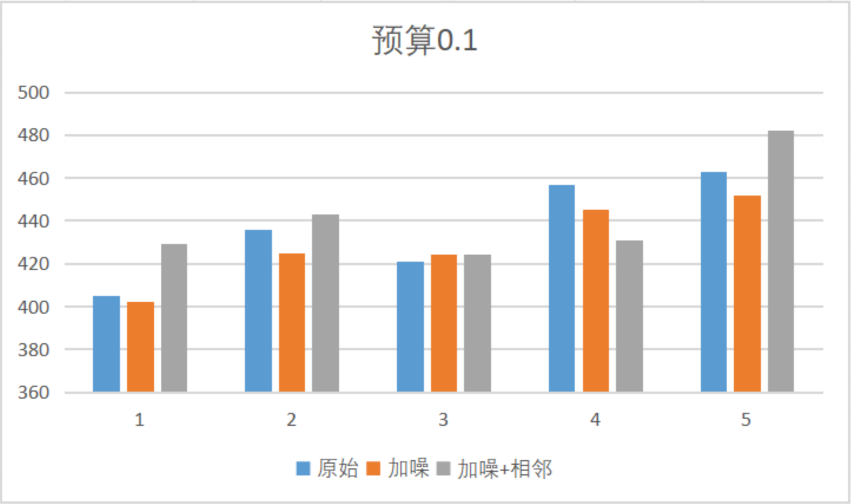


```
Under privacy budget 0.100000, sanitized original bucket with laplace noise:
Added noise:-2.819946     20-30    402.180054
Added noise:-10.990954    30-40    425.009046
Added noise:3.022434      40-50    424.022434
Added noise:-11.923963    50-60    445.076037
Added noise:-11.304962    60-70    451.695038
==================Using neighbour dataset==================
Added noise:23.989254     20-30    428.989254
Added noise:7.535990      30-40    442.535990
Added noise:3.246379      40-50    424.246379
Added noise:-25.508700    50-60    431.491300
Added noise:18.914856     60-70    481.914856
```

可以看到，随着噪音规模的提高，相邻数据集的变化影响导致查询结果不减反增。虽然数据的可用性变差，但这种方法能够保护实际数据的变化不被攻击者获取，从而抵御差分攻击。使用预算0.1的差分隐私算法生成的数据和直方图如下：

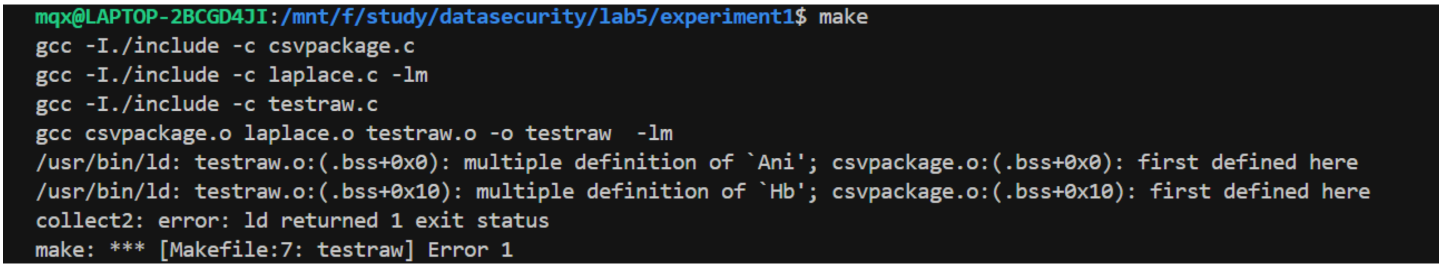


分区	原始	加噪	加噪+相邻
20	405	402	429
30	436	425	443
40	421	424	424
50	457	445	431
60	463	452	482



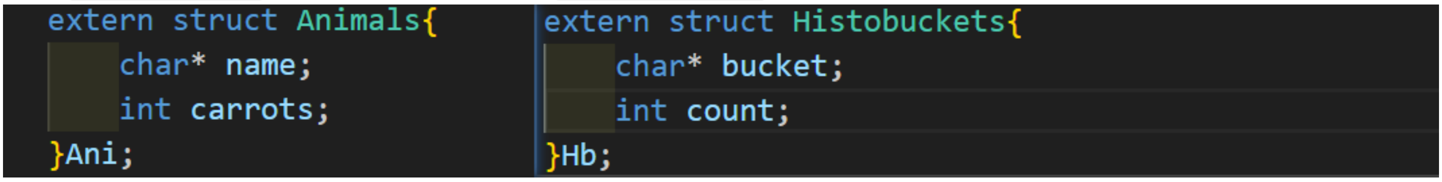
四、遇到的问题

执行make进行编译时，遇到重定义的问题：



```
mqx@LAPTOP-2BCGD4JI:/mnt/f/study/datasecurity/lab5/experiment1$ make
gcc -I./include -c csvpackage.c
gcc -I./include -c laplace.c -lm
gcc -I./include -c testraw.c
gcc csvpackage.o laplace.o testraw.o -o testraw  -lm
/usr/bin/ld: testraw.o:(.bss+0x0): multiple definition of `Ani'; csvpackage.o:(.bss+0x0): first defined here
/usr/bin/ld: testraw.o:(.bss+0x10): multiple definition of `Hb'; csvpackage.o:(.bss+0x10): first defined here
collect2: error: ld returned 1 exit status
make: *** [Makefile:7: testraw] Error 1
```

我将`csvpackage.h`中的两个结构体改为`extern struct`，解决了这个问题：



```
extern struct Animals{
    char* name;
    int carrots;
}Ani;
```

```
extern struct Histobuckets{
    char* bucket;
    int count;
}Hb;
```

五、心得体会

通过本次实验，我深入理解了差分隐私中拉普拉斯机制核心原理及其实际应用效果。实验通过调整隐私预算的大小，直观展示了隐私保护强度与数据可用性之间的权衡关系：当隐私预算较大时，噪声较小，数据查询结果与原始数据高度一致，但无法有效抵御差分攻击；而当隐私预算较小时，噪声显著增大，虽然数据可用性下降，但能够有效掩盖相邻数据集的微小差异，从而增强隐私安全性。此外，直方图进一步验证了在统计值加噪场景下，差分隐私如何通过分桶计数加噪保护群体数据隐私。实验过程不仅让我掌握了差分隐私机制的实现细节，也深刻认识到隐私预算分配策略在实际应用中的关键作用。