Victor Mora

Malte Schwarzkopf

CSCI 2390: Privacy-Conscious Computer Systems

29 October 2021

Differential Privacy Assignment

<u>Question 1</u>: Look at the result of this count query. Note that it does not include any name, email, or other personally identifiable information. What can you nevertheless learn about the TA's musical tastes? What possible genres might they have chosen? Alternatively, what genres is it impossible for them to have chosen?

Due to the assumption that the TA did not respond with 0 or "Prefer not to answer", I know that Kinan is neither of those individuals. Kinan's favorite is likely not Hip Hop either, as I know Kinan is older than me and most of the Hip Hops are concentrated at ages below mine. It is likely that Kinan is a Rock or Metal person who is either 27 or 34.

<u>Question 2</u>: What did you find out about the TA? Are your findings consistent with Question 1? Combine the two together to learn the TA's exact age.

On LinkedIn, Kinan identifies that he likes Metal and Progressive Rock, which is consistent with Question 1. I think Kinan's exact age is 27.

<u>Question 3</u>: Identify the TA's favorite color. What is it? How easy or obvious is this to do, and why?

The TA's favorite color is black. It was easy to determine this because Kinan is the only 27 year old and the 27 year old's favorite color is black.

Question 4: What information can you learn about the TA's favorite sport from the above query?

From the query, I know that Kinan's favorite sport is either Baseball, E-sports, or Hockey.

Question 5: What is our TA's favorite sport?

Kinan's favorite sport is E-sports, since that is the only overlapping sport between the two datasets for his age range.

Question 6: Run dp.py several times varying the epsilon privacy parameter for different values between 10 and 0.01, like so:

$ python3 dp.py 0.01

[...]

$ python3 dp.py 0.1

[...]

[etc.]

What happens when the privacy parameter grows larger or smaller? How does that affect privacy?

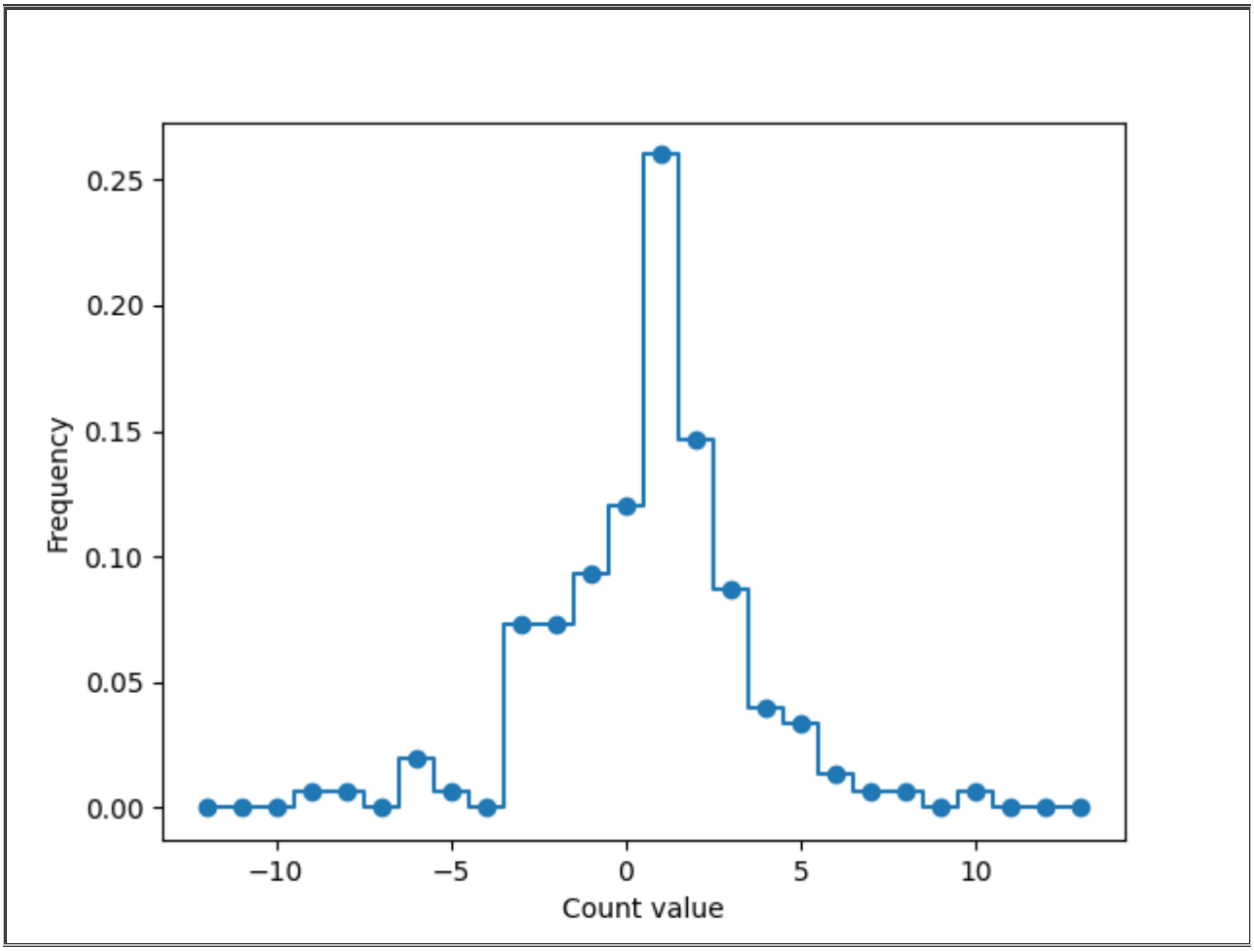
As the privacy parameter grows smaller, the privacy increases (the counts are more obscured from their true values). A larger epsilon reduces privacy (values are closer to true values).

Question 7: Look at the plot generated with privacy parameter epsilon = 0.5. What is the most likely value? What is the expected (i.e., average) value? How do they relate to the

actual value (i.e., the query executed without any noise via client.py)? How does the plot change for different values of the privacy parameter?

Please include the generated plot for epsilon = 0.5 in your submission.

The most likely value is 1, which is the true value and the expected value. For lower values of the privacy parameter, the counts are more spread out (more distinct counts), whereas for higher values of the privacy parameter there is more clustering by the true value.



Question 8: Run the composition attack against the average age grouped by programming experience. What can you deduce from the exposed averages about the programming

experience level of our TA? How confident are you in what you have deduced? Are there scenarios where they might be wrong?

I can deduce that the 34 year old in the class is in the “More than 10 years” group, and that they are not alone in that group. It is likely that the 27 or 28 year old is also in that group. Kinan is likely in this group. I am not entirely confident though. A scenario where I could be wrong is if the 28 year old and 34 year old are in the “More than 10 years” group, as the average age would be 31. If it was the 27 and 34 year olds in the same group, the average age would still be 30.5 ~ 31 years old. Also, it is possible that the 27 or 28 year old being in a group with a lot of younger students could dilute the average age, making it harder to tell if they are in that group.

Question 9: Reuse your composition attack from question 8 to compute the exact non-noised counts per programming experience level. Deduce the programming experience level of our TA, with high confidence, by looking at both the exposed counts and the previously exposed averages. Now summarize everything you've learned about the TA!

Kinan is in the “More than 10 years” group along with the 34 year old, as the “8-10 years” group only has 1 person in it with an average age of 22. I also know that Kinan is 27 and likes metal music, e-sports, and the color black.

Question 10: Does the class you implemented suffice to truly enforce that a dataset is never used beyond a certain privacy budget? Can developers intentionally or unintentionally over-use the dataset beyond the privacy budget? At a very high level, how would you design a different privacy budget enforcment mechanism that does not suffer these drawbacks?

The class I implemented does not suffice. A number of developers colluding together could achieve the effect of raising the privacy budget (5 developers each with a privacy budget of 2 could query 20 times collectively, effectively raising the privacy parameter to 10. I think one potential solution could be to design the system in such a way that any queries (from any developer) above the privacy budget would just recycle previously returned values. In this design, the privacy budget would have to reside within the system itself, so that it would know how many times it has been queried by any client, and so all individuals would share a collective privacy budget.