CS3110 Project Report

Jack Giuliani and Alex Shapovalov

12/09/2024

GitHub link

https://github.com/alex-shapovalov/CS-3110-FP

Problem statement

Using the topics we learned in class, how difficult is it to privatize sensitive patient data through a variety of methods.

Description of the project

Our project uses data from "SyntheticHealth" which is an organization that creates mock patient data for ML use. We used a variety of methods to analyze this data, including applying Laplace and Gaussian noise, k-anonymizing data, and clipping. We also tested the Propose-Test-Release and Sample and Aggregate frameworks as methods to find average and maximum values for some numeric categories while still achieving differential privacy. Overall we attempted to use all or most of the methods we learned in class in some fashion on our data.

Results of the project

We learned more about the difference between Laplace and Gaussian noise and the difficulty of k-anonymizing data. We found that, again, Laplace noise is the better option for differential privacy. For clipping, we tested both a linear range of values and an exponentially increasing range of values to clip at. Since the values for healthcare expenses differed so much, ranging from around 1,000 to around 20,000,000, we found that it was difficult to find a good clipping value while still achieving differential privacy. Because the range for the linear values was so large, the amount of noise that was added sometimes made the clipping values we found unusable, as they fluctuated greatly and had no real utility. We also found that finding clipping values using exponentially increasing values was difficult, as the distances between values was very large. However, less noise was added to these results, which made them a little more useful. Ultimately, we decided on using a clipping value of 5,000,000, as that value covered most of the data apart from a few outliers.

To perform a linkage attack, we first created the dataset data_pii, which was created using a subset of columns from our dataset: first name, last name, birthdate, city, state, and zip code. We then created the dataset data_deid, which was made by dropping all columns that could be used to uniquely identify someone. We then tried to link these two datasets with columns that they had in common, and found that there were many individuals that could be recovered. If any of these individuals in the data could be uniquely identified by something like a zip code, then this could pose some privacy concerns.

We found that both the Propose-Test-Release and Sample and Aggregate frameworks worked very well, though there were some problems for finding a max value when using them. For finding an average value for healthcare expenses, both PTR and SAA were very effective, each having a difference of about 2%, a very similar result to finding an average by using clipping. However, when trying to find a max value for the column, both of these frameworks were not very effective. This is because we clipped the values of the outliers in the columns, which were the highest values in the column. This caused a very high difference, around 80% or higher, but likely would have been better if we had chosen a more effective clipping value.