CIS 4230/5230 Homework 3: Membership Inference Attack

Prof Michael Kearns

Spring 2024

Due Date: TBD, but towards the end of reading period. Every student should work alone and submit their own notebook.

Last updated: April 24, 2024 10AM (will be refined/added to so please make sure you have latest version)

In this assignment, you are asked to implement and measure the success of a so-called Membership Inference Attack (MIA) on your Homework 2 implementation of the Laplace Mechanism (LM).

In an MIA, an attacker is provided with the following information: a dataset x; a dataset x' that is identical to x except that it is missing one element of x (so if x is of length n, x' is of length n-1); and a function f() that will be applied either to x or x'. More precisely, the attacker is given as input x and x', and a value y that is the output of the LM on *either* f(x) or f(x'). In other words, y is either f(x) plus Laplace noise, or f(x') plus Laplace noise. As in Homework 2, the Laplace noise added in either case will depend on the privacy parameter epsilon, and the sensitivity of the function f() (which in turn may depend on whether the input is of length n or n-1). The goal of the MIA is to use x, x' and y to guess/determine whether the input to the LM was f(x) or f(x').

You should design and write a subroutine MIA(f,x,x',y) that outputs a guess of the form "input was x" or "input was x' " (or a binary value encoding these two guesses). You will then write a cell that, for each of many choices for epsilon, will do the following:

- Pick a uniformly random entry of x to delete in order to obtain x'
- Call your LM subroutine to compute the output y of the LM on inputs f(), epsilon and either x or x' (chosen randomly with probability ½ each)
- Call MIA(f,x,x',y)
- Record whether MIA(f,x,x',y) was correct in its guess of x or x' as the input to LM

It's possible you'll need to refactor your code for LM from Homework 2 depending on how you implemented it.

For many trials of the above process (which involves several sources of randomization, namely the choice of a random element to delete, the choice of x or x', and the randomization of LM), you should record the success rate of MIA, and plot it versus epsilon for multiple choices of the length of x, for multiple functions. More precisely, using the same dataset as for Homework 2, for each of the 9 columns, and for each of the same functions as in Homework 2, and for each of many prefixes of each column as in Homework 2, and for each of many choices for epsilon, you should run the process enough times to get the MIA success rate (basically you want to do it enough times to make your plots relatively smooth). This success rate should always be a number between 0 and 1, with 1.0 indicating perfect success in guessing x or x', and 0.5 indicating no better than random guessing. For each column, function pair you should have a single plot showing the success rate vs. epsilon, with a separate curve for each value of n, so 45 plots total.

You are free to design MIA any way you like, but obviously it should input absolutely no information other than f, x, x' and y. A natural/obvious approach is to have MIA compute f(x) and f(x') and see whether y is closer to f(x) or f(x'), but you're encouraged to see if you can come up with something

better.

Here's pseudocode for the rough loop I have in mind:

For each column

For each prefix x of the column

For each of the five functions f()

For each value of epsilon

For enough times to get smooth curves:

- Pick a random element of x to remove, call the result x'
- Let z be either x or x' with equal probability
- Let y ← LM(f,z,epsilon)
- Call MIA(f,x,x',y) and see whether it guesses that z = x or z = x'
- Record whether guess was correct or not

Then for each of the 9 columns x 5 functions = 45 plots, on that plot you should show multiple curves plotting epsilon (x axis) vs. success rate of MIA (y axis), each curve corresponding to a different prefix length (n).