EECS700 Data Privacy and Security

Midterm Exam

Time:10/19/2023--10/20/2023（11:59 PM）

Question 1 (25 Points). The notion of k-anonymity over tabular datasets (e.g., the medical data with quasi-identifiers and sensitive attributes) has been well defined. Generalization and suppression can be executed to achieve k-anonymity for the datasets.

(a) What are the differences between the quasi-identifiers (QIs) and the sensitive attributes? (4 Points)

(b) We generalize QIs to achieve k-anonymity and modify sensitive values to achieve l-diversity. Do you agree with the above statement? Justify your answer. (5 Points)

(c) Apply the k-anonymity notion to social network data (graph structure). For instance, every node represents an individual person while every edge represents a relationship between two people. How to define k-anonymity in such context? (4 Points)

(d) Design a simple algorithm to ensure k-anonymity for the social network data. (8 Points)

(e) Can the adversaries apply the attacks on the k-anonymous tabular datasets (e.g., homogeneity attacks, skewness attacks) to the k-anonymous social network data? Justify your answer. (4 Points)

Question 2 (18 Points). Suppose a medical center would like to permit queries on the following relation (table) that ask for average weight of the patients without disclosing the weight of particular patients:

Name	*Gender*	*Weight (lbs)*
Allen	M	145
Baker	F	130
Cook	M	187
Gupta	M	152
Hall	F	119

(a) Design a differentially private query/mechanism for returning the average weight of patients, and discuss the required parameters (you can define any reasonable parameter). (5 points)

(b) Try to choose two different sensitivities (e.g., adding or removing any patient, and adding or removing any two patients) for the

differentially private average query, discuss the corresponding differentially private query results and the meanings of two different sensitivities as well as how they affect the privacy guarantee. (5 points)

(c) Design a differentially private query/mechanism that returns the gender with more patients. (8 points)

Question 3 (15 points).

(a) Given five parties (Alice, Bob, Chris, David, Edward), each of them can have up to two choices to select their preferred movies (out of four movies M1, M2, M3, M4). Design an LDP protocol to privately collect each party's preferences and estimate the top selected movie. (10 points)

(b) What are the differences between differential privacy and local differential privacy? (5 points)

Question 4 (20 points).

(a) Compared to the general hash functions, what are the features of

cryptographic hash functions? (5 points)

(b) What are the relationships between three different properties of cryptographic hash functions. (5 points)

(c) Please explain the relationships between three known attacks for cryptographic protocols. (5 points)

(d) The longer the encryption key is, the more secure the encrypted data is. Do you agree with this statement. Justify your answer. (5 points)

Question 5 (12 points).

Consider the query f(x)=average weight of the people in the database x. The weight of a person ranges between 50 and 200 lb and there are at least 200 people in the database. Considering the output of query satisfies the following distribution (where y=f(x) is the answer of the query, z is the reported answer, and c is a normalization factor):

$$d_y(z) = ce^{-|z-y|}$$

Does the mechanism satisfy ε- differential privacy, for some ε? If the answer is

yes, please give the minimum such ε (under the above assumptions on the dimension of the database and the range of the weight). If the answer is no, please find a counterexample.

Question 6（10 points).

If you try to send a message x which has 128 bits securely. There are three ways to send. First, selecting a 128-bit key k uniformly at random, you can use the key as a one time pad, sending $k \oplus x$. Second, you can use the block cipher (single block) to encrypt it and send Enc_k(x) (suppose the input size and key size are all 128 bits). Third, you can use the Probabilistic Encryption (single block) to encrypt x and send Enc_k(x, R). Assume attacker will see either $k \oplus x$, Enc_k(x) or Enc_k(x, R) and knows an initial portion of x (a standard header). If attacker has time to try out every possible key $k \in \{0,1\}^{128}$, which scheme would be more secure or the same secure?

1) a) Quasi-identifiers are attributes that can be used to re-identify an individual from an external dataset that may or may not have been anonymized, however they are not directly identifying.

Sensitive attributes are attributes that specifically need protection as revealing this information can be against the wishes of the person and be against law.

eg.

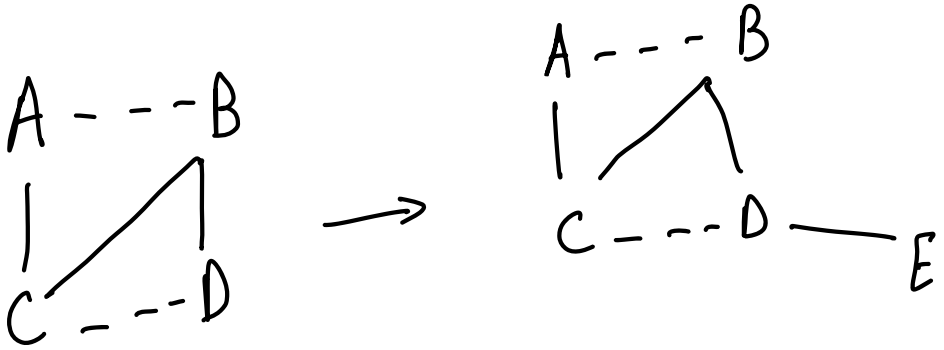
Quasi-Identifier	Sensitive Attribute
- Age	- Social Security Number
- Gender	- Salary
- ZIP Code	- Medical Conditions

1) b) Yes, I agree with that statement. We generalize quasi-identifiers so that identification becomes more difficult by linking as there should be at the minimum at least k-1 other values for every record. Doing the process the other way around could cause a loss of utility in the data.

1) c) In the context of social media (or) graph structure, k-anonymity would be defined in a way that a set of nodes should be considered k-anonymous if each node in that set is indistinguishable from at least (k-1) other nodes based on their edges.

For example,

Given A, B, C and D that can be identified



we add a node E such that it has the same connections as D. This will make the graph k-anonymous for a value of 2 for D. This can be repeated for every node.

1) d) To ensure k-anonymity for a graph structure

Step 1: Create an empty graph as "k-anon"

Step 2: Copy all the nodes from the original graph to the "k-anon" graph

Step 3: For each node in the graph, you count the number of neighbours it has.

Step 4: Group the nodes in a way that all the nodes that have equal neighbours are in the same group.

Step 5: Define a value k

Step 6: For each group that has neighbours less than the k-value, we create random dummy nodes to match the k-value.

Step 7: The returned graph should contain each node in a way that there exists (k-1) other nodes with the same relationships

1) c) Assuming that the relationships are the sensitive attributes. The attacks that work on a tabular form of data would not work because table is a flat data structure that represents a single entity. Since, the attack would have to account uniformity and distribution in a table would not apply to the graph.

2) a) To return the average weight without revealing the actual weight, we can use Laplace mechanism.

The first part would be to calculate the true average of the data

$$\text{True Average} = \frac{145 + 130 + 187 + 152 + 119}{5}$$

$$= 146.6$$

Then we calculate the sensitivity of the dataset. Sensitivity can be defined as the change in output when a single data point is added or removed.

$$\text{Sensitivity} = \frac{\text{Max Weight} - \text{Min Weight}}{n}$$

$$= \frac{187 - 119}{5} = 13.6$$

Let's consider a privacy budget of 1 to control the loss in utility.

$$\varepsilon = 1$$

To compute the noise : $\lambda = \frac{\text{Sensitivity}}{\text{Privacy Budget}}$

$$\lambda = \frac{13.6}{1}$$

Then for the query we return the noisy average

$$q'(D) = 146.6 + \text{Laplace}(13.6)$$

2) b) Considering two sensitivities :

when one datapoint is added

$$\text{Sensitivity}_1 = \frac{\text{Max Weight} - \text{Min Weight}}{n}$$

$$= 13.6$$

when two datapoints are added

$$\text{sensitivity}_2 = 2 \times \frac{187 - 119}{5}$$

$$= 2 \times 13.6$$

$$= 27.2$$

The sensitivities change drastically between the two scenarios. Essentially, as more data points are added, the privacy in the data increases, however, the data becomes more noisy.

2) i) Considering the same laplace Mechanism, we start by calculating the true counts.

No. of M = 3

No. of F = 2

Sensitivity = 1 (since maximum change in output when a single data point is added will be 1)

Privacy Budget (ε) = 1

The noise for our parameters will be

$$\text{Noise} = \frac{\text{Sensitivity}}{\text{Privacy Budget}}$$

$$\text{Noise} = \frac{1}{1} = 1$$

The generated noise is added to both true counts.

Furthermore, instead of releasing the actual data we release a binary statement which would range from either "more males" to "more females".

3) a)

Step 1: Alice makes two choices based on her preference.

Step 2: Alice adds some Laplace Noise to her selection.

Step 3: Alice sends her noisy data to the collector.

Step 4: Step 1, 2, 3 is repeated by Bob, Chris, David and Edward.

Step 5: The data collector uses noise reduction to estimate real distribution

Step 6: The movie with the highest count is considered top choice.

3) b)

	LDP	DP
Trust:	No trust in the data collector	- The data collector is a trusted entity
Noise:	Noise is added by the users	- Noise is added by the data collector
Utility Loss:	Utility Loss is higher	- Utility Loss is lower
Protection:	Privacy guaranteed for every user	- Privacy guaranteed for one record relational to the dataset.

4) a) Compared to normal hash, cryptographic hash is designed to be deterministic yet unpredictable. Cryptographic hash should should computationally infeasible to retrieve original output from hash output.

Cryptographic hashes are designed to be computationally non-colliding.

Cryptographic hashes are designed to be quick in accordance with the other properties.

4) b) Pre-image resistance states that a hash should not be mapped to an element that already has a cryptographic hash value

second pre-image states that for M1 and M2, there should be no M1 value that has the same hash as M2.

Given the above two, if a hash is collission resistant, it should be pre-image resistant. However, all pre-image resistant hashes might not be

collission resistant.

Furthermore, given cryptographic hashes should be fast, they should not sacrifice other properties for speed.

4) c) Brute force and Dictionary Attack: Both attacks try to compute the original input from hash value but differ in strategy. Both attacks are conducted offline.

Dictionary and Man-in-the-Middle: Man in the Middle attacks are conducted by inserting the attacker in the middle of a secure channel to capture data. This data can be used to conduct a dictionary later.

Brute force and Man-in-the-Middle: Similar to above, data collected through a secure channel can be used to collect information to weaken the encryption which makes brute force more feasible.

4) d) The statement is partially true, since a longer encryption would make a bruteforce attack computationally harder. However, other properties of encryption function have to hold, in order for the statement to be completely true.

5) Considering the given information, lets calculate

$$\frac{Pr[F(D1) \in S]}{Pr[F(D2) \in S]} \leq e^{\varepsilon}$$

Range = 50 − 200

Sensitivity = $\frac{200 - 50}{200} = \frac{150}{200}$

This model does not satisfy ε-differential privacy. However, to fix that we just need add Laplace Noise in accordance with calculated sensitivity to make it differentially private.

This model does not work without noise as given data if there is a record added that is not noise and the query is based on true average, the new average changes by the factor of the sensitivity. Thus making it easy to calculate new data points.

Therefore, without noise, the mechanism is not ϵ-differentially private.

6) Given unlimited computational resources, all the algorithms would be considered insecure, since the properties of cryptography account for infeasibility. However, considering relativity among the three, One time Pad would only require bitwise OR operation on each candidate key.

Block Cipher would require multiple substitutions, permutations and operations which would make it slower.

Probabilistic Encryption uses more complex functions and would make computation even slower.

Least Secure — OTP < Block Cipher < Probabilistic Encryption — Most Secure