Contest Rule

PWS2020 Committee
Cup Working Group

1st version: 2020/08/26
2nd version: 2020/09/04

PWS2020 Committee Cup WG Members

- Koji Chida (NTT)
- Hitomi Arai (RIKEN)
- Makoto Iguchi (Kii)
- Hidenobu Oguri (Fujitsu Laboratories)
- Hiroaki Kikuchi (Meiji University)
- Atsushi Kuromasa (FJCT)
- Hiroshi Nakagawa (RIKEN)
- Yuichi Nakamura (Waseda University)
- Kenshiro Nishiyama (BizReach)
- Ryo Nojima (NICT)
- Satoshi Hasegawa (NTT)
- Takuma Hatano (NS Solutions)
- Koki Hamada (NTT)
- Ryo Furukawa (NEC)
- Takao Murakami (AIST)
- Yuji Yamaoka (Fujitsu Laboratories)
- Akira Yamada (KDDI Research)
- Chiemi Watanabe (Tsukuba University of Technology)

Schedule

08/07(Fri)-08/26(Wed)	Contest entry
08/26(Wed)	Contest rule open
08/27(Thu)-09/07(Mon)	Preliminary round (Anonymization Phase)
09/09(Wed)-09/18(Fri)	Preliminary round (Attack Phase)
09/22(Tue)	Preliminary round result announcement
09/24(Thu)-10/05(Mon)	Final round (Anonymization Phase)
10/07(Wed)-10/20(Tue)	Final round (Attack phase)
10/27(Tue)	Final result announcement @ CSS2020
10/27(Tue)	Poster presentation by each team @ CSS2020 (Data anonymization and attack methods)

Overview

- PWSCUP2020: The 6th data anonymization and attack contest
- A.K.A. AMIC ("Anonymity against Membership Inference" Contest)
- As in previous years, each team will anonymize given data and attack anonymized data made by other teams.

- In previous PWSCUP, we mainly used the re-identification attack (record linkage) as a privacy measure. This year, we will use the membership inference attack as a privacy measure.
  - Membership inference: Estimate whose data is included in anonymized data.
  - The membership inference can be used as a privacy measure for many anonymization methods, including synthetic data generation that is attracting attention in the machine learning area.
  - The hide-and-seek privacy challenge hosted in NeurIP2020 (one of the top conferences in machine learning) also focuses on the membership inference attack on synthetic data.
- The membership inference attack may also be used as a privacy measure in cases where the existence of the data subject in the learning data is sensitive by nature (e.g., data analysis of a person infected with a COVID-19).

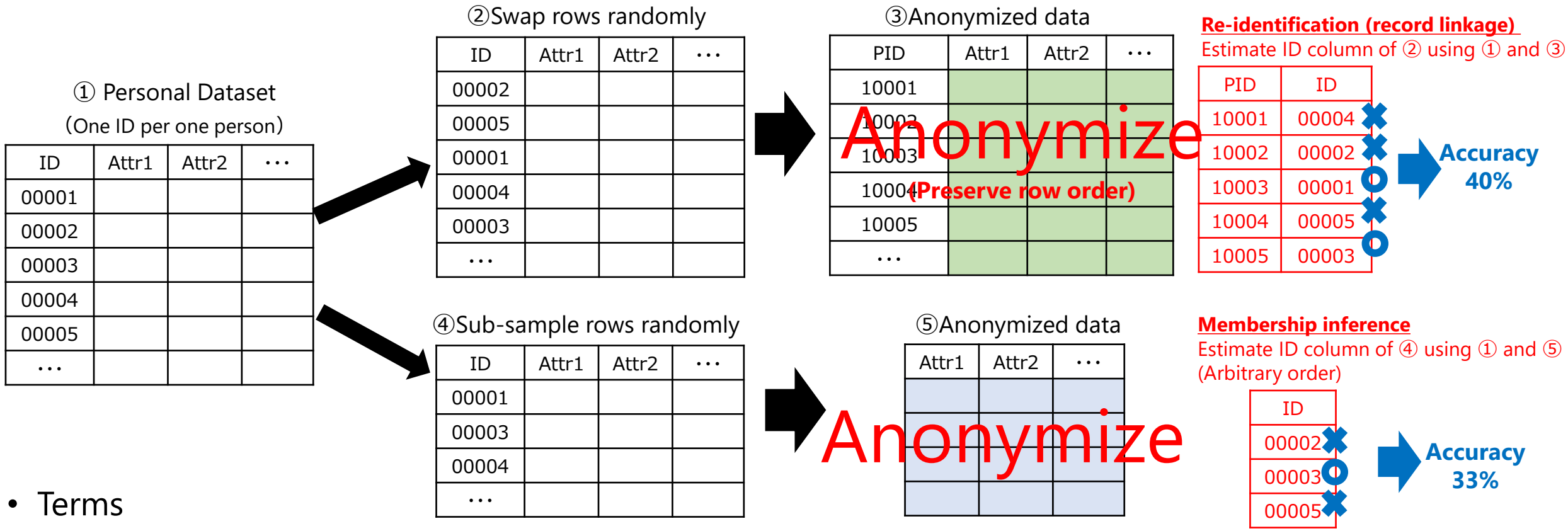
Overview (continue)

- Awards (For each award, the top 3 teams with high scores will win. Check page 10 for how the score is counted.)
  - Overall award: 1st, 2nd, and 3rd place
  - Anonymization award: 1st, 2nd, and 3rd place
  - Attack award: 1st, 2nd, and 3rd place

  } You have a chance to receive multiple awards!

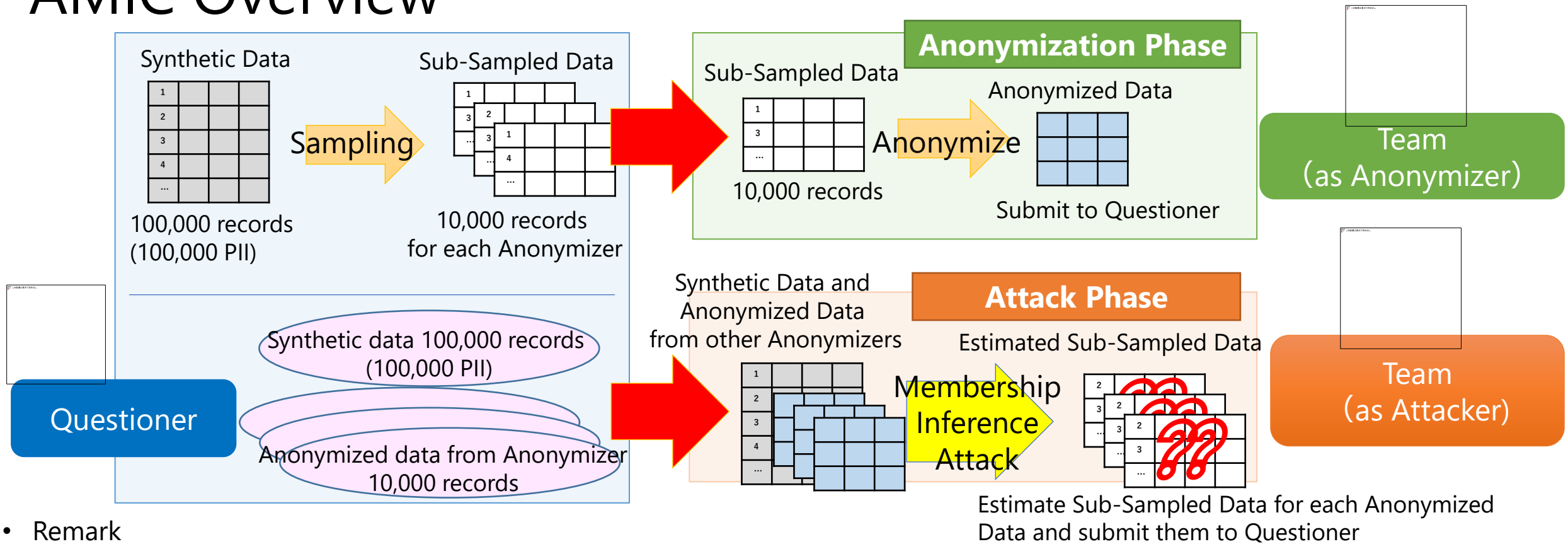
  - We will present the award winners on PWSCUP2020 site.
- Prohibited actions
  - Sharing sub-sampled data, that is separately and privately distributed to each team, to other teams.
  - Sharing codes, that implement an idea a team has invented, to other teams.
    - Discussing with other teams for brainstorming and confirming rules is OK.
  - Intentionally obstructing the committee
- Others
  - The rules of the final round may change based on the result of the preliminary round. If we change the rule, we will notify you immediately.
  - Some members of the PWS2020 committee Cup WG will also participate in the contest. We will ensure the fairness (e.g., removing them from the contact ML).
  - There will be a team from WG who will participate only in the anonymization phase for evaluating the anonymized data. This team will be excluded from the award for each phase.
  - We will present the result of preliminary rounds (overall, anonymization, and attack phase ranking) on PWSCUP2020 site. Note that no award will be given to the preliminary round winners.

Re-Identification (Record Linkage) vs Membership Inference



- Terms
  - Personal Dataset: Tabular data with personal data in each row
    - In this contest, only one row for one person (i.e., the same ID does not appear in multiple rows)
  - Attack: Performing membership inference or re-identification (record linkage). An attack is strong if its accuracy is high.
  - Anonymize: Processing personal dataset to make the data tolerant to attacks (①→③, ①→⑤, ②→③, ④→⑤)

AMIC Overview



- Remark
  - **We use synthetic data derived from Personal dataset** to prevent Anonymizers from knowing non-sampled data.
    - Synthetic data: a dataset having similar characteristics (e.g., distribution) to the personal dataset
    - We use Census Income Data Set, which is a public dataset, as a personal dataset.
  - **Synthetic data is not disclosed to Anonymizers** (only sub-sampled data is disclosed. Synthetic data is disclosed to Attackers in Attack phase)
  - Sub-sampled data is generated by **sampling Synthetic data randomly for each team**.
  - Each Attacker submits **row numbers of the most confident 100 records** from each anonymized data as a result of the membership inference attack.

Personal Dataset Used in AMIC

- **Census Income Data Set** https://archive.ics.uci.edu/ml/datasets/census+income
- Public data with 15 attributes for machine learning. Training data (32,561 records) and test data (16,281 records)
- From this dataset, we generate **synthetic data (100,000 records) with no duplicate records** using the synthetic data generation method described in the next page.
  - First, generate 1,000,000 synthetic data with duplicated records. Then, remove duplicated records and sample 100,000 records randomly
- We use **the following 9 attributes**. They are selected by considering the number and distribution of attribute values (73*8*16*7*14*6*2*99*2=2,175,731,712 possible combinations)

Table 1: Attributes and their values used in AMIC

age: continuous. [17-90]
workclass(8): Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
education(16): Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
marital-status(7): Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
occupation(14): Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
relationship(6): Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
sex(2): Female, Male. hours-per-week: continuous. [1-99] income(2): >50K, <=50K

Header→
(The header is excluded in files used in the contest)

age	workclass	education	marital-status	occupation	relationship	sex	hours-per-week	income
39	State-gov	Bachelors	Never-married	Adm-clerical	Not-in-family	Male	40	<=50K
50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	Husband	Male	13	>50K
38	Private	HS-grad	Divorced	Handlers-cleaners	Not-in-family	Male	40	<=50K
53	Private	11th	Married-civ-spouse	Handlers-cleaners	Husband	Male	40	<=50K
28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Wife	Female	40	>50K
…	…	…	…	…	…	…	…	…

Synthetic Data Generation Algorithm Used in AMIC

- [OMTH17] : Okada et al. Privacy Preserved Synthetic Datasets Generation by Using Statistics, CSS2017 3F3-4. (in Japanese)
- The algorithm generates **synthetic data having the same averages for all attributes and variance-covariance matrix** as those of the personal dataset (input).
  - The actual values differ slightly due to discretization and the maximum/minimum value adjustment.
- Category attributes are converted to dummy variables and treated as numerical attributes.
- **The number of records in synthetic data can be set to an arbitrary number.**

Input: A vector of average values for each attributes $\boldsymbol{\mu}$, variance-covariance matrix Σ , Histogram of each attribute, the number of records in synthetic data

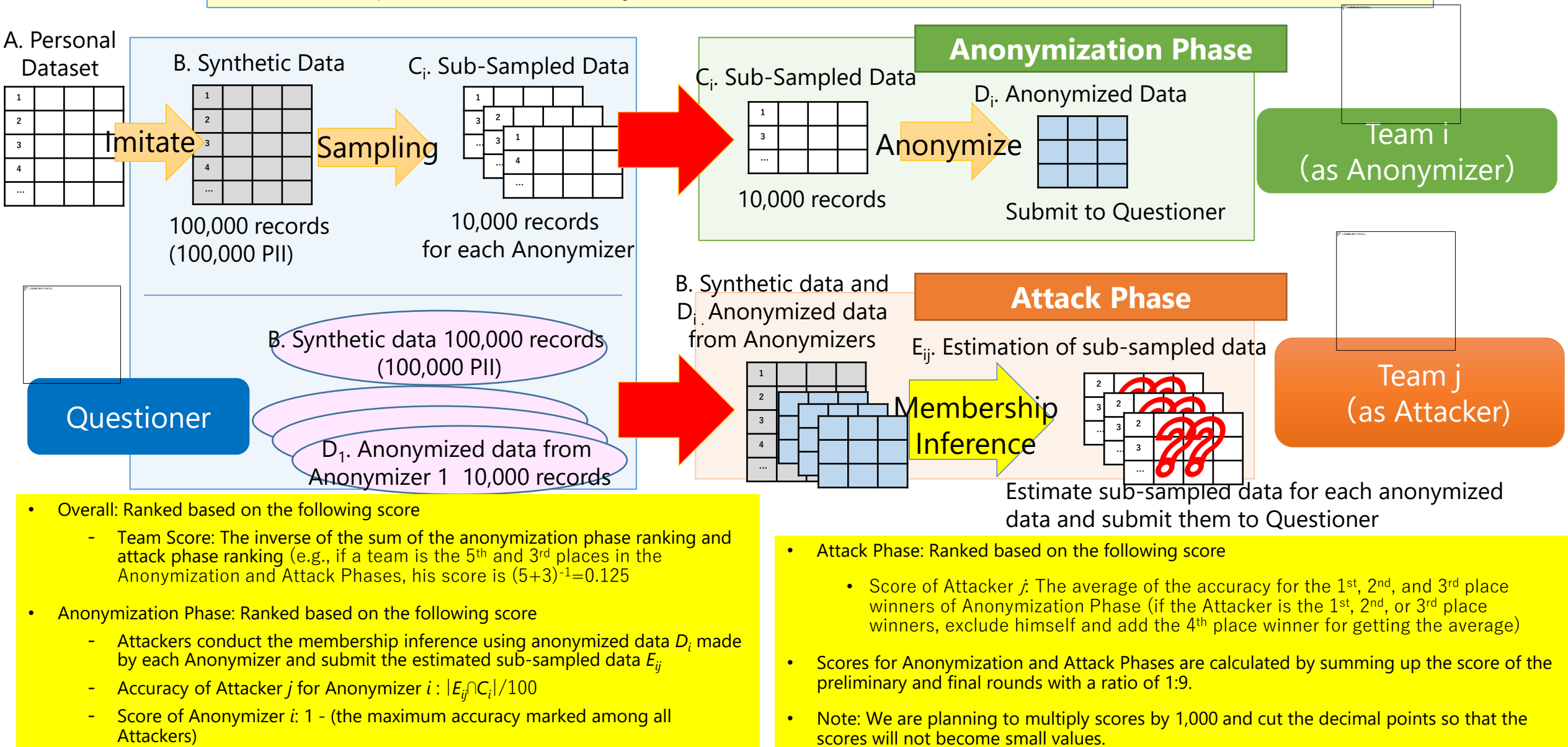
Output: Synthetic data Z

1. Generate a random dataset Y having the same histograms and record numbers as those of the input.
2. Whiten Y and generate dataset Y' where the average of each attributes is 0 and the variance-covariance matrix is an identity matrix.
3. Calculate a rotation matrix $\Sigma = U\Lambda U^T$ and a scaling matrix $\Lambda^{1/2}$ (convert each element of Λ to the square root)
4. Calculate $Y'(U\Lambda^{1/2})^T$ and generate dataset Z^* by adding $\boldsymbol{\mu}$ in each row
5. Perform discretization and the maximum/minimum value adjustment to Z^* and output the result as synthetic data Z

Scoring

A. Personal dataset (Open Data)
B. Synthetic data (Closed Data)
C_i. Sub-Sampled data for Anonymizer i
D_i. Anonymized data made be Anonymizer i
E_{ij}. Estimation of Anonymizer i's Sub-sampled data made by Attacker j

Please remember the notation "A, B, C, D, E"

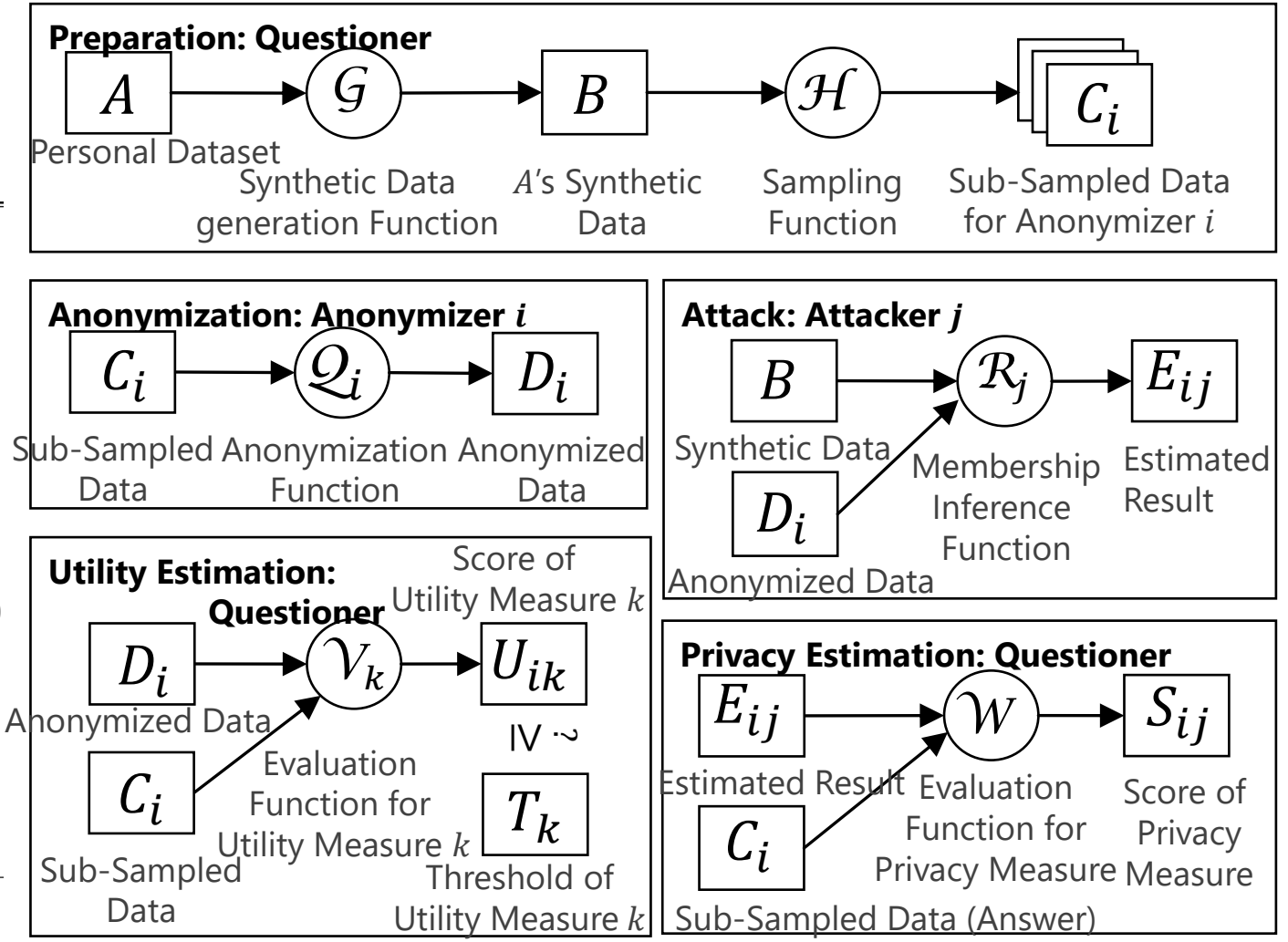


- Overall: Ranked based on the following score
  - Team Score: The inverse of the sum of the anonymization phase ranking and attack phase ranking (e.g., if a team is the 5th and 3rd places in the Anonymization and Attack Phases, his score is $(5+3)^{-1}=0.125$
- Anonymization Phase: Ranked based on the following score
  - Attackers conduct the membership inference using anonymized data D_i made by each Anonymizer and submit the estimated sub-sampled data E_{ij}
  - Accuracy of Attacker j for Anonymizer i : $|E_{ij} \cap C_i|/100$
  - Score of Anonymizer i: 1 - (the maximum accuracy marked among all Attackers)

- Attack Phase: Ranked based on the following score
  - Score of Attacker j: The average of the accuracy for the 1st, 2nd, and 3rd place winners of Anonymization Phase (if the Attacker is the 1st, 2nd, or 3rd place winners, exclude himself and add the 4th place winner for getting the average)
- Scores for Anonymization and Attack Phases are calculated by summing up the score of the preliminary and final rounds with a ratio of 1:9.
- Note: We are planning to multiply scores by 1,000 and cut the decimal points so that the scores will not become small values.

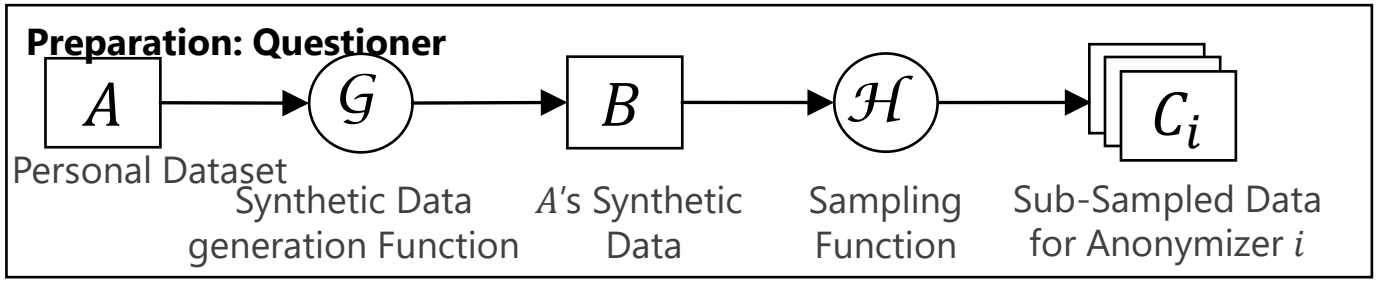
Notation and flow

A	Personal data set
B	Synthetic data of A
C_i	Sub-sampled data of anonymizer i
D_i	Anonymized data of anonymizer i
E_{ij}	Estimated result for anonymizer i by attacker j
$\mathcal{G}$	Synthetic data generation function $(B = \mathcal{G}(\mathsf{seed}, A))$
$\mathcal{H}$	Sub-sampling function $(C_i = \mathcal{H}(i, B))$
$\mathcal{Q}_i$	Anonymization function of anonymizer i $(D_i = \mathcal{Q}_i(C_i))$
$\mathcal{R}_j$	Membership inference function of attacker j $(E_{ij} = \mathcal{R}_j(B, D_i))$
S_{ij}	Score of privacy measure of anonymizer i against attacker j
T_k	Threshold of utility measure k
U_{ik}	Score of utility measure k of anonymizer i
$\mathcal{V}_k$	Estimation function of utility measure k $(U_{ik} = \mathcal{V}_k(C_i, D_i))$
$\mathcal{W}$	Estimation function of privacy measure $(S_{ij} = \mathcal{W}(C_i, E_{ij}))$



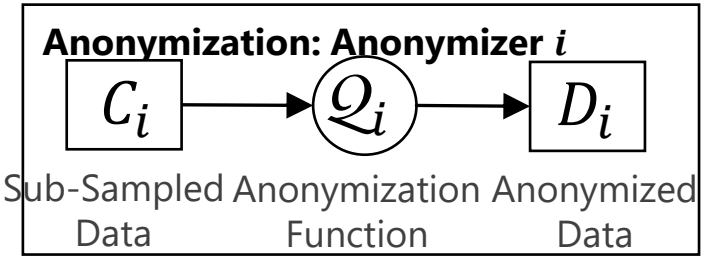
- Utility measure
  - A quantitative evaluation of the similarity between Personal Dataset and their Anonymized Data, such as statistics and the prediction/classification results by machine learning.
  - The score is high if these data are more similar.
- Privacy measure
  - A quantitative evaluation of the tolerance against attack.
  - The score is high if the accuracy of membership inference against the data is low.

Preparation: Questioner's task



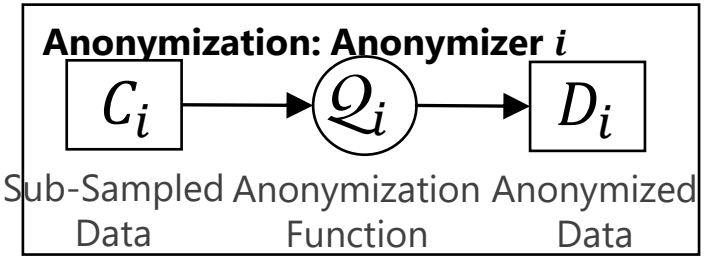
- Personal Dataset
  - Extract 30,162 records with no missing value from Census Income Data Set training data (32,561 records)
  - Use 9 attributes: age, workclass, education, marital-status, occupation, relationship, sex, hours-per-week, income
- Synthetic Data Generation Function
  - Use [OMTH17] method
  - The function can output any number of records (Here, we generate 100,000 records with no duplication)
  - Average and variance of each attribute and covariance between two attributes are similar to the original personal dataset (category attributes are converted to dummy variables)
  - **Implemented code: gen.py → Released for your trial**
- Synthetic Data
  - Dataset with 100,000 records with no duplicate. Its average, variance, and covariance are similar to the personal dataset.
  - This dataset is not disclosed to Anonymizer (only a sub-sampled data will be disclosed)
- Sampling Function
  - Take a dataset as an input, output records by sampling the dataset with the specified sample rate (0-1).
  - In the contest, we set the sampling rate to 0.1. Sub-sampled data with 10,000 records is generated by sampling 100,000 record Synthetic Data randomly.
  - **Implemented code: randomsampling.py → Released for your trial**
- Sampled Data
  - 10,000 records generated by sampling the synthetic data randomly. Each Anonymizer will receive different sampled data.
- Other
  - Questioner will record and hold line numbers of synthetic data corresponding to sampled data for each Anonymizer (Note: **the line number starts with 0**)

Anonymization: Anonymizer's task



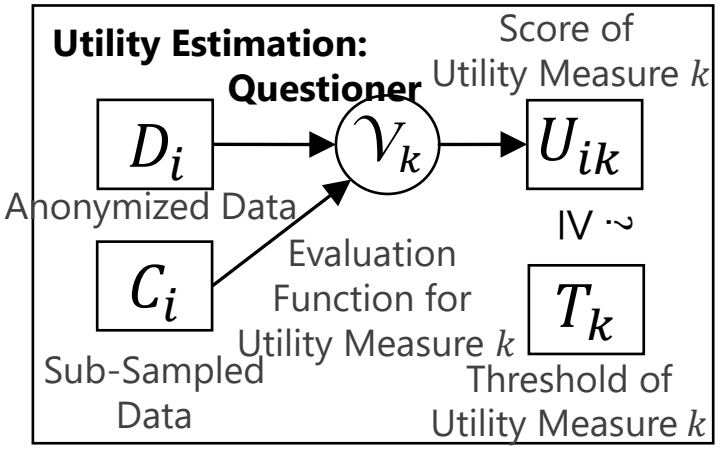
- Sampled Data
  - 10,000 records generated by sampling the synthetic data randomly. Each Anonymizer will receive different sampled data.
- Anonymization Function
  - Implemented by Anonymizer
  - The function generates anonymized data with high privacy score (i.e., strong against membership inference attacks by Attackers)
  - **We will release some sample codes (to be described later)**
- Anonymized Data
  - To be submitted to Questioner
  - The number of records in anonymized data does not have to match with that of sub-sampled data, but the number has to be from 1,000 to 100,000.
  - Note that the privacy score is designed to descrease as the difference in the number of records increase.
  - **The privacy score must be higher than the designated threshold value (to be described later), or your team will be disqualified.**
  - **Limitation: For the convenience of the privacy score, the attributes of anonymized data are 9 attributes covered in Table 1 on page 7. The values of the attributes should match with the ones covered in Table 1 on Page 7.**
    - Example: The values of the attribute "age" is an integer from 17 to 99. The values of the attribute "age" in anonymized data is also an integer from 17 to 99 (values such as 100, 20s, [20-24] are not allowed).

Anonymization: Sample Codes



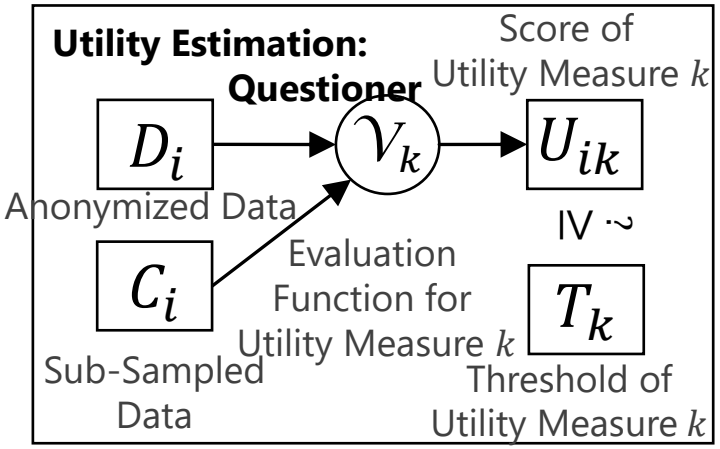
- synthetic.py (Synthetic data generator)
  - Generate synthetic data with similar mean and variance of each attribute and covariance between two attributes of the input data.
  - Can specify the number of records in synthetic data.
- rr.py (Randomizer/Randomized Response/PRAM)
  - Given the retention probability *p* (0-1), maintain the value of each cell with probability *p* and replace the value with a random attribute value with probability 1-*p*.
- rrp.py (Randomizer/Randomized Response/PRAM)
  - Given the retention probability *p* (0-1), maintain the value of each cell with probability *p* and replace the value with a random attribute value according to the distribution of input data with probability 1-*p*.
- kanony.py (*k*-anonymity with record suppression)
  - Given the attribute(s) and threshold *k*, check the values of the specified attributes. If there are *k* or more identical sets of attribute values, maintain them. If there are *k*-1 or fewer sets of attribute values, delete the corresponding records.

Utility Estimation: Questioner and Anonymizer's task



- Utility Measure
  - A quantitative evaluation of the similarity between personal dataset and their anonymized data, such as statistics and the prediction/classification results by machine learning.
  - The score is high if these data are more similar.
- Evaluation Function for Utility Measure
  - Anonymizer needs to generate anonymized data with **its utility scores of the histogram, the variance-covariance matrix, and the decision tree analysis** being greater than or equal to the threshold values (the details explained in the following pages).
  - **Evaluation code (utilityfunc.py) is released** → Anonymizer can check if the utility score of their anonymized data is above the threshold.
- Score and Threshold for Utility Measure
  - Details explained in the following pages
- Note
  - Questioner will execute utilityfunc.py on anonymized data submitted by Anonymizers to check if the utility score is above the threshold value → **If the score is below the threshold, the team will be disqualified !**

Utility Measure: Histogram



- Be aware that you will be disqualified if anonymized data you've submitted to Questioner has the utility score below the threshold !!

Histgram:

The frequencies of each value in each attribute of anonymized data D_i and sub-sampled data C_i, respectively. For each value X_{lm} of attribute X_l and the frequencies X^D_{lm} and X^C_{lm} of D_i and C_i resp., the score of utility measure is defined as follows.

$$U_{i\,\text{histogram}} = 1 - \frac{\sum_{l,m} \left| X^D_{lm} - X^C_{lm} \right|}{2C_i^{\text{Rec}} C_i^{\text{Att}}}, \qquad (1)$$

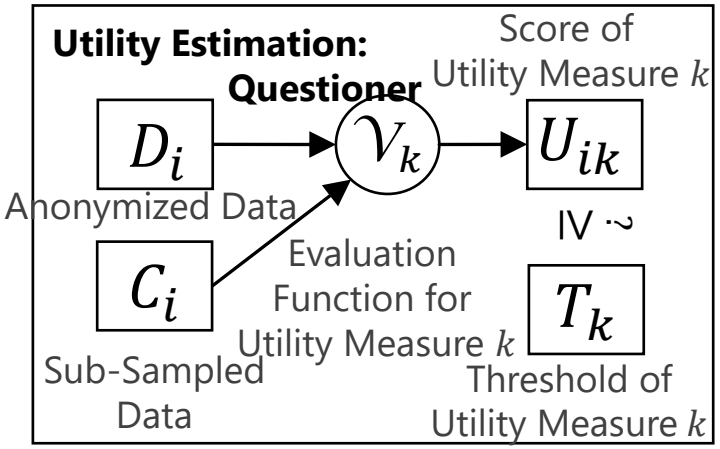
where C_i^{Rec} and C_i^{Att} are the numbers of records and attributes of C_i, respectively.

- Evalution code utilityfunc.py is released. You can check if your anonymized data clears the threshold.
- Utility score threshold: $T_{\text{histgram}} = 0.99$

Attribute	Type	Values
X_1: age	Integer	17 – 90
X_2: workclass	Categorical	8 values such as "Private"
X_3: education	Categorical	16 values such as "Bachelors"
X_4: marital-status	Categorical	7 values such as "Married-civ-spouse"
X_5: occupation	Categorical	14 values such as "Tech-support"
X_6: relationship	Categorical	6 values such as "Wife"
X_7: sex	Categorical	Female, Male
X_8: hours-per-week	Integer	1 – 99
X_9: income	Categorical	>50K, <=50K

Utility Measure: Variance-Covariance Matrix

- Be aware that you will be disqualified if anonymized data you've submitted to Questioner has the utility score below the threshold !!



Variance-covariance matrix:

The score is based on the difference of each element of the variance-covariance matrices obtained from anonymized data D_i and sub-sampled data C_i. Assume all attributes are numeric thanks to a dummy variable conversion for simplicity. For variance σ_{ll} of X_l and covariance $\sigma_{ll'}$ of X_l and $X_{l'}$, the score of utility measure is defined as follows.

$$U_{i\,\mathrm{VCM}} = \left(\sum_{l,l'} \left| \sigma^D_{ll'} - \sigma^C_{ll'} \right| \right)^{-1}, \qquad (2)$$

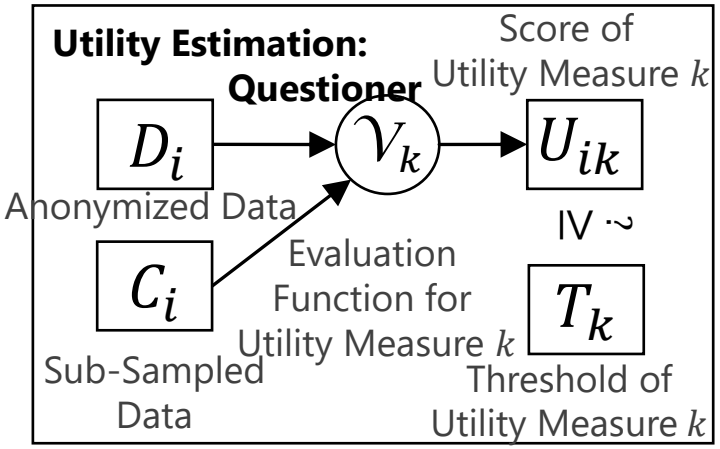
where $\sigma^D_{ll'}$ and $\sigma^C_{ll'}$ are variance or covariance of D_i and C_i resp. and Eq (2) returns ∞ when $\sum_{l,l'} \left| \sigma^D_{ll'} - \sigma^C_{ll'} \right| = 0$.

Attribute	Type	Values
X_1: age	Integer	17 – 90
X_2: workclass	Categorical	8 values such as "Private"
X_3: education	Categorical	16 values such as "Bachelors"
X_4: marital-status	Categorical	7 values such as "Married-civ-spouse"
X_5: occupation	Categorical	14 values such as "Tech-support"
X_6: relationship	Categorical	6 values such as "Wife"
X_7: sex	Categorical	Female, Male
X_8: hours-per-week	Integer	1 – 99
X_9: income	Categorical	>50K, <=50K

- Evalution code utilityfunc.py is released. You can check if your anonymized data clears the threshold.
- Utility score threshold: $T_{VCM} = 0.4$

Utility Measure: Decision tree analysis

- Be aware that you will be disqualified if anonymized data you've submitted to Questioner has the utility score below the threshold !!



Decision-tree analysis:

The score is based on the value of F-measure obtained from decision-tree functions $\mathcal{Y}^D_{X_l}$ of anonymized data D_i and $\mathcal{Y}^C_{X_l}$ of sub-sampled data C_i, where X_l is an objective variable. The test data are 16,281 training records $Z_0, \ldots, Z_{16280}$ in Census Income Data Set. The objective variables are X_6: relationship and X_9: income. Define TP_6, FP_6, FN_6, FN_6 are respectively the frequencies such that

$$(\mathcal{Y}^D_{X_6}(Z_m), \mathcal{Y}^C_{X_6}(Z_m)) = \begin{cases} \text{“Husband”, “Husband”} \\ \text{“Husband”, “Others”} \\ \text{“Others”, “Husband”} \\ \text{“Others”, “Others”} \end{cases}$$

for $m = 0, \ldots, 16280$. Then the score of utility measure for X_6 is defined as follows.

$$U_{i\,\mathrm{DTA}} = \frac{2\mathrm{Recall} \cdot \mathrm{Precision}}{\mathrm{Recall} + \mathrm{Precision}}, \qquad (3)$$

where Precision$= TP_6/(TP_6 + FP_6)$ and Recall$= TP_6/(TP_6 + FN_6)$. The score of utility measure for X_9 can be defined similarly.

Attribute	Type	Values
X_1: age	Integer	17 – 90
X_2: workclass	Categorical	8 values such as “Private”
X_3: education	Categorical	16 values such as “Bachelors”
X_4: marital-status	Categorical	7 values such as “Married-civ-spouse”
X_5: occupation	Categorical	14 values such as “Tech-support”
X_6: relationship	Categorical	6 values such as “Wife”
X_7: sex	Categorical	Female, Male
X_8: hours-per-week	Integer	1 – 99
X_9: income	Categorical	>50K, <=50K

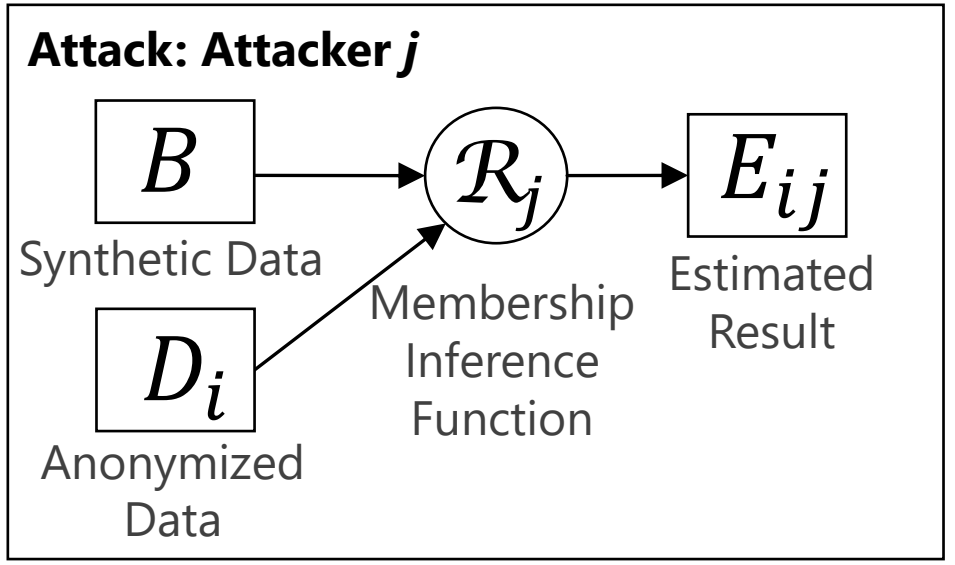
- Evaluation code utilityfunc.py is released. You can check if your anonymized data clears the threshold.
- Utility score threshold: $T_{\mathrm{DTA}} = 0.85$ (for both X_6 and X_9)

Decision tree analysis algorithm

- Decision tree analysis
  - Use sklearn.tree.DecisionTreeClassifer in scikit-learn
    - https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifer.html

- DecisionTreeClassifer parameters are as follows:
  - random_state = 0
  - max_depth = 5 (for income estimation)
  - max_depth = 3 (for relationship estimation)
  - For other parameters, use the default settings.

Attack: Attacker's task



- Synthetic data, anonymized data from Anonymizers
  - Receive from Questioner
- Membership Inference Function
  - From synthetic and anonymized data, estimate which records in synthetic data is sampled and anonymized.
  - **Sample code attack.py is released**
    - Based on the distance between each record of synthetic data *B* and anonymized data D_i, estimate sub-sampled data C_i and generate the estimated result E_{ij}.
    - Calculate Euclidean distance between records
      - For an integer attirbute, the distance is the difference of values
      - For a category attribute, compare values and count up the number of matching values.
    - Return the row number of synthetic data *B* record that has the smallest distance from each record in anonymized data

Anonymized Data

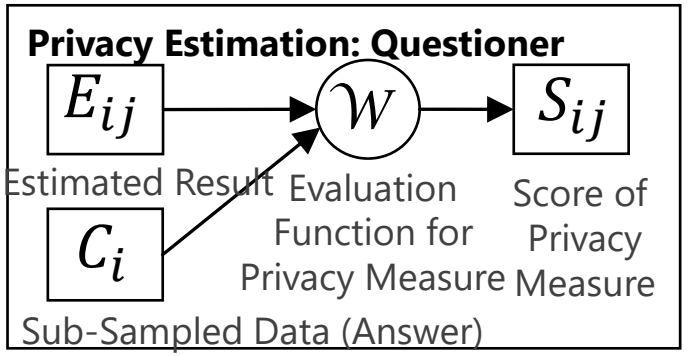
37	Male	<=50K
44	Female	>50K
28	Female	<=50K

Synthetic Data

			Dist.
43	Male	<=50K	6
39	Male	>50K	3
52	Female	>50K	17
…	…	…	

- Estimated Result
  - Submitted to Questioner from Attacker
  - Estimate which records in synthetic data are used to generate anonymized data.
  - 100 row numbers of synthetic data (Note: the row number starts from 0)
- TIPS (?)
  - Anonymized data and sub-sampled data have similar histogram, variance-covariance matrix, and classification results of decision tree analysis (meet the threshold restriction).
    → Find sub-sampled data candidate from synthetic data that has similar histogram, variance-covariance matrix, and classificaiotn results of decision tree analysis with those of anonymized data.

Privacy Estimation: Qestioner's task



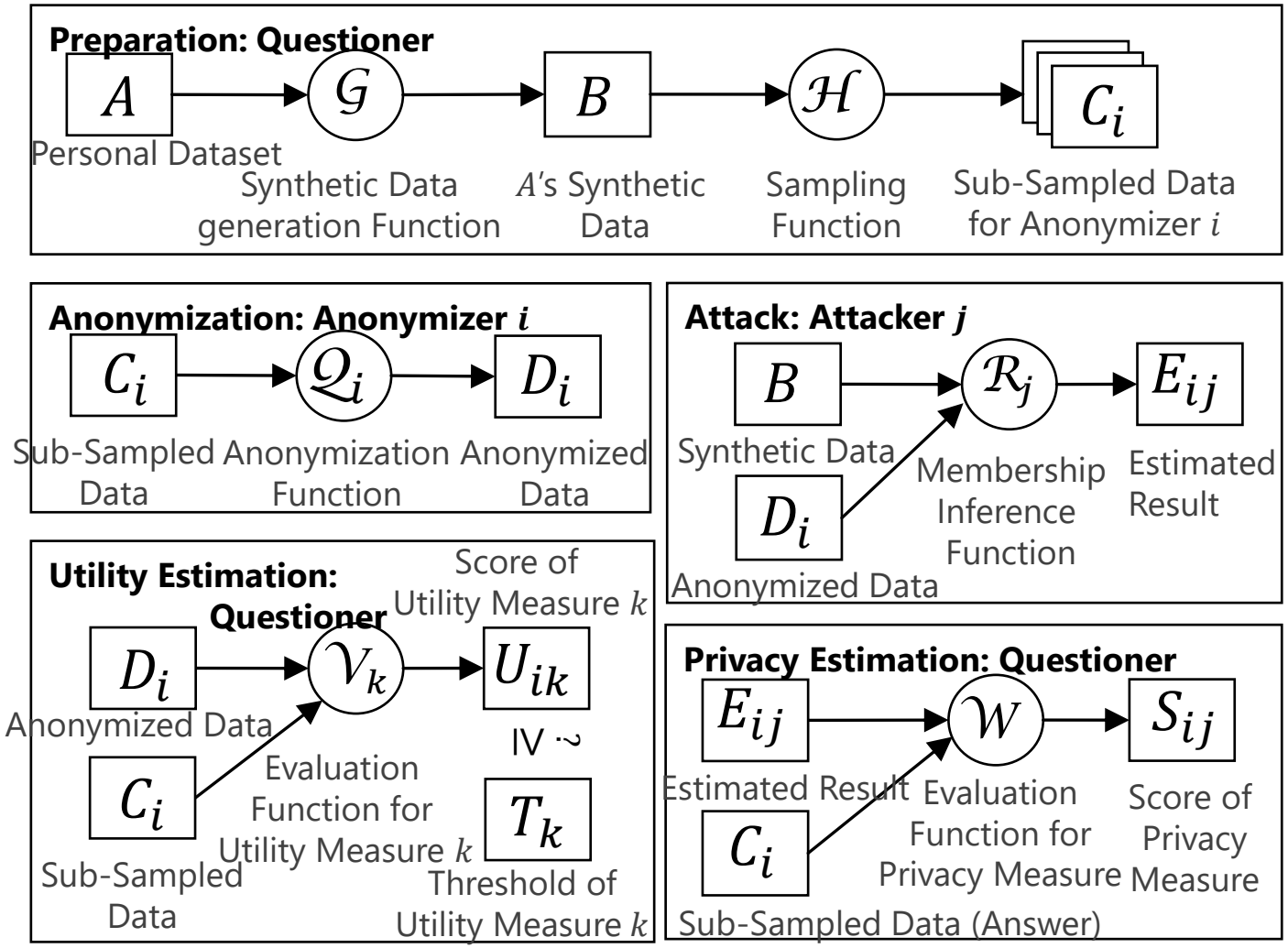
- Estimated Result
  - 100 row numbers of synthetic data that represent the estimate of which records in synthetic data are used to generate anonymized data submitted by Anonymizer (Note: the row number starts at 0)
- Answer (Sub-sampled Data)
  - Sub-sampled data that Questioner distributed to each Anonymizer
  - Only the row numbers that indicate which rows in synthetic data were sampled is needed (Note: the row number starts at 0)
- Evaluation Function of Privacy Measure
  - Return the number of matches between an estimated result (100 rows) and the answer (10,000 rows) as the evaluation score of the privacy measure.
    - The evaluation score of the privacy measure ranges from 0 to 100.
  - **Implemented code privacymeasure.py** released for trial.

Summary of Data and Codes provided by Questioner

- Data and codes provided to Anonymizers
  - Sub-sampled data (independent per Anonymizer)
  - Anonymization sample codes
    - synthetic.py
    - rr.py
    - rrp.py
    - kanony.py

  Generate anonymized data and submit it to Questioner
  - Utility index evaluation function utilityfunc.py
- Data and codes provided to Attackers
  - Synthetic data
  - Anonymized data generated by Anonymizers
  - Attack sample code: attack.py

  Submit an estimated result (attack result) to Questioner
- Sample data and codes
  - Census Income Data Set
  - Sample synthetic data
  - Sample sub-sampled data
  - Sample anonymized data
  - Sample answer data
  - Synthetic data generation function: gen.py
  - Sampling function: randomsampling.py
  - Privacy Evaluation Function: privacymeasure.py
- Thresholds for Utility Measure
  - Histogram: 0.99
  - Variance-Covariance Matrix: 0.4
  - Decision Tree Analysis: 0.85



Data and Code Formats

- Character code: UTF-8
- pre/main: preliminary round=pre, final round=main

category	Description	filename	format	Remark
Data retained by Questioner	Row numbers of sub-sampled data	[pre/main]_answer_[anonymizer number].index	Index file with 1 column and 100 rows	• Row numbers are set from “0”
Data and codes submitted to Anonymizers	Sub-sampled data	[pre/main]_samplingdata_[anonymizer number].csv	CSV file with 9 columns and 10,000 rows	• Anonymizer numbers: 01 to 99
	Sample code 1 for anonymization	synthetic.py	Please refer readme.txt	
	Sample code 2 for anonymization	rr.Py	Please refer readme.txt	
	Sample code 3 for anonymization	rrp.py	Please refer readme.txt	
	Sample code 4 for anonymization	kanony.py	Please refer readme.txt	
	Utility estimation function	utilityfunc.py	Please refer readme.txt	
	Test data for decision tree analysis	test.csv	CSV file with 9 columns and 15,060 rows	
Data received from Anonymizers	Anonymized data	[pre/main]_anonymizeddata_[anonymizer number].csv	CSV file with 9 columns and 1,000 to 100,000 rows	
Data and codes submitted to Attackers	Synthetic data	[pre/main]_syntheticdata.csv	CSV file with 9 columns and 100,000 rows	
	Anonymized data for all anonymizers	[pre/main]_anonymizeddata_[anonymizer number].csv	CSV file with 9 columns and 1,000 to 100,000 rows	
	Sample code for attack	attack.py	Please refer readme.txt	
Data received from Attackers	Estimated data for membership inference	inference_[anonymizer number]_[attacker number].index	CSV file with 1 column and 100 rows	
Data and codes for trial	Census Income Data Set	census_income.data.csv	CSV file with 9 columns and 30,162 rows	
	Synthetic data	trial_syntheticdata.csv	CSV file with 9 columns and 100,000 rows	
	Sub-sampled data	trial_samplingdata.csv	CSV file with 9 columns and 10,000 rows	
	Answer data for membership inference	trial_answer.index	CSV file with 1 column and 10,000 rows	
	Anonymized data	trial_anonymizeddata.csv	CSV file with 9 columns and 10,000 rows	
	Synthetic data generation function	gen.py	Please refer readme.txt	
	Sub-sampling function	randomsampling.py	Please refer readme.txt	• Random sampling
	Privacy estimation function	privacymeasure.py	Please refer readme.txt	

These data are just for a trial. You do not need to use them when anonymizing and attacking the actual contest data.

Questionnaire

- For all anonymizers, please answer a questionnaire below after the preliminary/final-anonymization phase.
  - We will inform the attackers about the anonymization methods you used
  - The reason is an analyst might offer what anonymization methods are used in the anonymized data in practice

Privacy Measures	
Differential Privacy	The value ϵ, δ :
k-Anonymity	The value *k* :
Pk-Anonymity	The value *k* :
δ-Presence	The value δ :
Others :	The value ?? :

Anonymization Methods	
Record Deletion	Applied Record Numbers (roughly) :
Top (Bottom) Coding	Applied Attribution Numbers (1 and/or 8) : Applied Record Numbers (Roughly) :
Partial Records Extraction	Applied Record Numbers (roughly) :
Microaggregation	Applied Attribution Numbers (1 and/or 8) : Applied Record Numbers (Roughly) :
Data Exchange (Swapping)	Applied Attribution Numbers (1 to 9) : Applied Record Numbers (Roughly) :
Noise Addition	Applied Attribution Numbers (1 to 9) : Applied Record Numbers (Roughly) :
Synthetic Data Generation	Applied Attribution Numbers (1 to 9) : Applied Record Numbers (Roughly) :
Dummy Record Addition	Applied Record Numbers (roughly) :
Randomization / Randomized Response / PRAM	Applied Attribution Numbers (1 to 9) : Applied Record Numbers (Roughly) :
Others :	Applied Attribution Numbers (1 to 9) : Applied Record Numbers (Roughly) :