Database & Storage Security

Prof. Dr. Mohammad Abu Yousuf
yousuf@juniv.edu

- Security Problem Description in SDB

Statistical Database

- A statistical database (SDB) system is a database system that enables its users to retrieve only aggregate statistics (e.g., sample mean and count) for a subset of the entities represented in the database.
- A statistical database is a database which provides statistics on subsets of records
- Statistics may be performed to compute SUM, MEAN, MEDIAN, COUNT, MAX AND MIN of records

Statistical Database

- Statistics are derived from a database by means of a **characteristic formula**, *C*, which is a logical formula over the values of attributes. A characteristic formula uses the operators OR, AND, and NOT (+, ·, ~), written here in order of increasing priority.
- For example, the formula

  (Sex = Male) · ((Major = CS) + (Major = EE))

  specifies all male students majoring in either CS or EE.

  For simplicity, omit attribute names when they are clear from context. Thus, the preceding formula becomes

  Male · (CS + EE).

Types of Statistical Databases

- **Static** – a static database is made once and never changes
- Example: U.S. Census

- **Dynamic** – changes continuously to reflect real-time data
- Example: most online research databases

Types of Statistical Databases

- **Centralized** – one database
- **General purpose** – like census

- **Decentralized** – multiple decentralized databases
- **Special purpose** – like bank, hospital, academia, etc

Accuracy vs. Confidentiality

Accuracy –

Researchers want to extract accurate and meaningful data

Confidentiality –

Patients, laws and database administrators want to maintain the privacy of patients and the confidentiality of their information

Data Compromise

- A statistical user of an underlying database of individual records is restricted to obtaining only aggregate, or statistical, data from the database and is prohibited access to individual records.

- The inference problem in this context is that a user may infer confidential information about individual entities represented in the SDB. Such an inference is called a **compromise**.

Data Compromise

- **Exact compromise** – a user is able to determine the exact value of a sensitive attribute of an individual
- **Partial compromise** – a user is able to obtain an estimator for a sensitive attribute with a bounded variance
- **Positive compromise** – determine an attribute has a particular value
- **Negative compromise** – determine an attribute does not have a particular value
- **Relative compromise** – determine the ranking of some confidential values

Data Compromise

- Consider the hospital database contains these data about patients:

  (Age, Sex, Employer, Social Security Number, Diagnosis Type).
- Suppose there is a malevolent researcher who wants to obtain information about the diagnosis type of a given patient, Mr. X. A malevolent user who wants to compromise the database is referred to as a snooper.
- In our example, assume that the snooper knows the age and employer of Mr. X. He can then issue the query

  Q1: COUNT (Age = 42) & (Sex = Male) &(Employer = ABC).

Data Compromise

- If the answer is 1, the snooper has located Mr. X and can then issue such queries as

  Q2: COUNT (Age = 42) & (Sex = Male) & (Employer = ABC) & (Diagnosis Type = Schizophrenia).

Analysis:

- If the answer to Q2 is 1, the database is said to be positively compromised and the user is able to infer that Mr. X has the diagnosis type schizophrenia.
- If the answer is 0, the database is said to be partially compromised, because the user was able to infer that the diagnosis type of Mr. X is not schizophrenia

Data Compromise

- Partial compromise refers to the situation in which some inference about a confidential attribute of an entity can be made, even if the exact value cannot be determined.
-
- It may take the form of a negative compromise, that is, it is inferred that an attribute of a certain entity does not lie within a given range.

Data Compromise

- **Types of user:**
- As we have seen from the above example, there are basically three types of authorized users:

  1) the non-statistical user (e.g., the physician in the hospital example) who is authorized to issue queries and update the database,

  2) the researcher who is authorized to retrieve aggregate statistics from the database,

  3) and the snooper(a person who is making improper use of the data normally available to him as an authorized user) who is interested in compromising the database.

Security Methods

- Access Restriction
- Query Set Restriction
- Micro-aggregation
- Data Perturbation
- Output Perturbation
- Auditing

Access Restriction

- Databases normally have different access levels for different types of users.

- Access restrictions can control reading and editing of database objects. The current user is able to read or edit a database object only if the access restrictions allow them to perform these actions. Otherwise a read or edit operation will not be executed on the database object.

Access Restriction

- User ID and passwords are the most common methods for restricting access
  - In a medical database:
    - Doctors/Healthcare Representative – full access to information
    - Researchers – only access to partial information (e.g. aggregate information)
- Statistical database: allow query access only to aggregate data, not individual records

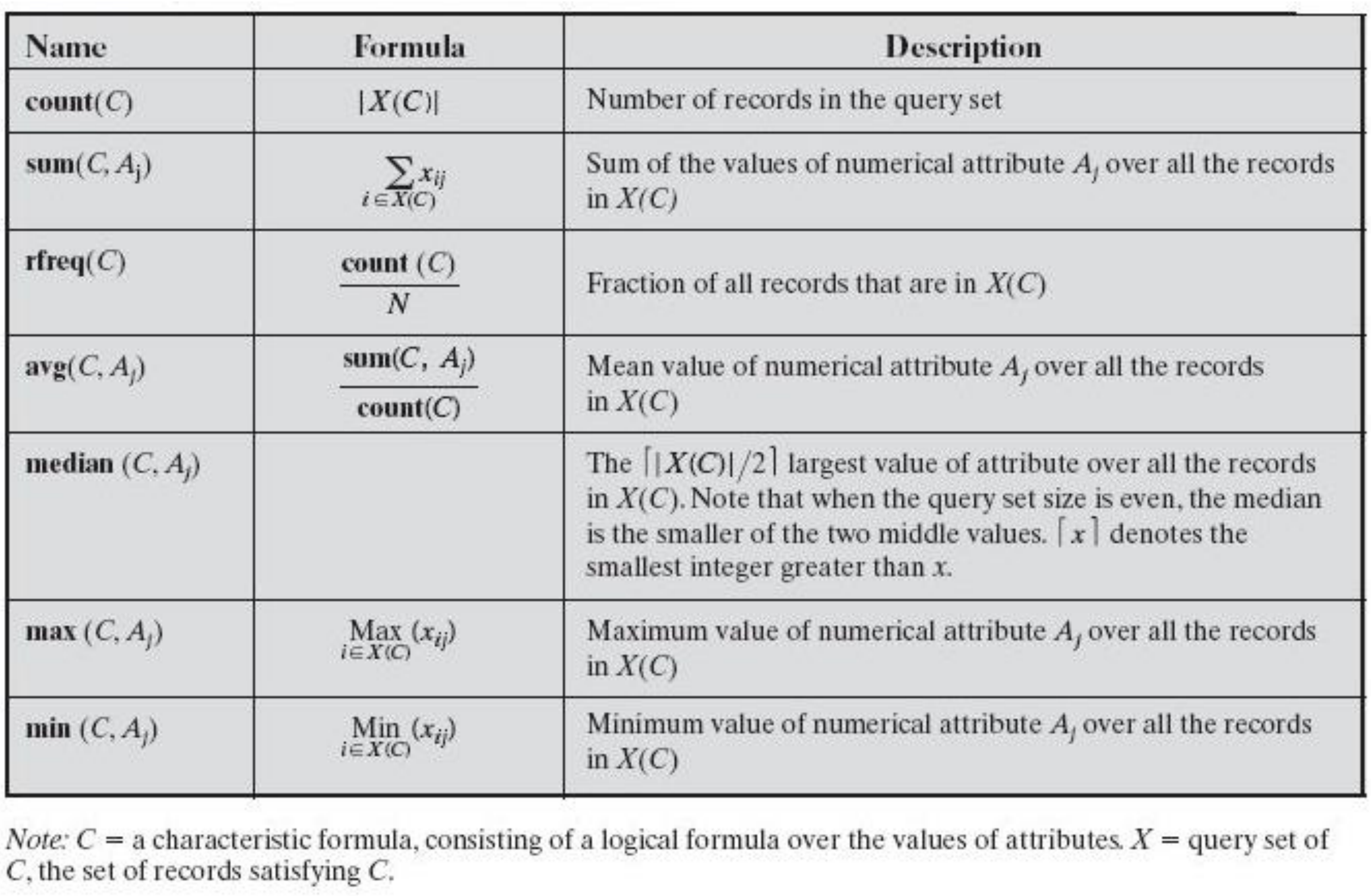
Query Set Restriction

- Query set restriction
  - Query size control
  - Query set overlap control
  - Query auditing

Query Set Size Control

- Rejects a query that can lead to a compromise (disclosure). The answers provided are accurate.
- A query-set size control can limit the number of records that must be in the result set
- Allows the query results to be displayed only if the size of the query set satisfies the condition
- Setting a minimum query-set size can help protect against the disclosure of individual data.

The **query set** of characteristic formula *C*, denoted as X(*C*), is the set of records matching that characteristic. For example, for *C* = Female · CS, X(*C*) consists of records 1 and 4, the records for Allen and Davis.



Name	Formula	Description
count(C)	$\lvert X(C)\rvert$	Number of records in the query set
sum(C, A_j)	$\sum_{i \in X(C)} x_{ij}$	Sum of the values of numerical attribute A_j over all the records in $X(C)$
rfreq(C)	$\frac{\textbf{count}\ (C)}{N}$	Fraction of all records that are in $X(C)$
avg(C, A_j)	$\frac{\textbf{sum}(C,\ A_j)}{\textbf{count}(C)}$	Mean value of numerical attribute A_j over all the records in $X(C)$
median (C, A_j)		The $\lceil \lvert X(C)\rvert/2 \rceil$ largest value of attribute over all the records in $X(C)$. Note that when the query set size is even, the median is the smaller of the two middle values. $\lceil x \rceil$ denotes the smallest integer greater than x.
max (C, A_j)	$\underset{i \in X(C)}{\text{Max}}\ (x_{ij})$	Maximum value of numerical attribute A_j over all the records in $X(C)$
min (C, A_j)	$\underset{i \in X(C)}{\text{Min}}\ (x_{ij})$	Minimum value of numerical attribute A_j over all the records in $X(C)$

Note: C = a characteristic formula, consisting of a logical formula over the values of attributes. X = query set of C, the set of records satisfying C.

Query Set Size Control

- The query-set-size control method permits a statistic to be released only if the size of the query set |C| (i.e., the number of records included in the response to the query) satisfies the following condition

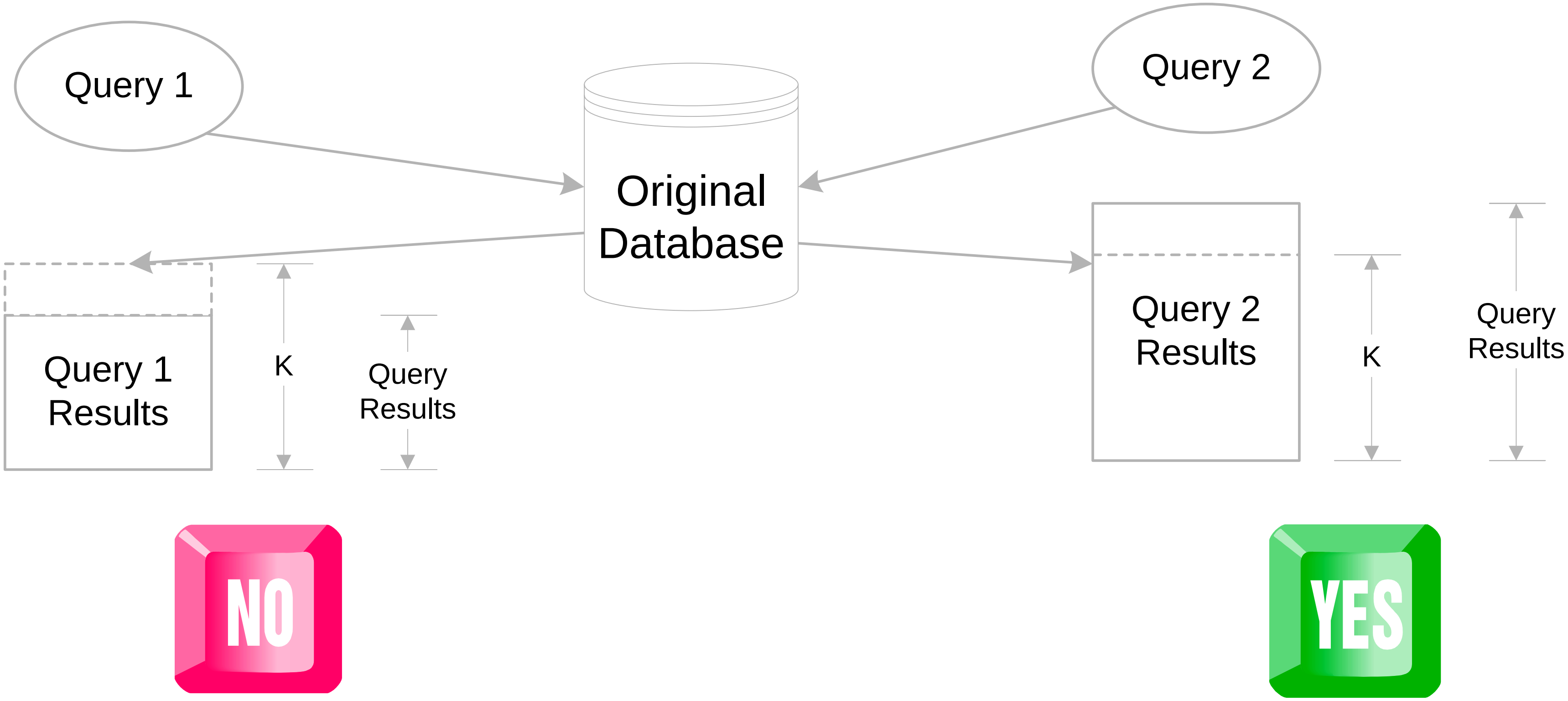
  $$K <= |C| <= L – K$$

  where L is the size of the database (the number of records represented in the database) and K is a parameter (set by the DBA) that satisfies

  $$0 <= K <= L/2.$$

  K represents the minimum number or records to be present for the query set

Query Set Size Control



Query 1

Original Database

Query 2

Query 1 Results

K

Query Results

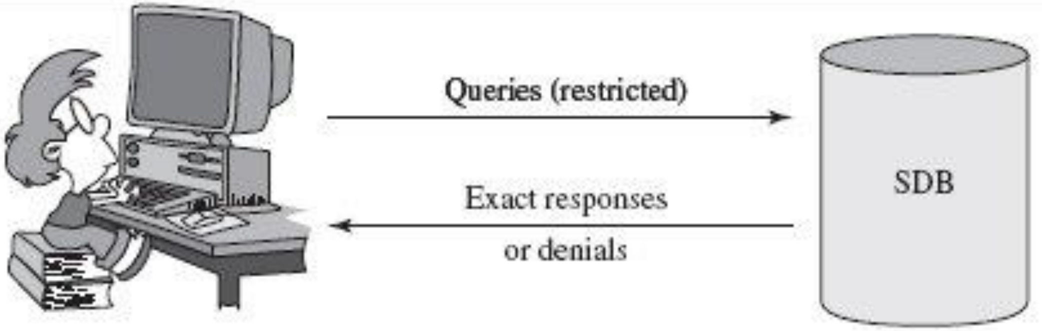
Query 2 Results

K

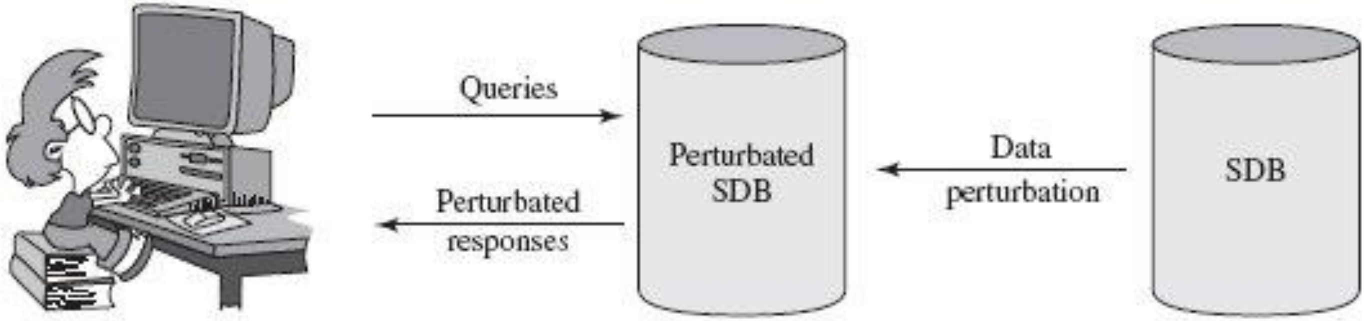
Query Results

NO

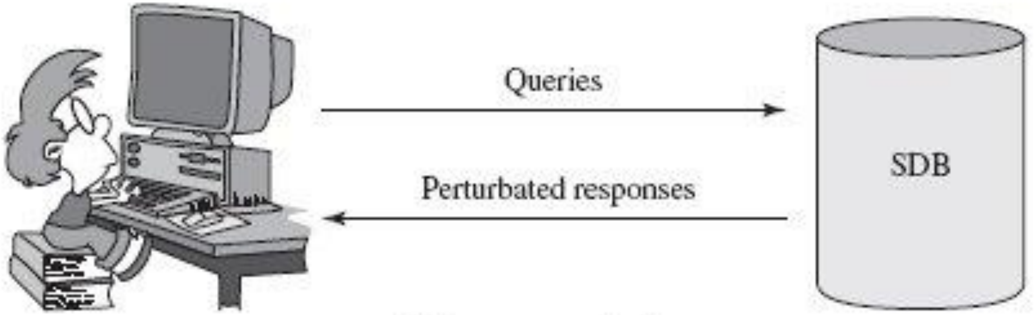
YES



(a) Query set restriction



(b) Data perturbation



(c) Output perturbation

Tracker (snooping tool)

- To illustrate the basic idea of a tracker, consider the following queries
- Q1: Count (Sex = Female) = A
- Q2: Count (Sex = Female OR

  (Age = 42 & Sex = Male & Employer = ABC)) = B

  If B = A+1

- Q3: Count (Sex = Female OR

  (Age = 42 & Sex = Male & Employer = ABC) &

  Diagnosis = Schizophrenia)

  Positively or negatively compromised!

Tracker (snooping tool)

- Suppose that the response to Q1 and Q2 are, respectively, A and B, where K <= A <= L – K and K <= B <= L – K.
- If B=A+1, then the target is uniquely identified by the clause "Age = 42 & Sex = Male and Employer = ABC."
- In this case, the database is positively compromised if the response to Q3 is B and negatively compromised if the response to Q3 is A.

Query Denial and Information Leakage

- A general problem with query restriction techniques is that the denial of a query may provide sufficient clues that an attacker can deduce underlying information. This is generally described by saying that query denial can leak information.

- Example: next slide

Query Denial and Information Leakage

- Suppose a database consists of real-valued entries and that a query is denied only if it would enable the requester to deduce a value.
- Now suppose the requester poses the query **sum**(*x*1, *x*2, *x*3) and the response is 15.
- Then the requester queries **max**(*x*1, *x*2, *x*3) and the query is denied. What can the requester deduce from this? We know that the **max**(*x*1, *x*2, *x*3) cannot be less than 5 because then the sum would be less than 15.
- But if **max**(*x*1, *x*2,*x*3) > 5, the query would not be denied because the answer would not reveal a specific value. Therefore, it must be the case that **max**(*x*1, *x*2, *x*3) = 5, which enables the requester to deduce that *x*1= *x*2 = *x*3 = 5.

Query set size control

- With query set size control the database can be easily compromised within a frame of 4-5 queries.
- For query set control, if the threshold value *k* is large, then it will restrict too many queries.
- And still does not guarantee protection from compromise

Query Set Overlap Control

- Basic idea: successive queries must be checked against the number of common records.
- If the number of common records in any query exceeds a given threshold, the requested statistic is not released.
- A query *q(C)* is only allowed if:

$$|X(C) \cap X(D)| \leq r, \quad r > 0$$

- If K denotes the minimum query set size and r denotes the maximum number of overlapping entities allowed between pairs of queries, number of queries needed for a compromise has a lower bound $1 + (K\text{-}1)/r$

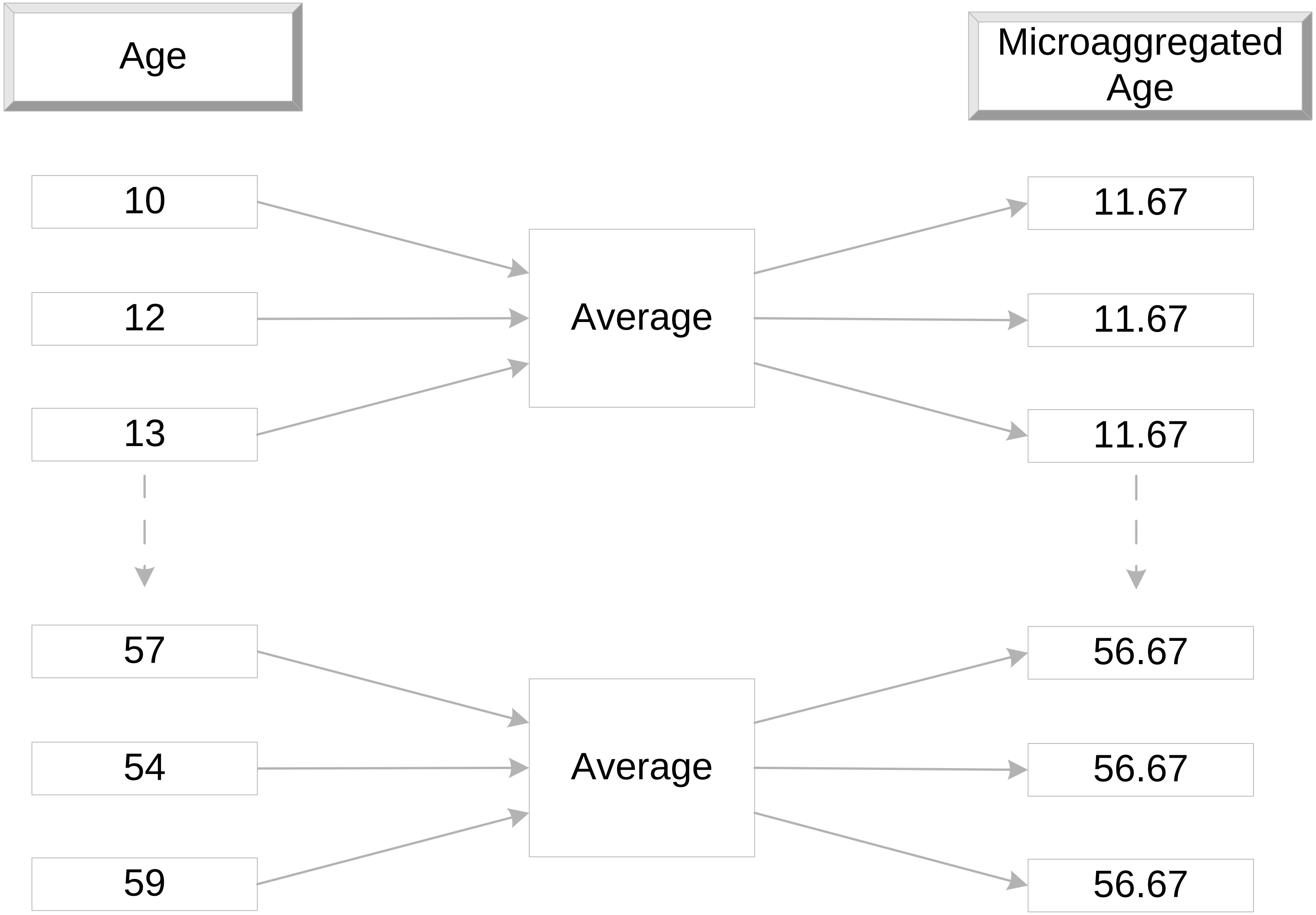
Query Set Overlap Control

- Drawback: High processing overhead – every new query compared with all previous ones

Microaggregation

- Raw (individual) data is grouped into small aggregates before publication
- The average value of the group replaces each value of the individual
- Data with the most similarities are grouped together to maintain data accuracy
- Helps to prevent disclosure of individual data

Microaggregation



Age

Microaggregated Age

10

12

13

Average

11.67

11.67

11.67

57

54

59

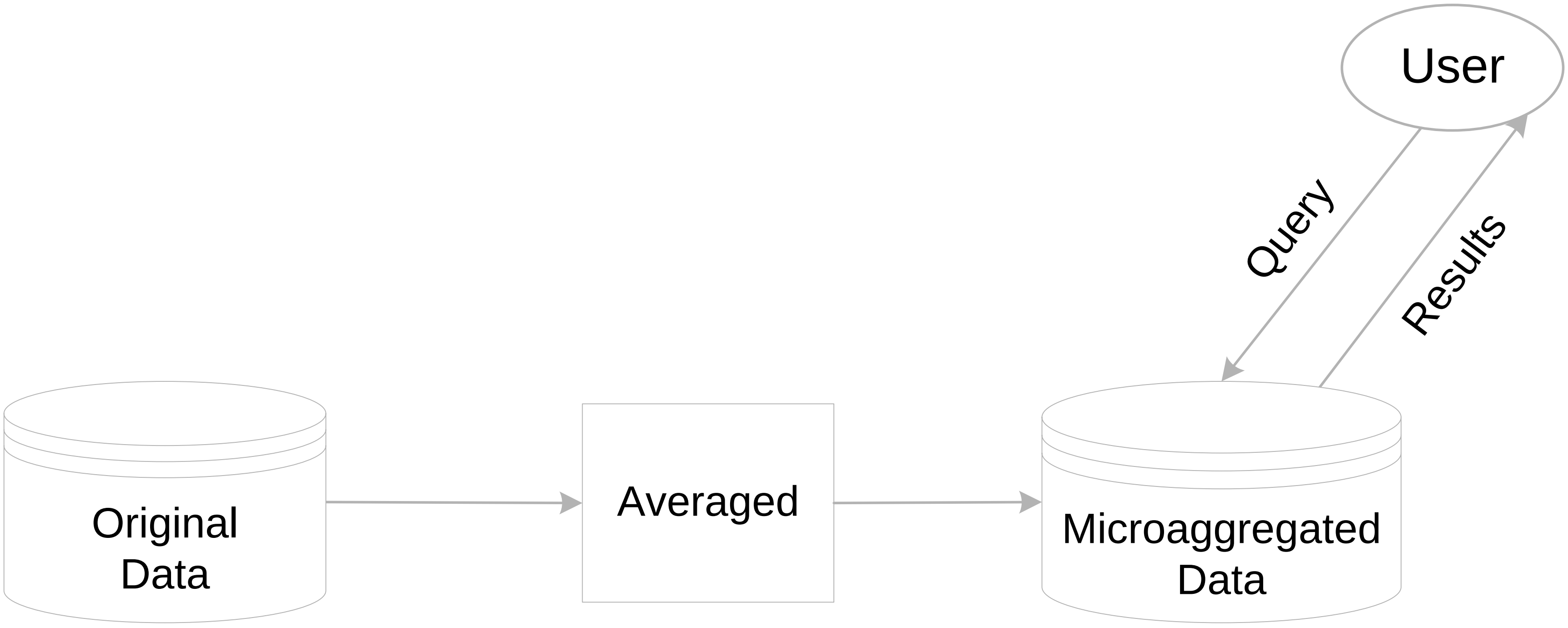
Average

56.67

56.67

56.67

Microaggregation



Data Perturbation

- Provides answers to all queries, but the answers are approximate.

- For larger databases, a simpler and more effective technique is to, in effect, add noise to the statistics generated from the original data.

- Perturbed data is raw data with noise added

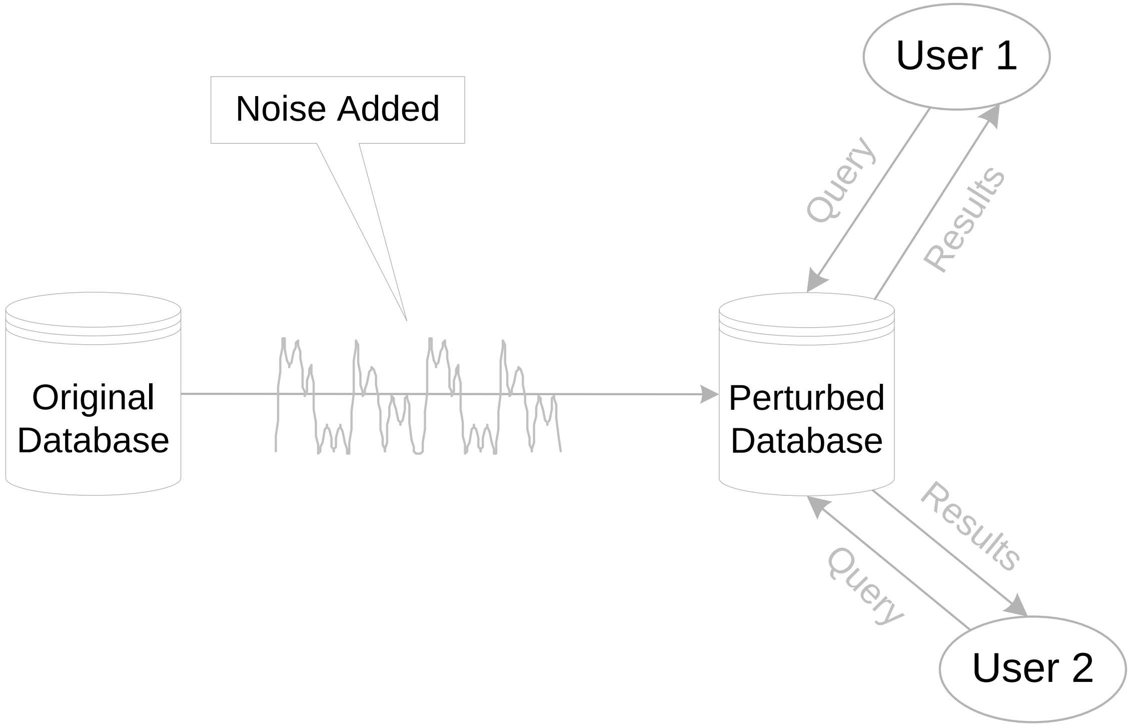
- **Pro**: With perturbed databases, if unauthorized data is accessed, the true value is not disclosed

- **Con**: Data perturbation runs the risk of presenting biased data

- Can be input perturbation and output perturbation

Data Perturbation : Input perturbtion

This can be done in one of two ways .The data in the SDB can be modified (perturbed) so as to produce statistics that cannot be used to infer values for individual records; we refer to this as **data perturbation**.

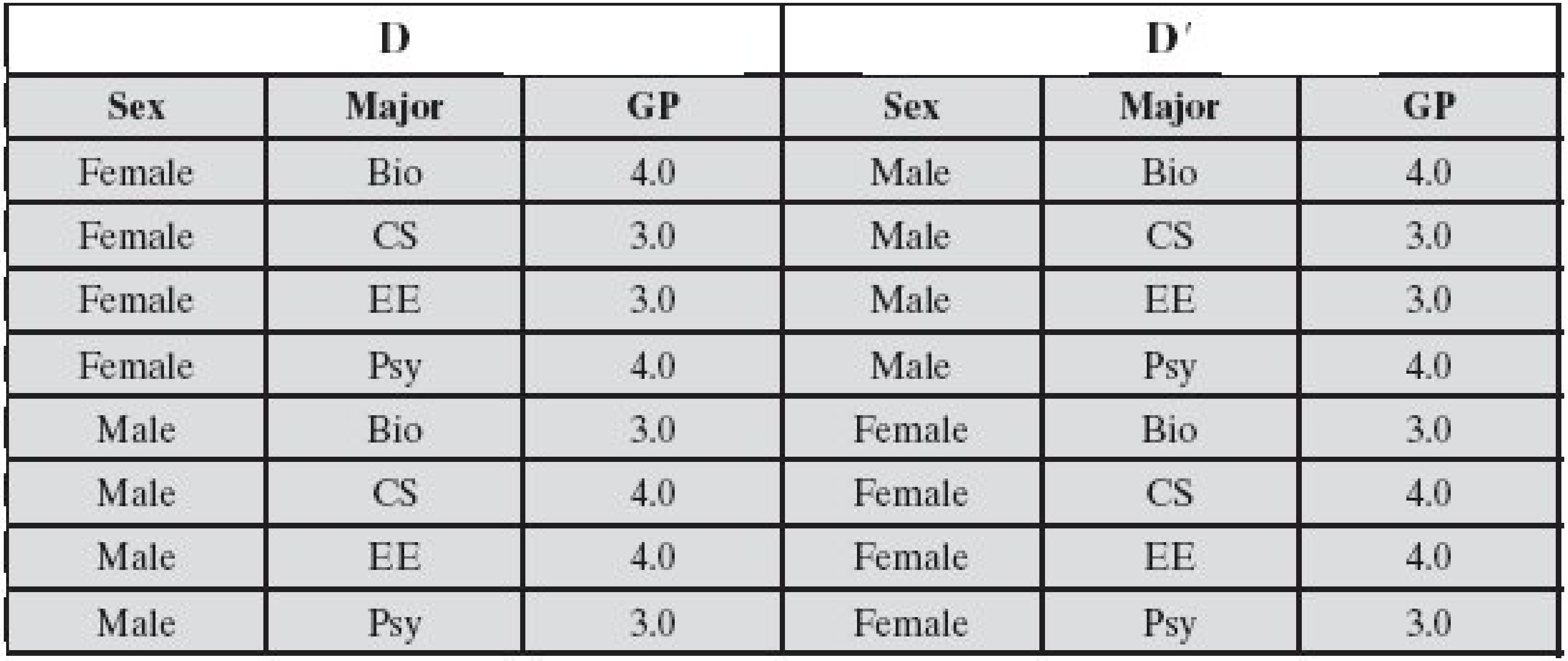


Data Perturbation (Input perturbation) Techniques

- The method is referred to as **data swapping**. In this method, attribute values are exchanged (swapped) between records in sufficient quantity so that nothing can be deduced from the disclosure of individual records.

Data Perturbation (Input perturbation) Techniques

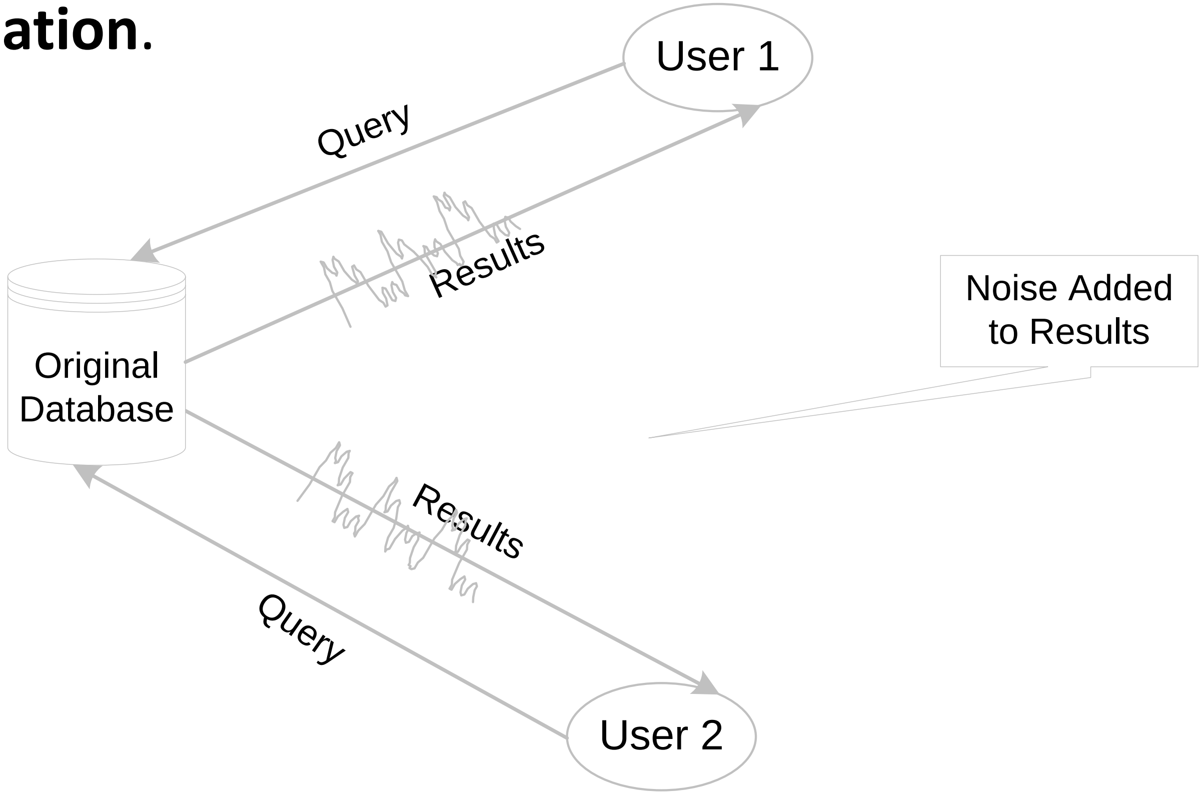
- The table shows a simple example, transforming the database D into the database D’. The transformed database D’ has the same statistics as D for statistics derived from one or two attributes. However, three-attribute statistics are not preserved. For example, **count**(Female, CS, 3.0) has the value 1 in D but the value 0 in D’.



D			D′		
Sex	Major	GP	Sex	Major	GP
Female	Bio	4.0	Male	Bio	4.0
Female	CS	3.0	Male	CS	3.0
Female	EE	3.0	Male	EE	3.0
Female	Psy	4.0	Male	Psy	4.0
Male	Bio	3.0	Female	Bio	3.0
Male	CS	4.0	Female	CS	4.0
Male	EE	4.0	Female	EE	4.0
Male	Psy	3.0	Female	Psy	3.0

Output Perturbation

Alternatively, when a statistical query is made, the system can generate statistics that are modified from those that the original database would provide, again thwarting attempts to gain knowledge of individual records; this is referred to as **output perturbation**.



Output Perturbation Technique

- The technique works as follows:
- A user issues a query *q*(*C*) that is to return a statistical value. The query set so defined is *X*(*C*).
- The system replaces *X*(*C*) with a sampled query set, which is a properly selected subset of *X*(*C*).
- The system calculates the requested statistic on the sampled query set and returns the value.

Limitations of Perturbation Techniques

- The main challenge in the use of perturbation techniques is to determine the average size of the error to be used. If there is too little error, a user can infer close approximations to protected values. If the error is, on average, too great, the resulting statistics may be unusable.

- For a small database, it is difficult to add sufficient perturbation to hide data without badly distorting the results. Fortunately, as the size of the database grows, the effectiveness of perturbation techniques increases.

Auditing

- Keeping up-to-date logs of all queries made by each user and check for possible compromise when a new query is issued.
  - ▪ Usually done with up-to-date logs.
  - ▪ Each time a user issues a query, the log is checked to see if the user is querying the database maliciously.
- One of the major drawbacks of auditing, however, is its excessive CPU time and storage requirements to store and process the accumulated logs.

Partitioning

- With partitioning, the records in the database are clustered into a number of mutually exclusive groups.

- The user may only query the statistical properties of each group as a whole. That is, the user may not select a subset of a group.

- Thus, with multiple queries, there must either be complete overlap (two different queries of all the records in a group) or zero overlap (two queries from different groups).

Partitioning

- The rules for partitioning the database are as follows:

  1) Each group *G* has *g* = |*G*| records, where *g* = 0 or *g*≥ *n*, and *g* even, where *n* is a fixed integer parameter.

  2) Records are added or deleted from *G* in pairs.

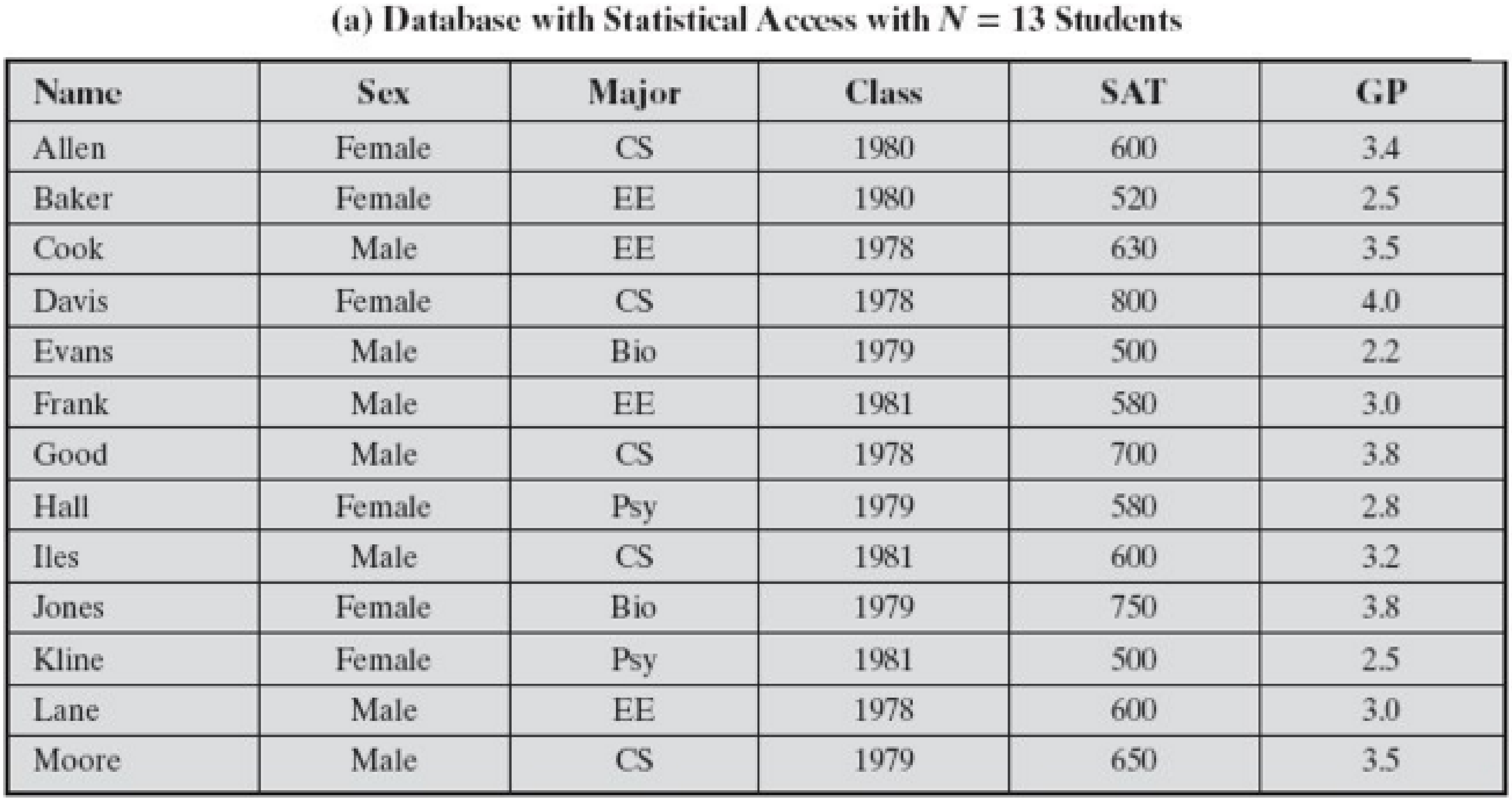
  3) Query sets must include entire groups. A query set may be a single group or multiple groups.

Partitioning

- A group of a single record is forbidden, for obvious reasons.
- The insertion or deletion of a single record enables a user to gain information about that record by taking before and after statistics.
- As an example, the database of Table 1 can be partitioned as shown in Table 2. Because the database has an odd number of records, the record for Kline has been omitted. The database is partitioned by year and sex, except that for 1978, it is necessary to merge the Female and Male records to satisfy the design requirement.

Partitioning

Table 1

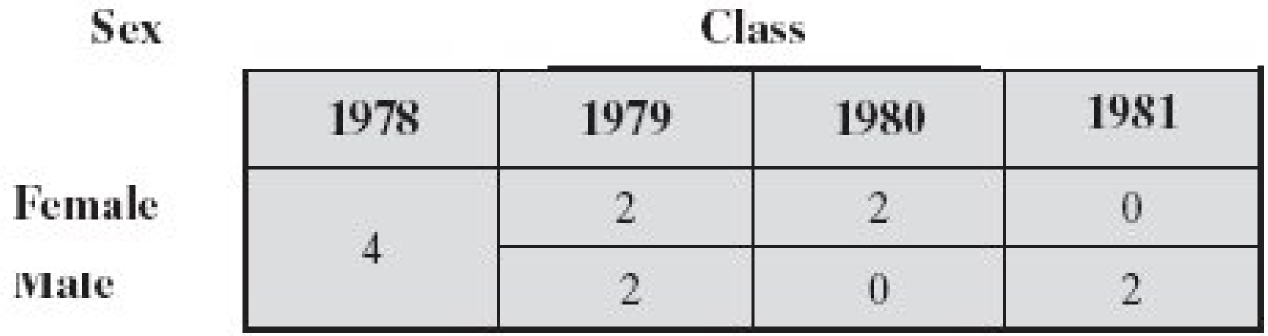


(a) Database with Statistical Access with $N = 13$ Students

Name	Sex	Major	Class	SAT	GP
Allen	Female	CS	1980	600	3.4
Baker	Female	EE	1980	520	2.5
Cook	Male	EE	1978	630	3.5
Davis	Female	CS	1978	800	4.0
Evans	Male	Bio	1979	500	2.2
Frank	Male	EE	1981	580	3.0
Good	Male	CS	1978	700	3.8
Hall	Female	Psy	1979	580	2.8
Iles	Male	CS	1981	600	3.2
Jones	Female	Bio	1979	750	3.8
Kline	Female	Psy	1981	500	2.5
Lane	Male	EE	1978	600	3.0
Moore	Male	CS	1979	650	3.5

Partitioning

Table 2



Sex	Class			
	1978	1979	1980	1981
Female	4	2	2	0
Male		2	0	2

Some Comparisons

Method	Security	Richness of Information	Costs
Query-set-size control	Low	Low[1]	Low
Query-set-overlap control	Low	Low	High
Auditing	Moderate-Low	Moderate	High
Partitioning	Moderate-low	Moderate-low	Low
Microaggregation	Moderate	Moderate	Moderate
Data Perturbation	High	High-Moderate	Low
Varying Output Perturbation	Moderate	Moderate-low	Low
Sampling	Moderate	Moderate-Low	Moderate