Improved Reconstruction Attacks on Encrypted Data Using Range Query Leakage

Marie-Sarah Lacharité, **Brice Minaud**, Kenny Paterson

Information Security Group

ROYAL HOLLOWAY UNIVERSITY OF LONDON

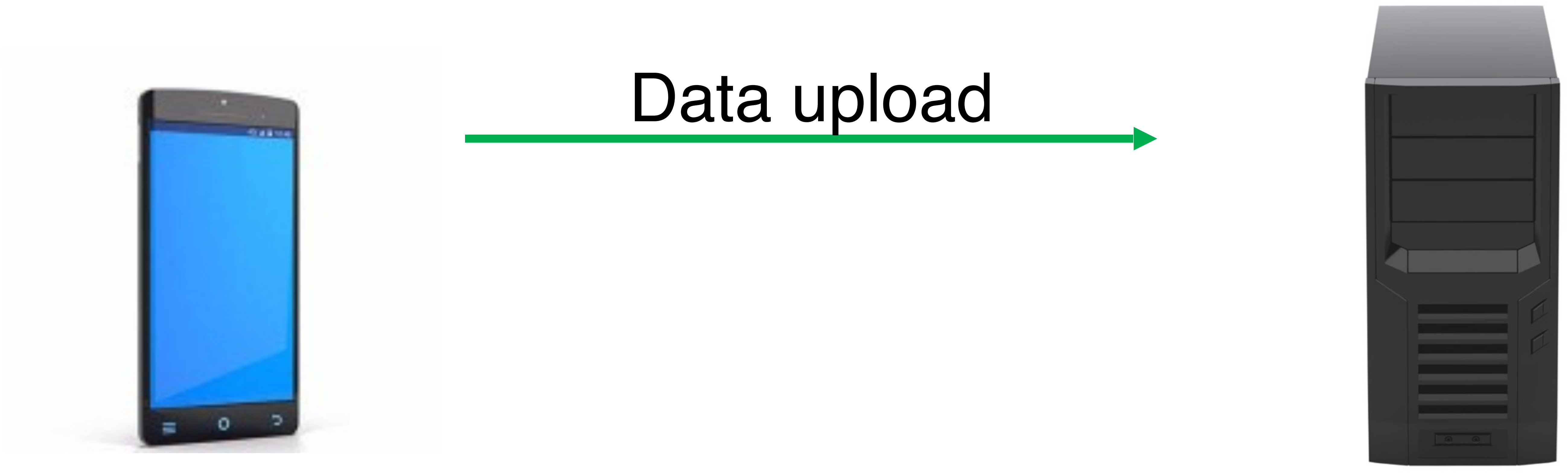
IEEE Symposium on Security and Privacy, May 21, 2018

Outsourcing Data with Search Capabilities

Client

Server

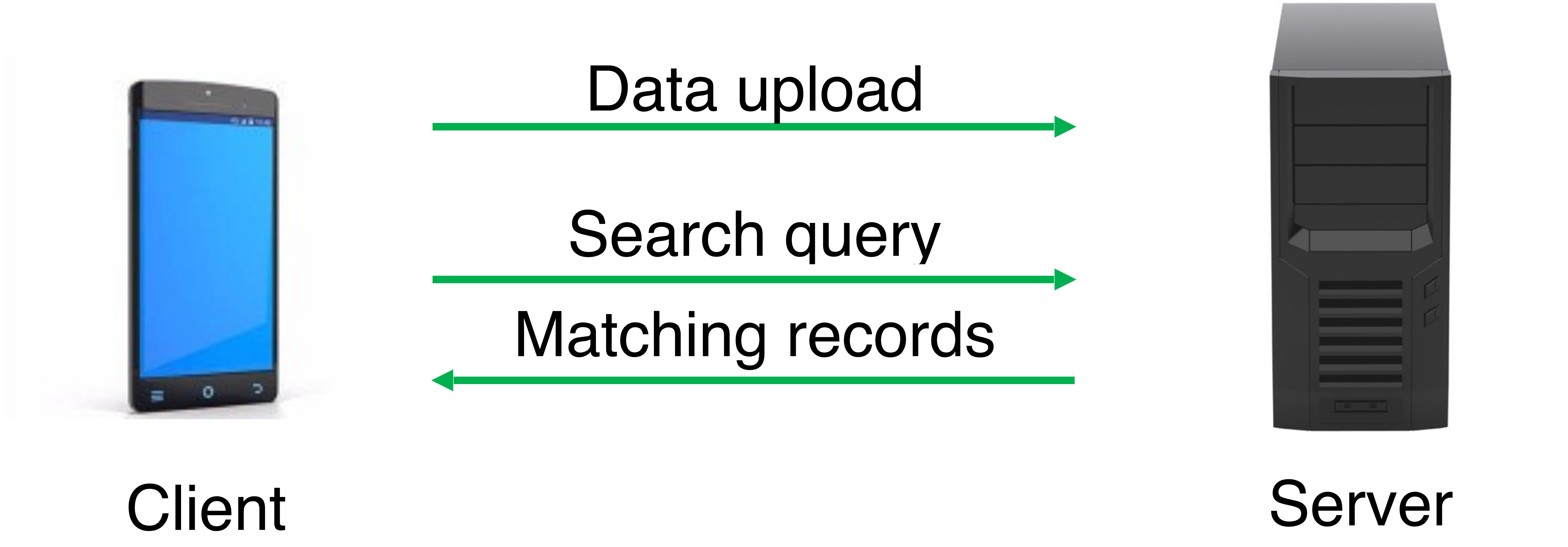
Outsourcing Data with Search Capabilities



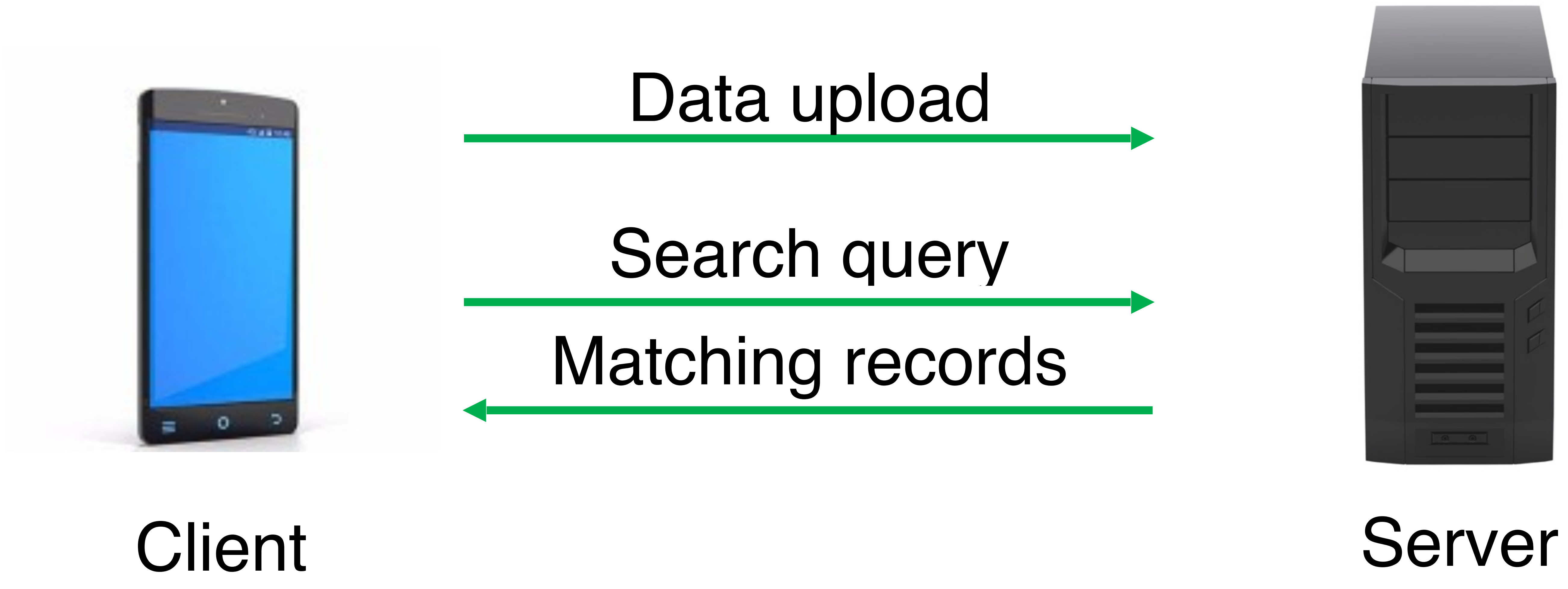
Client

Server

Outsourcing Data with Search Capabilities



Outsourcing Data with Search Capabilities



For an **encrypted database management system**:

- Data = collection of records in a database. *e.g. health records.*
- Search query examples:
  - find records with given value. *e.g. patients aged 57.*
  - find records within a given range. *e.g. patients aged 55-65.*

Security of Data Outsourcing Solutions

Adversaries:

- **Snapshot**: breaks into server, gets snapshot of memory.
- **Persistent**: corrupts server, sees all communication transcripts.
  Can be server itself.

Security goal = privacy.

→ Adversary learns as little as possible about the client's data and queries.

Solutions

- **Structure-preserving encryption.**
  Vulnerable to snapshot attackers.

Solutions

- **Structure-preserving encryption.**
  Vulnerable to snapshot attackers.

- **Second-generation schemes**:
  Aim to protect against snapshot and persistent attackers.

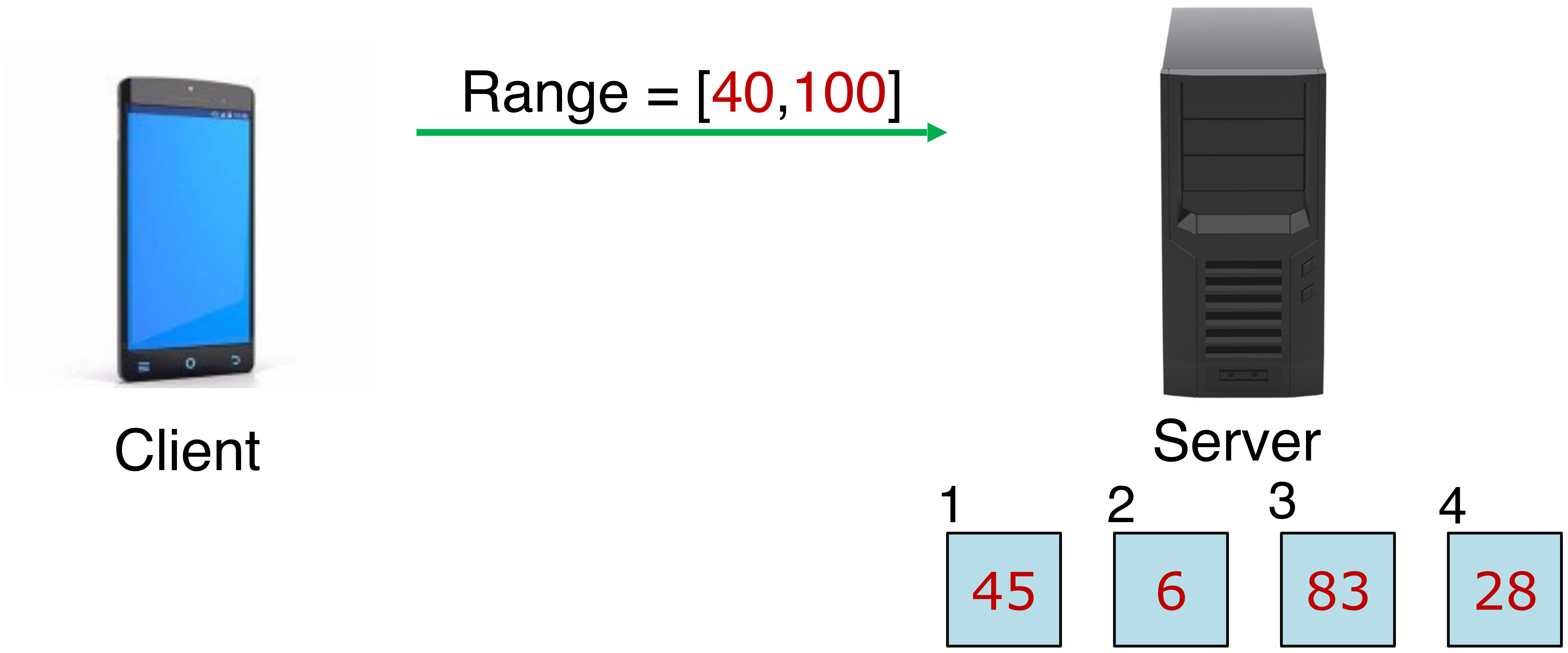
Solutions

- **Structure-preserving encryption.**
  Vulnerable to snapshot attackers.

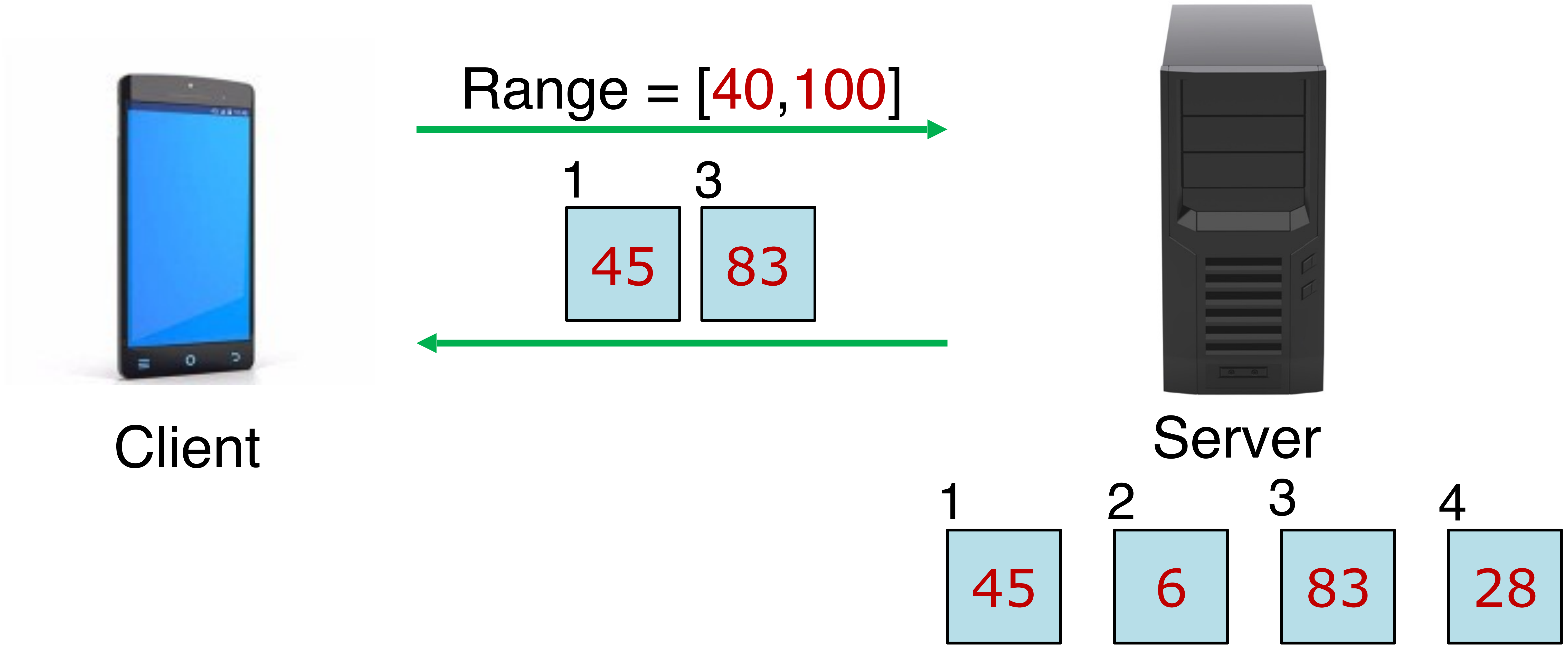
- **Second-generation schemes**:
  Aim to protect against snapshot and persistent attackers.

- **Very active research topic.**
  [AKSX04], [BCLO09], [PKV+14], [BLR+15], [NKW15], [KKNO16], [LW16], [FVY+17], [SDY+17], [DP17], [HLK18], [PVC18], [MPC+18]…

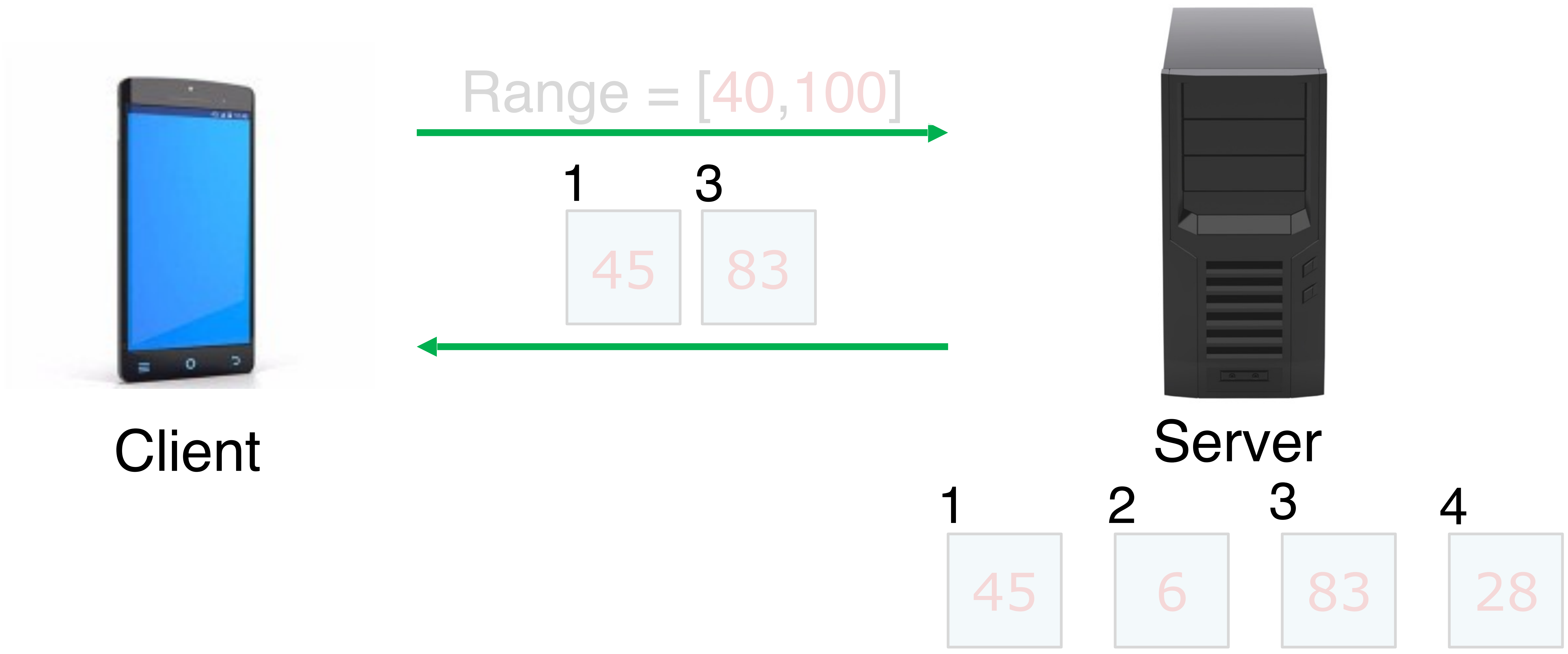
Schemes Supporting Range Queries



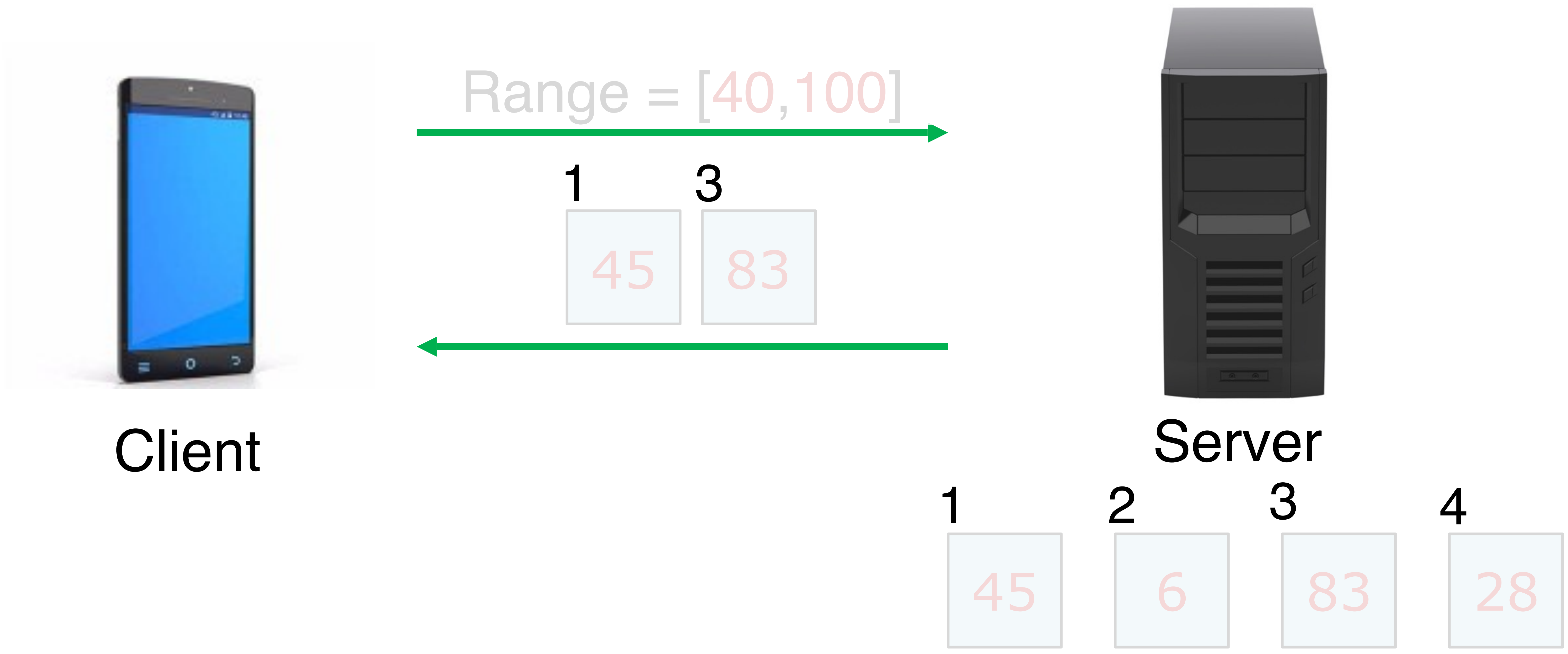
Schemes Supporting Range Queries



Schemes Supporting Range Queries



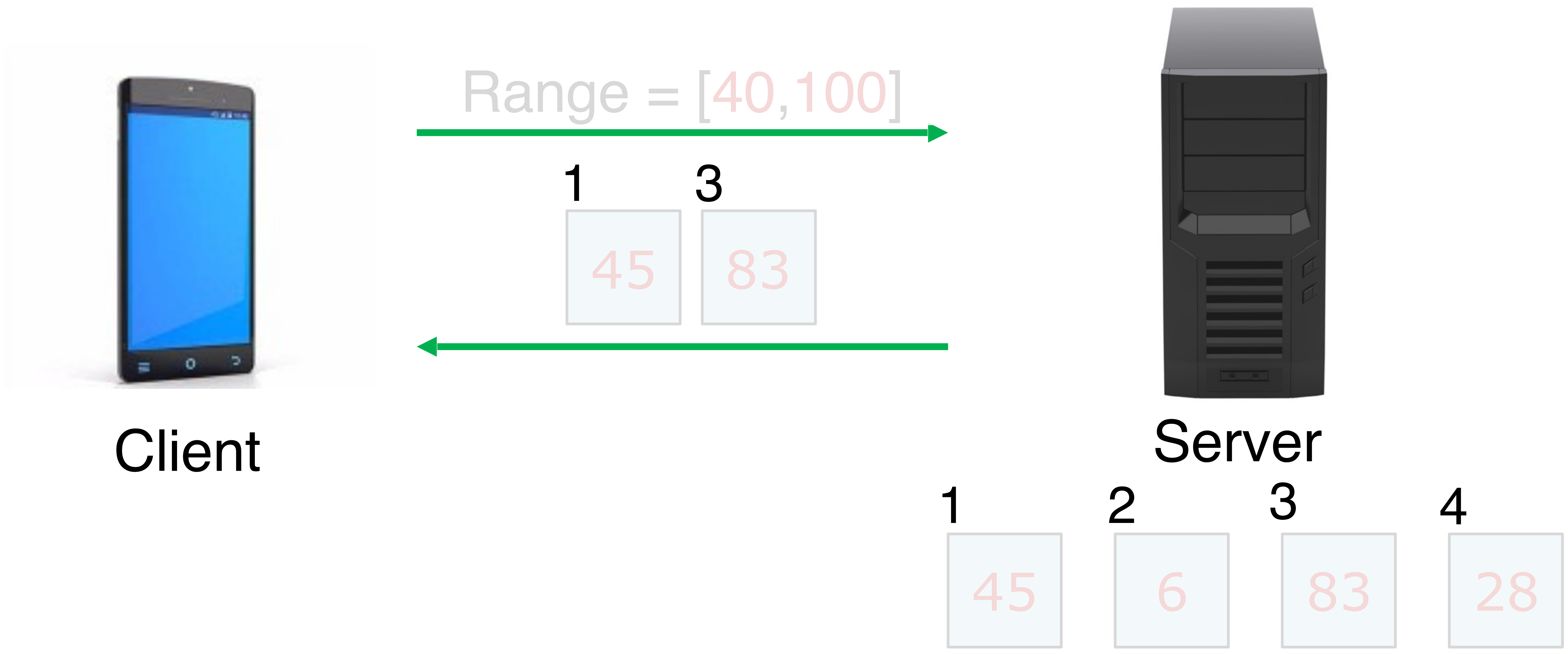
Schemes Supporting Range Queries



- Most schemes leak set of matching records = **access pattern** leakage.

  OPE, ORE schemes, POPE, [HK16], BlindSeer, [Lu12], [FJ+15], …

Schemes Supporting Range Queries



- Most schemes leak set of matching records = **access pattern** leakage.

  OPE, ORE schemes, POPE, [HK16], BlindSeer, [Lu12], [FJ+15], …

- Some schemes also leak #records below queried endpoints = **rank** leakage.

  FH-OPE, Lewi-Wu, Arx, Cipherbase, EncKV, …

Exploiting Leakage

- Most schemes prove that nothing more leaks than their leakage model allows.

  For example, leakage = access pattern + rank.

  What can we really learn from this leakage?

Exploiting Leakage

- Most schemes prove that nothing more leaks than their leakage model allows.

  For example, leakage = access pattern + rank.

  What can we really learn from this leakage?

- **Our goal**: **full reconstruction** = recovering the exact value of every record.

Exploiting Leakage

- Most schemes prove that nothing more leaks than their leakage model allows.

  For example, leakage = access pattern + rank.

  What can we really learn from this leakage?

- **Our goal**: **full reconstruction** = recovering the exact value of every record.

- **[KKNO16]**: O(N^2 log N) queries suffice for full reconstruction using only access pattern leakage!

  - where N is the number of possible values (e.g. 125 for age in years).

Assumptions for our Analysis

- Data is **dense:** all values appear in at least one record.

- Queries are **uniformly distributed**.

  Our algorithms don't actually care though – the assumption is for computing data upper bounds.

Our Main Results

- **Full reconstruction** with $O(N \cdot \log N)$ queries from access pattern leakage
  - in fact, $N \cdot (3 + \log N)$.

Our Main Results

- **Full reconstruction** with $O(N \cdot \log N)$ queries from access pattern leakage
  - in fact, $N \cdot (3 + \log N)$.

- **Approximate reconstruction** with relative accuracy ε with $O(N \cdot (\log 1/\varepsilon))$ queries.

Our Main Results

- **Full reconstruction** with $O(N \cdot \log N)$ queries from access pattern leakage
  - in fact, $N \cdot (3 + \log N)$.

- **Approximate reconstruction** with relative accuracy ε with $O(N \cdot (\log 1/\varepsilon))$ queries.

- **Approximate reconstruction** using an *auxiliary distribution* and access pattern + rank leakage.

Our Main Results

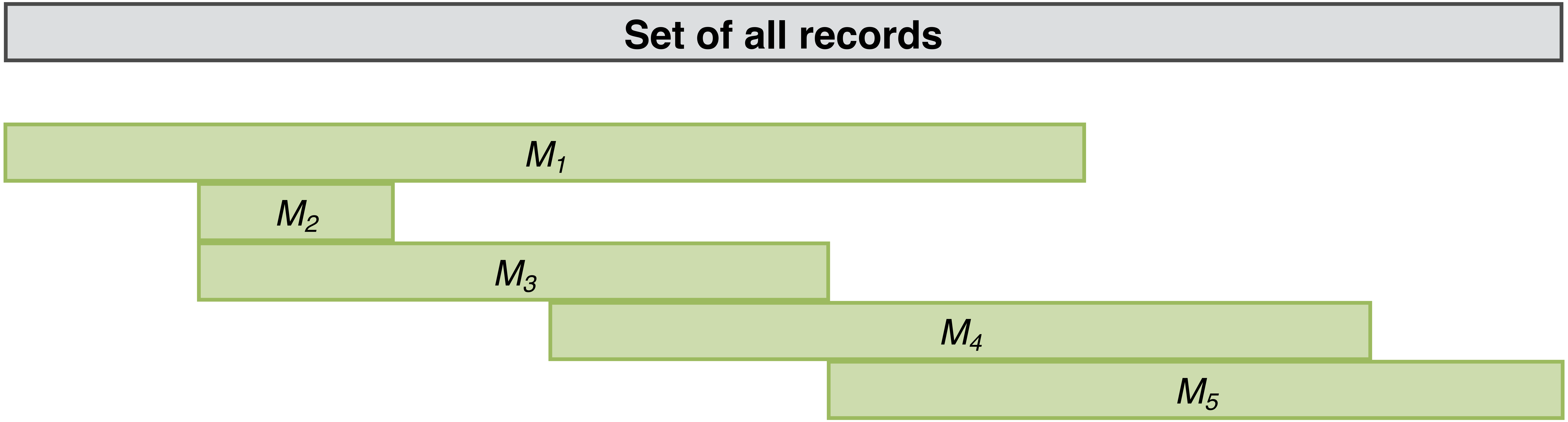
- **Full reconstruction** with $O(N \cdot \log N)$ queries from access pattern leakage
  – in fact, $N \cdot (3 + \log N)$.

- **Approximate reconstruction** with relative accuracy ε with $O(N \cdot (\log 1/\varepsilon))$ queries.

- **Approximate reconstruction** using an *auxiliary distribution* and access pattern + rank leakage.

Full reconstruction

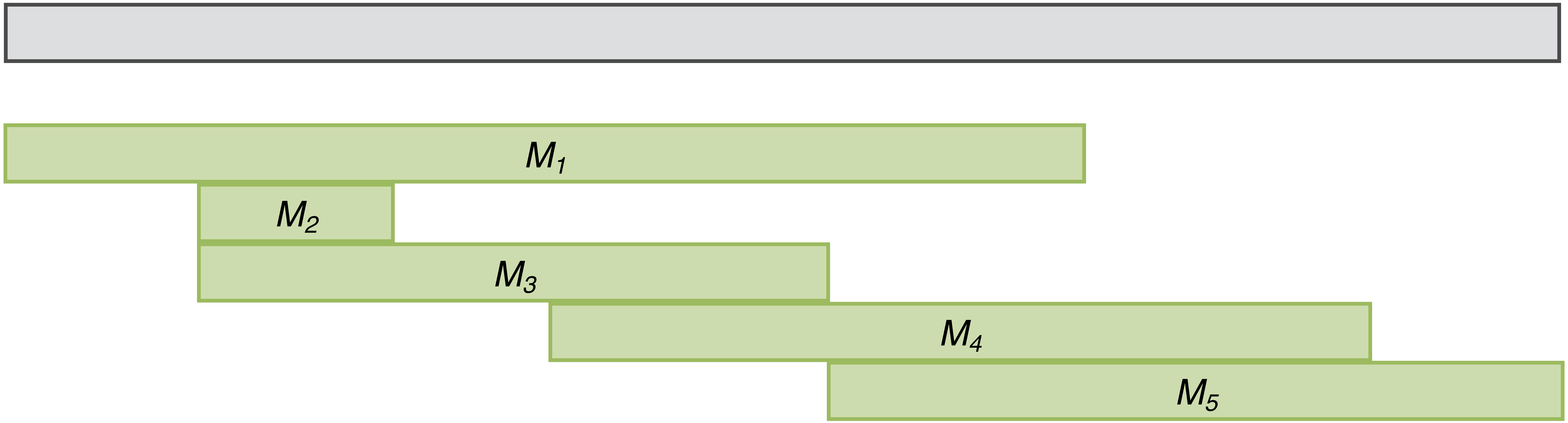
Full Reconstruction Algorithm



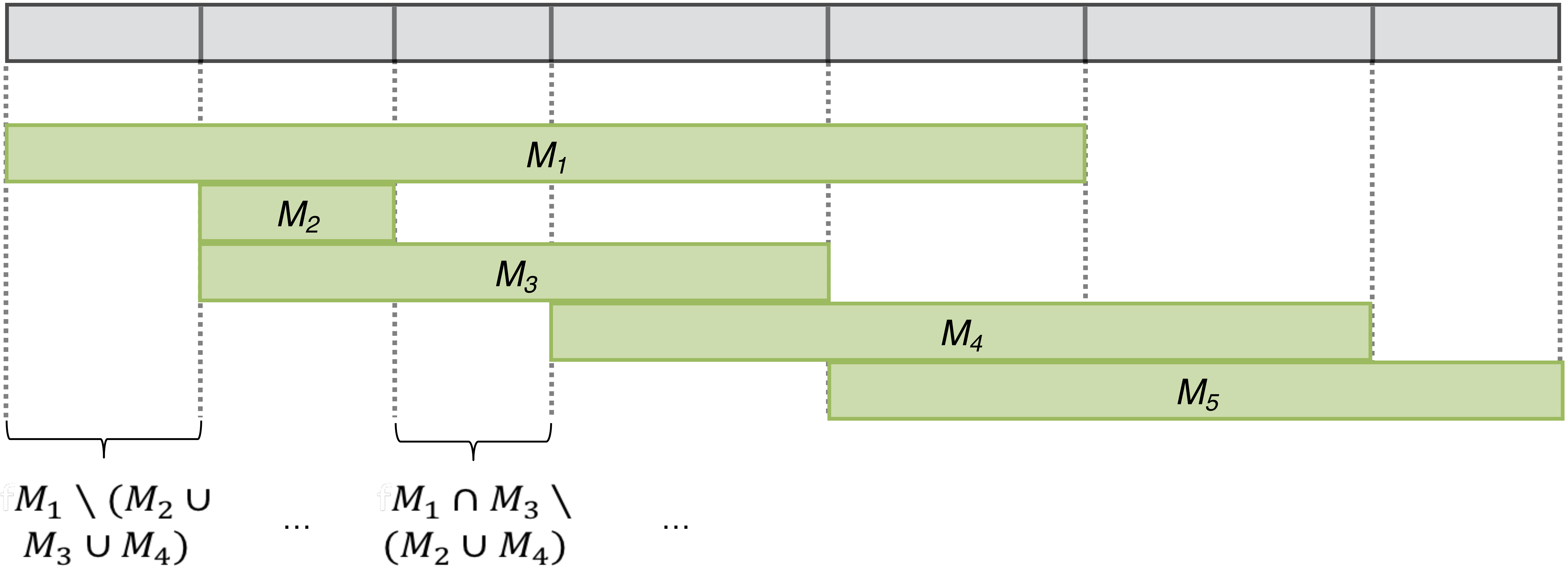
Assume $N = 7$ values, and 5 queries.

M_i = set of records matched by i-th query.

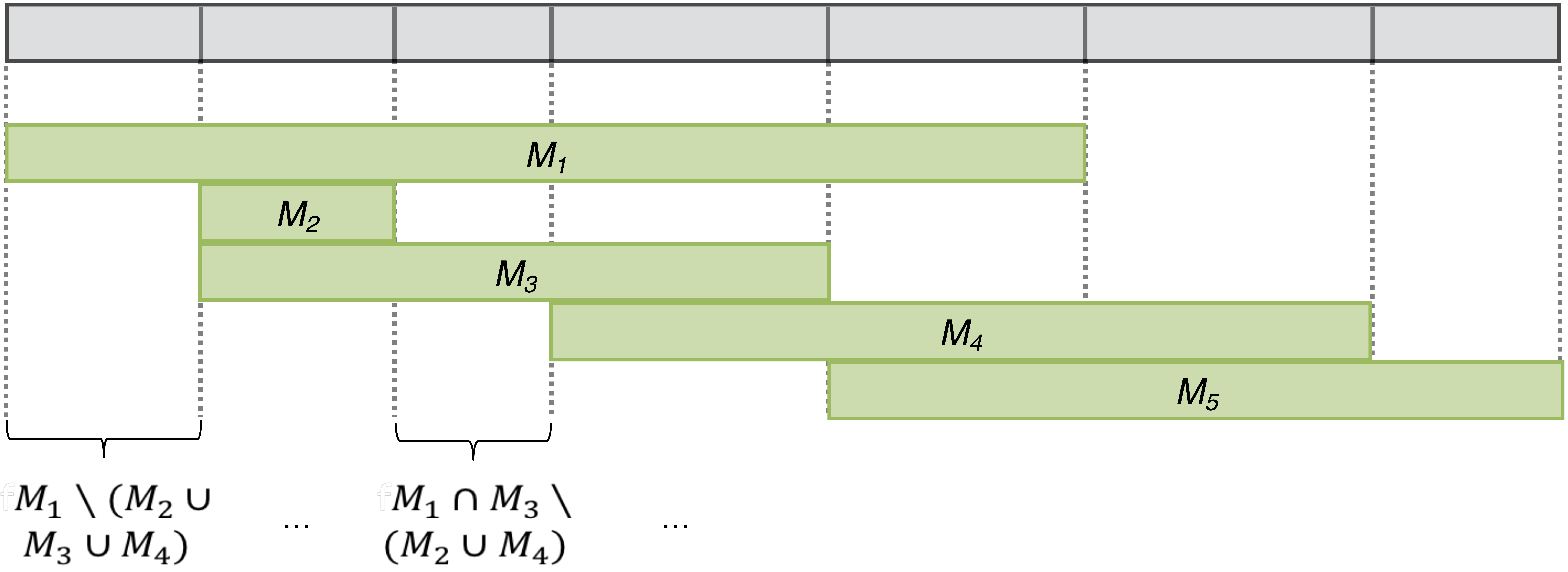
Step 1: Partitioning



Step 1: Partitioning

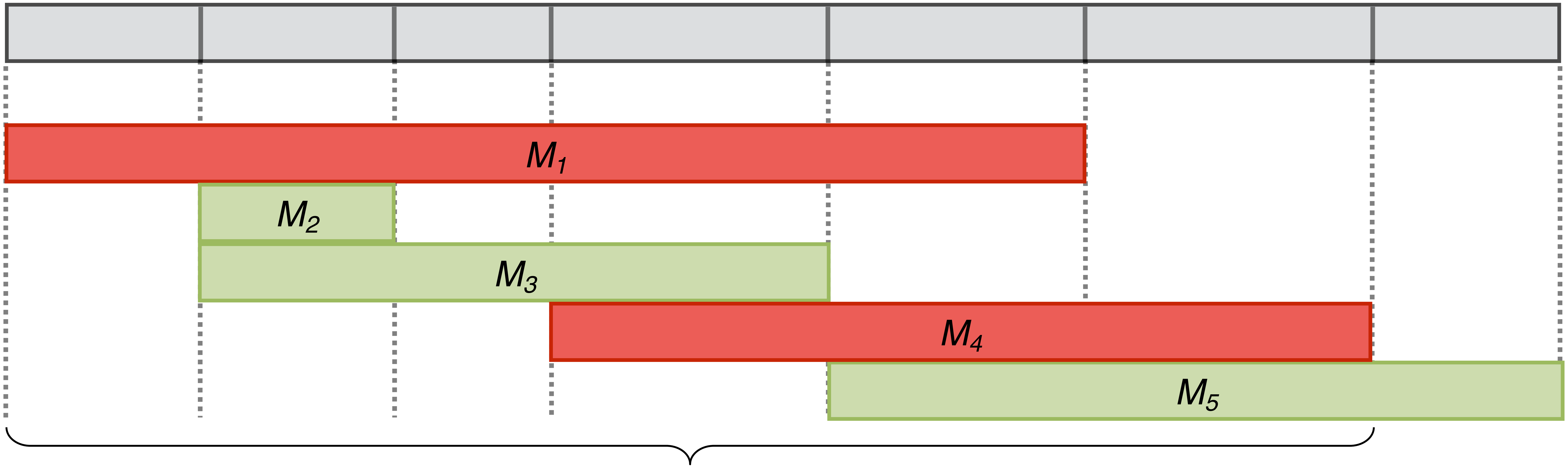


Step 1: Partitioning



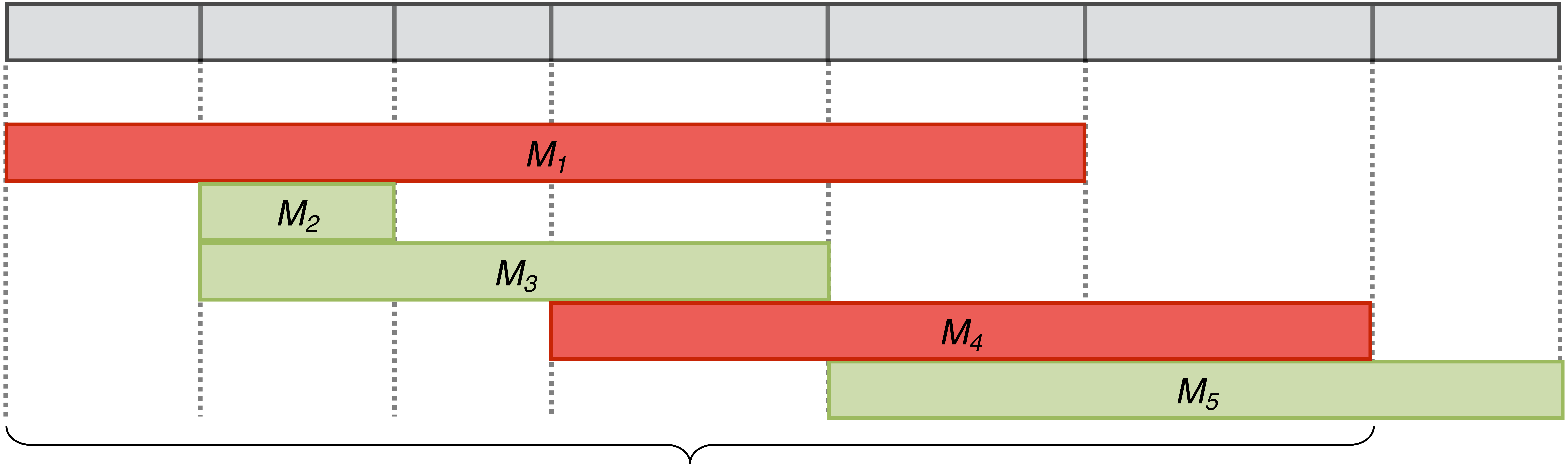
If there are N minimal subsets → each of them correspond to a single value.

Step 2a: Finding an Endpoint



$M_1 \cup M_3$ cover all but 1 minimal set

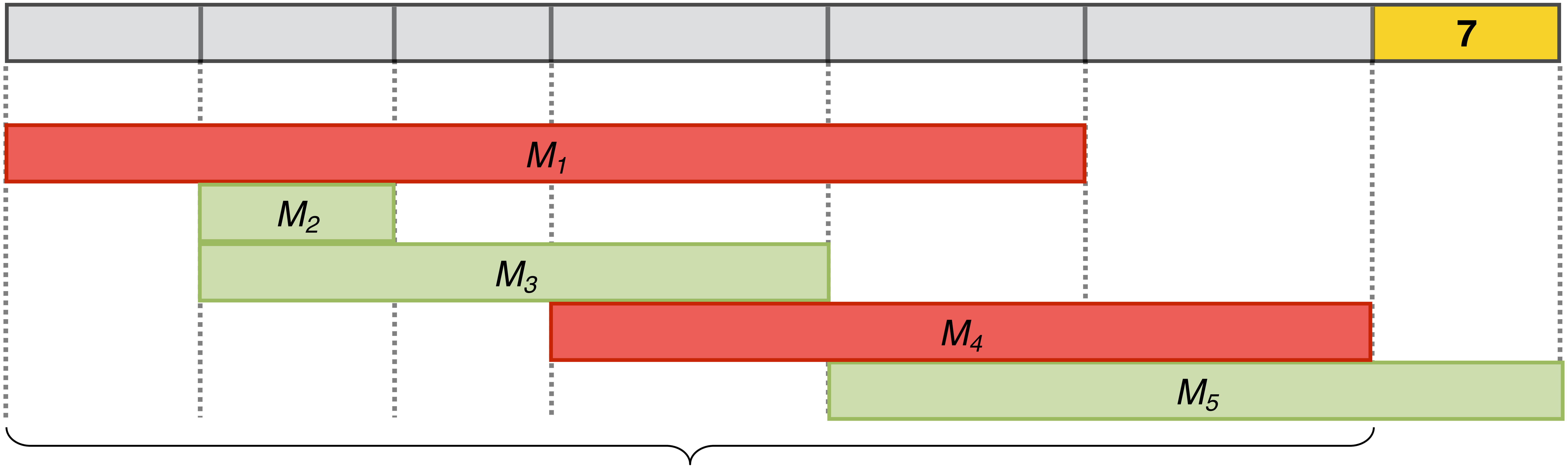
Step 2a: Finding an Endpoint



$M_1 \cup M_3$ cover all but 1 minimal set

Endpoint!

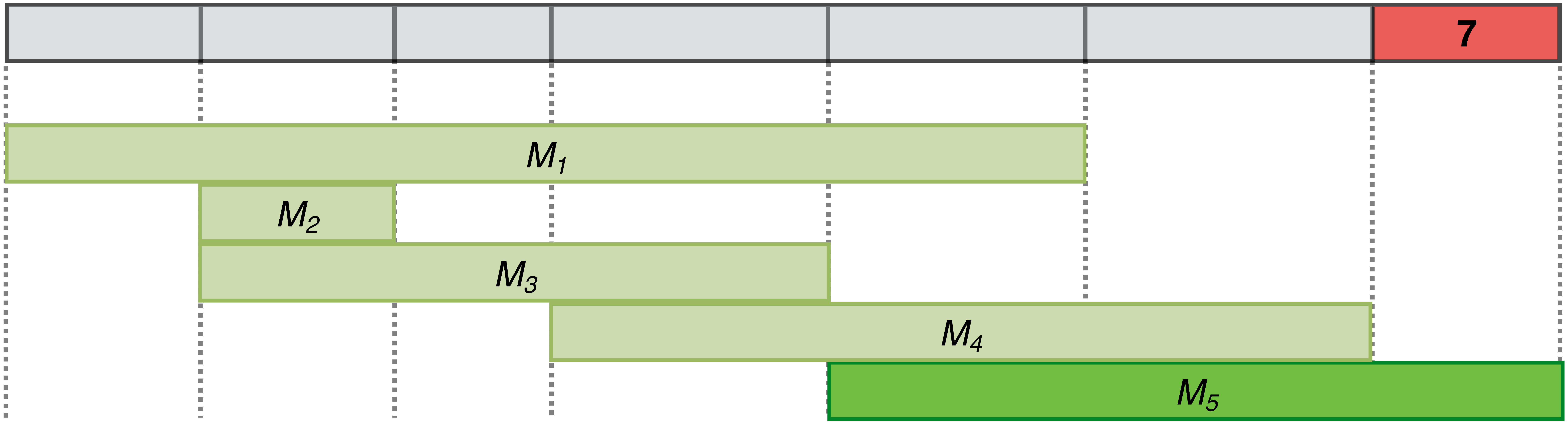
Step 2a: Finding an Endpoint



$M_1 \cup M_3$ cover all but 1 minimal set

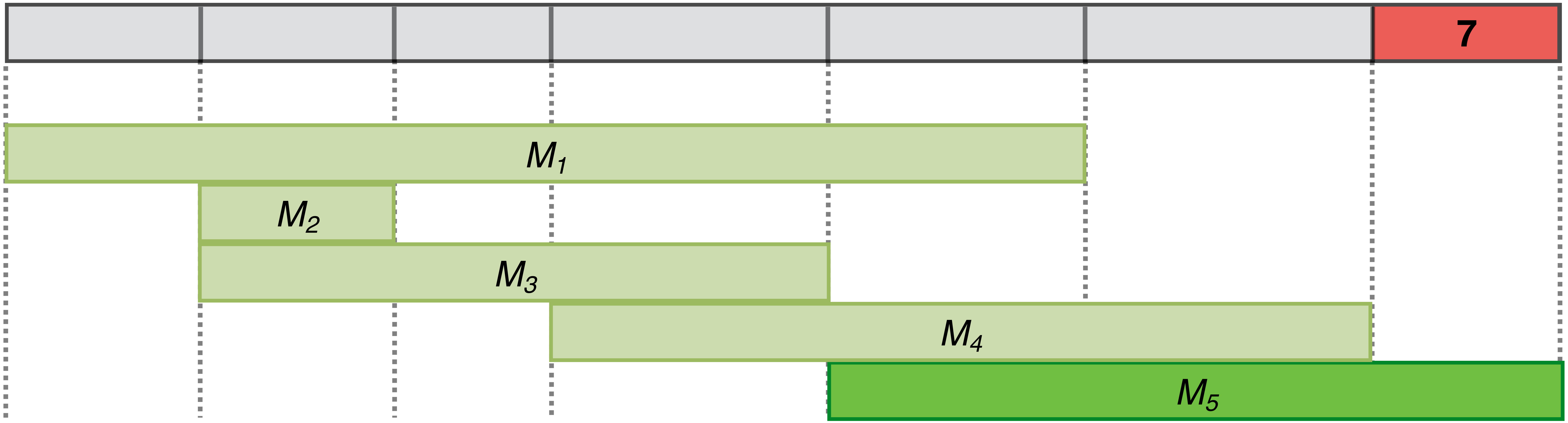
Endpoint!

Step 2b: Propagating



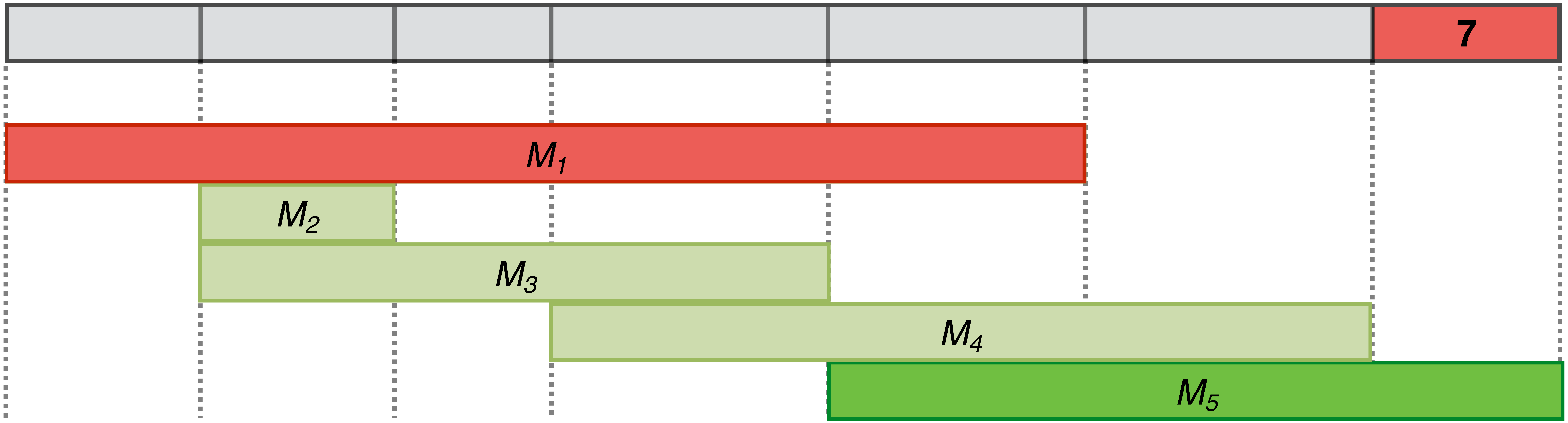
- Intersect

Step 2b: Propagating



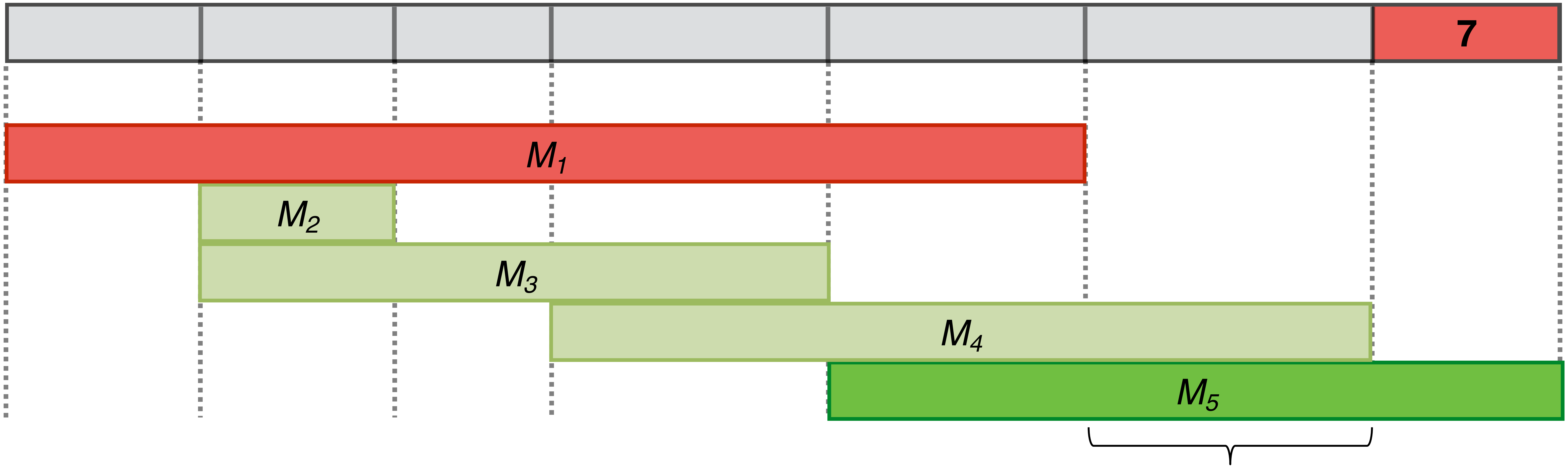
- Intersect
- Trim

Step 2b: Propagating



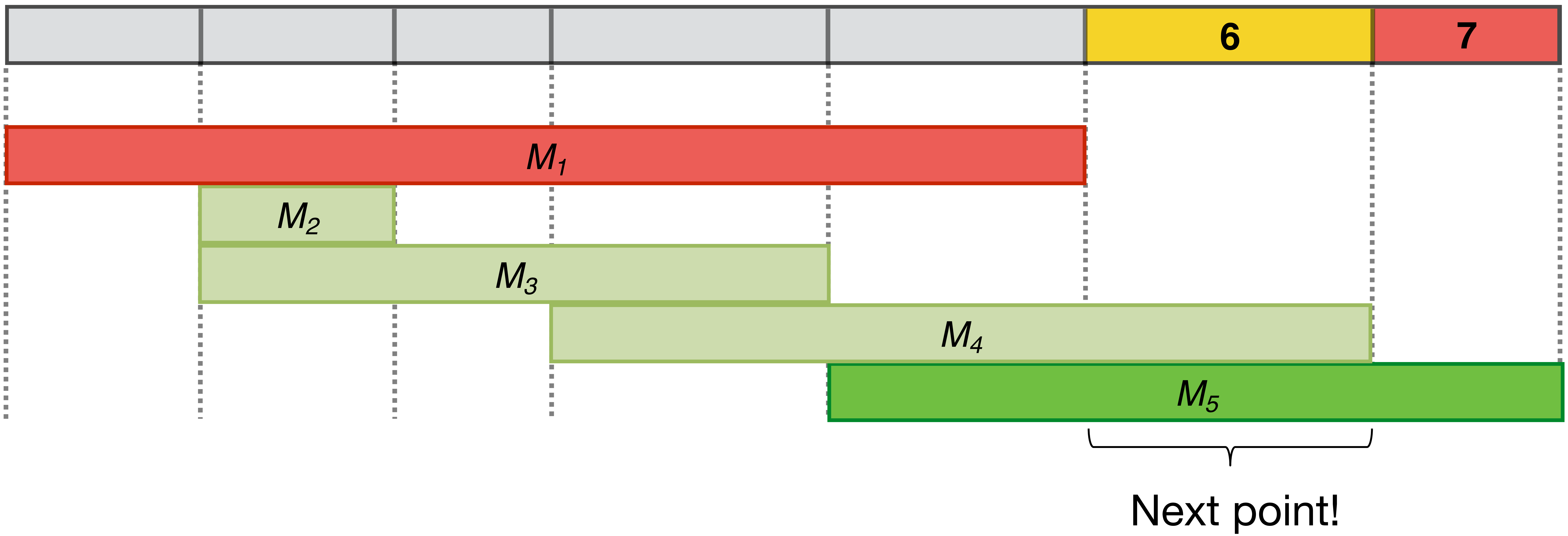
- Intersect
- Trim

Step 2b: Propagating



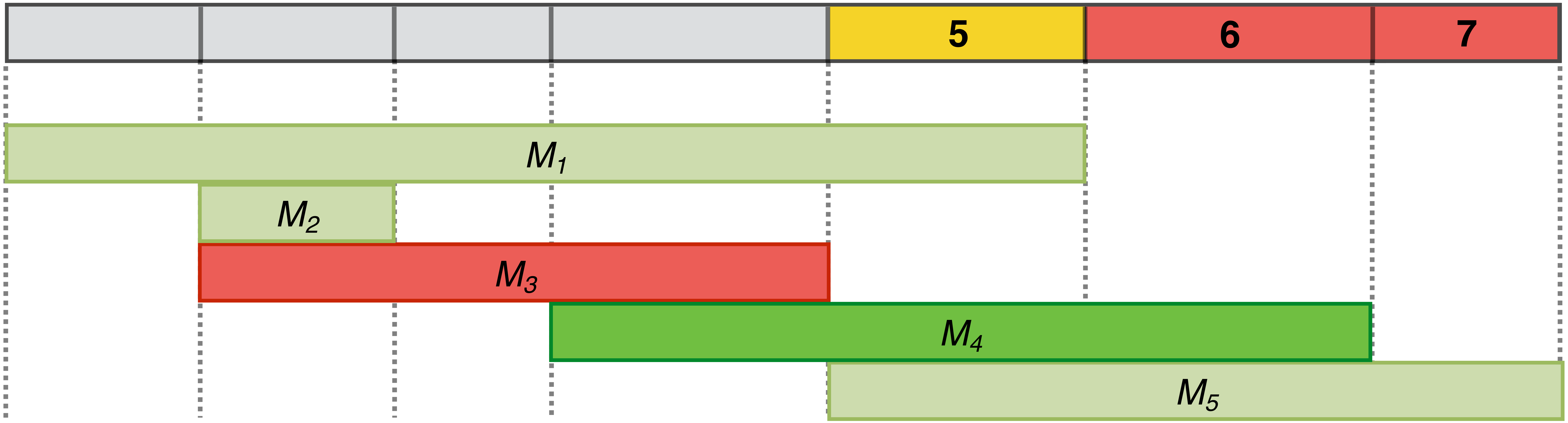
- Intersect
- Trim

Step 2b: Propagating



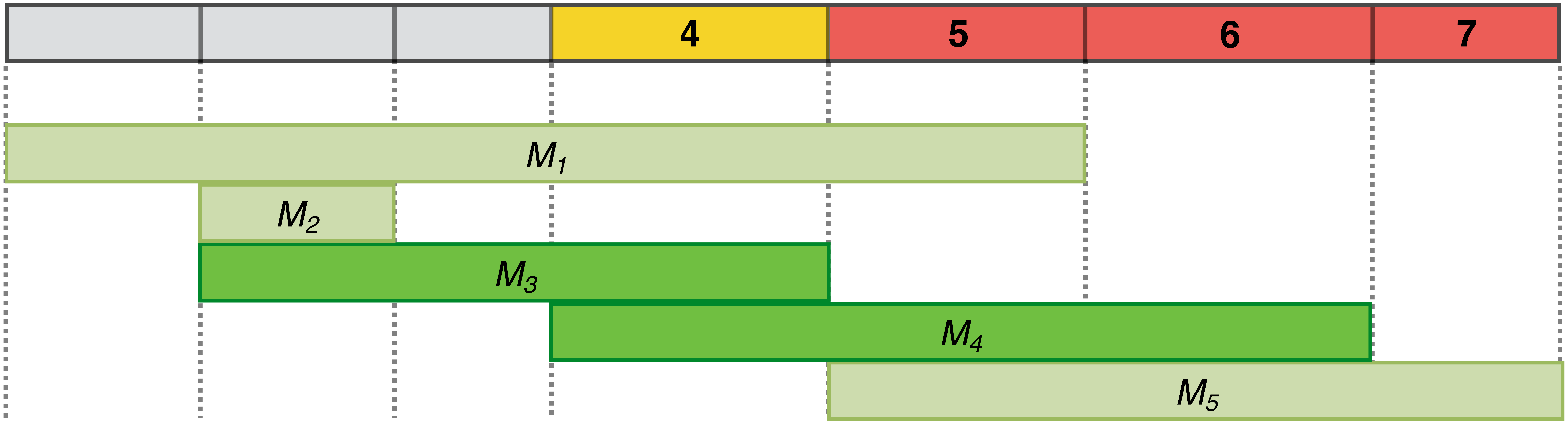
- Intersect
- Trim

Step 2b: Propagating



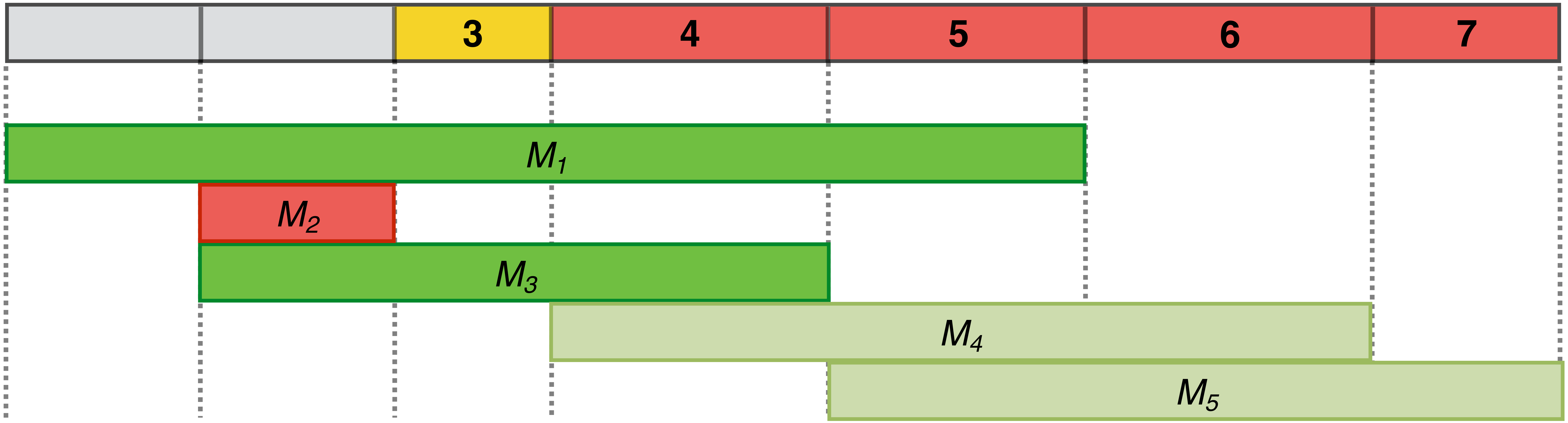
- Intersect
- Trim

Step 2b: Propagating



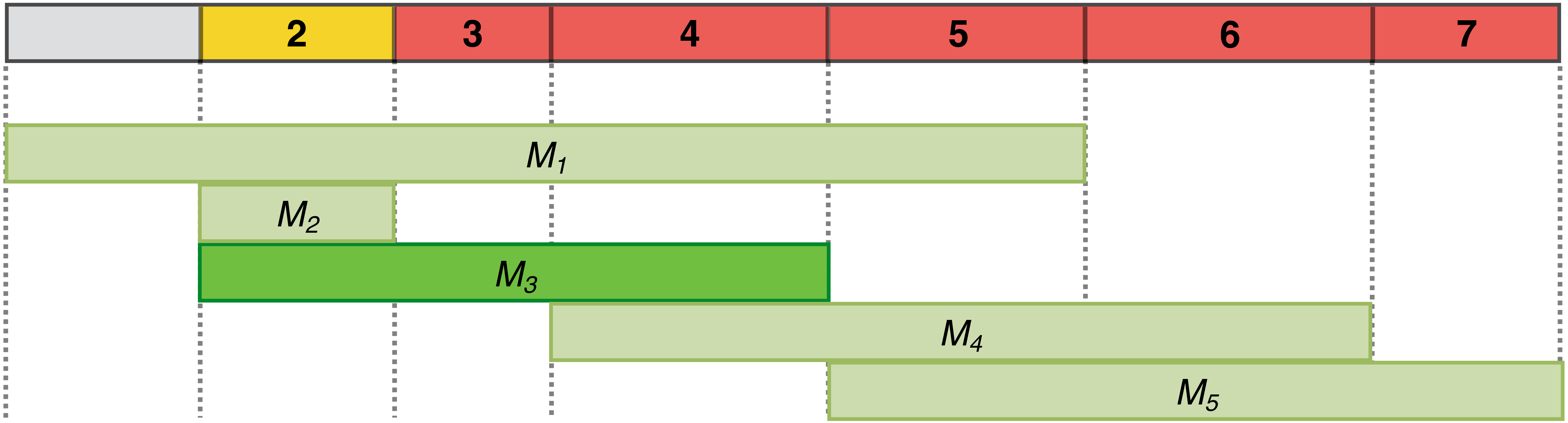
- Intersect
- Trim

Step 2b: Propagating



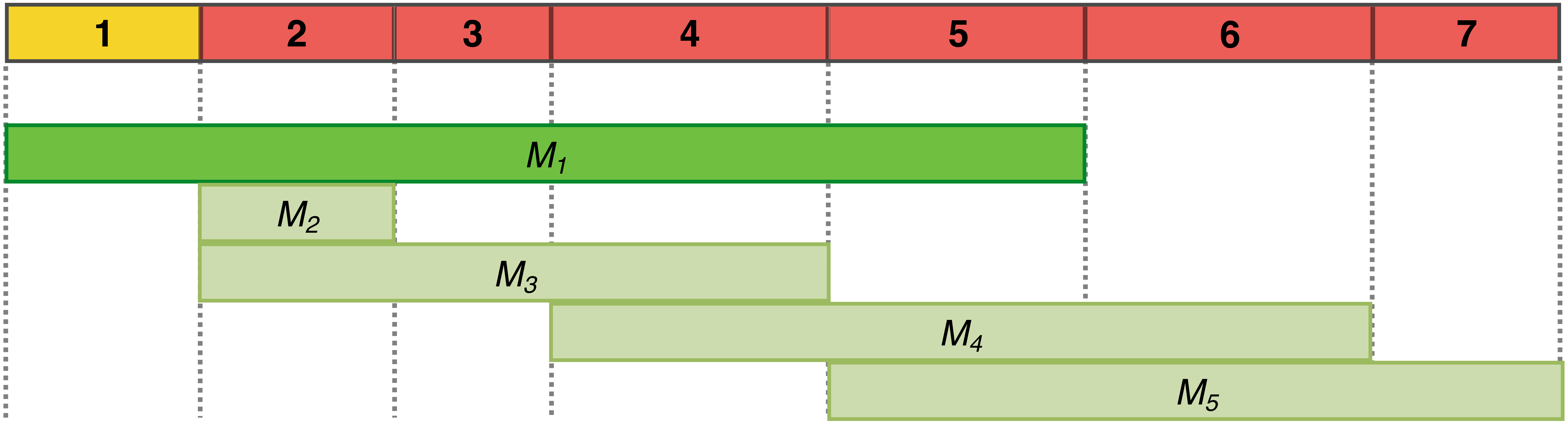
- Intersect
- Trim

Step 2b: Propagating



- Intersect
- Trim

Done!



- Intersect
- Trim

Full Reconstruction: Conclusion

- **Generic setting:** only access pattern leakage.
- **Partiotioning**, then **sorting** steps.

- Expectation of #queries **sufficient** for reconstruction:

$$N \cdot (3 + \log N) \qquad \text{for } N \geq 26$$

- Expectation of #queries **necessary** for reconstruction:

$$1/2 \cdot N \cdot \log N - O(N)$$

  for *any* algorithm.

- Our algorithm is data-optimal.

Reconstruction with Auxiliary Data + Rank Leakage

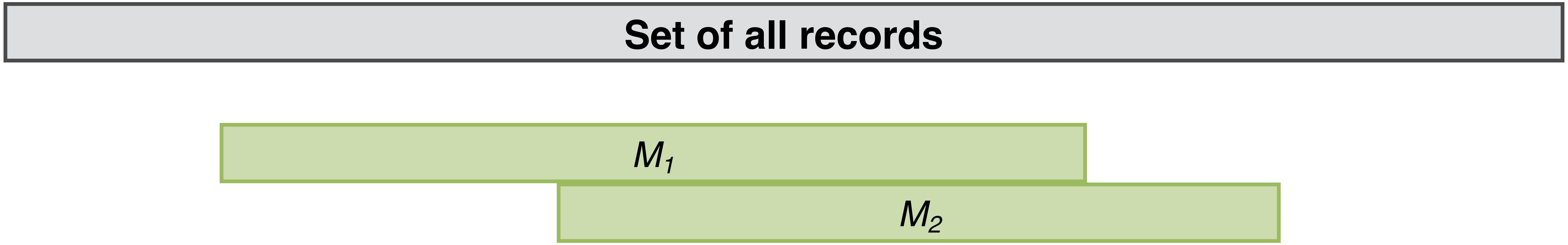
Auxiliary Data Attack with Rank Leakage

- Assume access pattern + rank leakage.
- Also assume an **approximation to the distribution on values** is known.

  "Auxiliary distribution".

  From aggregate data, or from another reference source.
- We show experimentally that, under these assumptions, **far fewer queries** are needed.

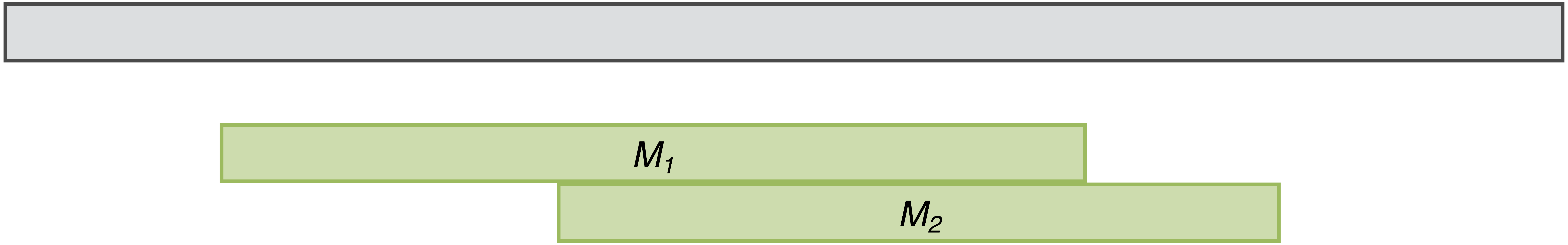
Auxiliary Data Attack Algorithm



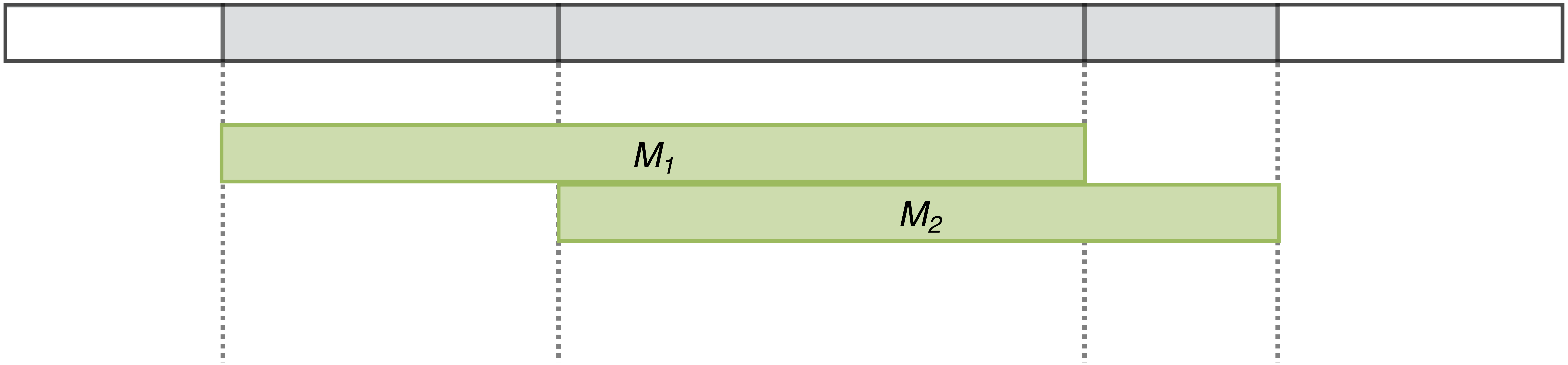
Assume $N = 125$ values, and 2 queries.

M_i = set of records matched by i-th query.

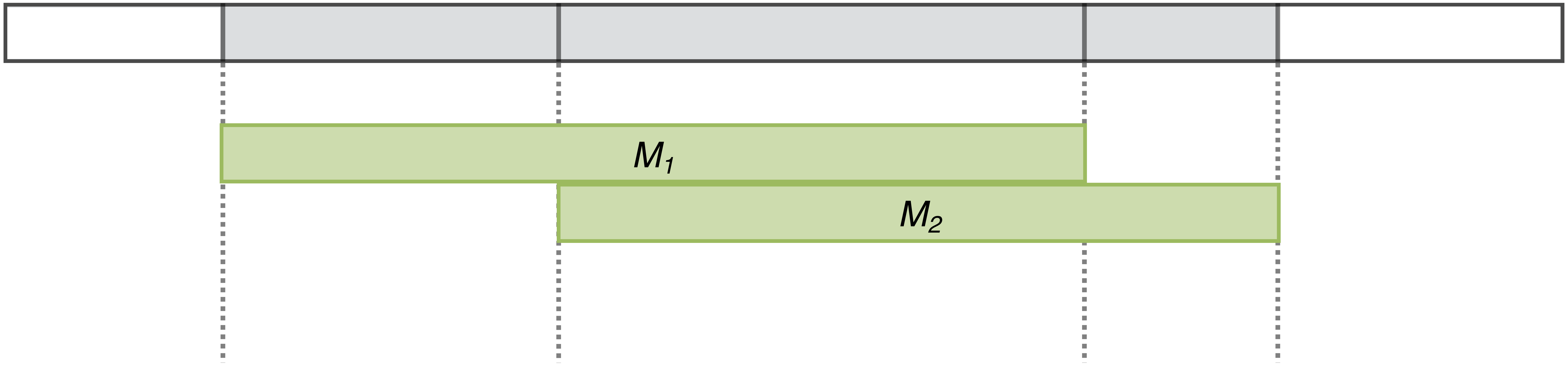
Partitioning and Matching



Partitioning and Matching



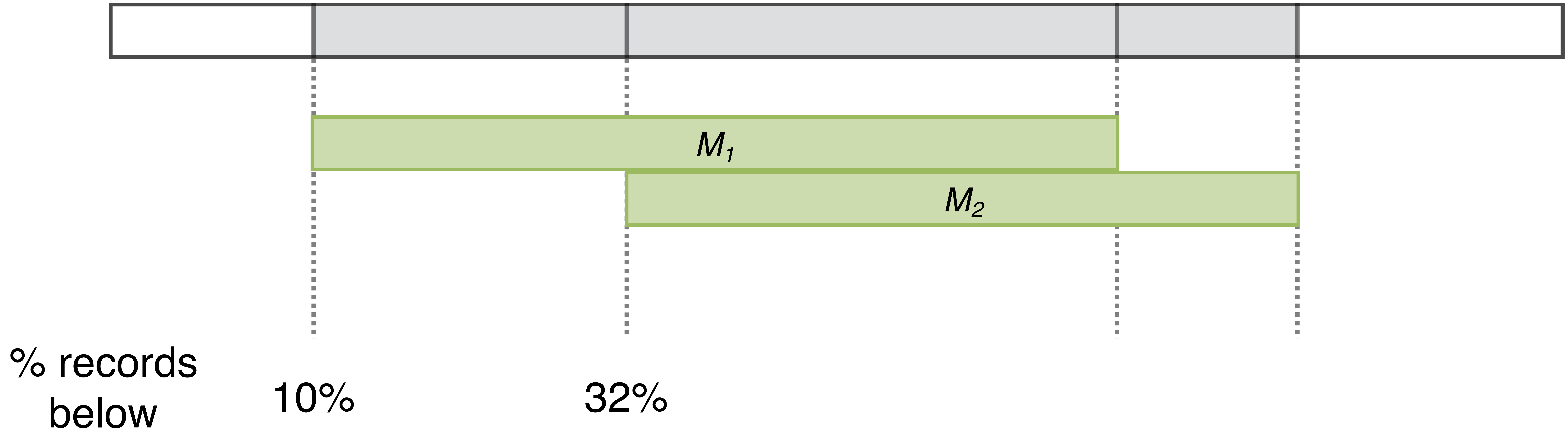
Partitioning and Matching



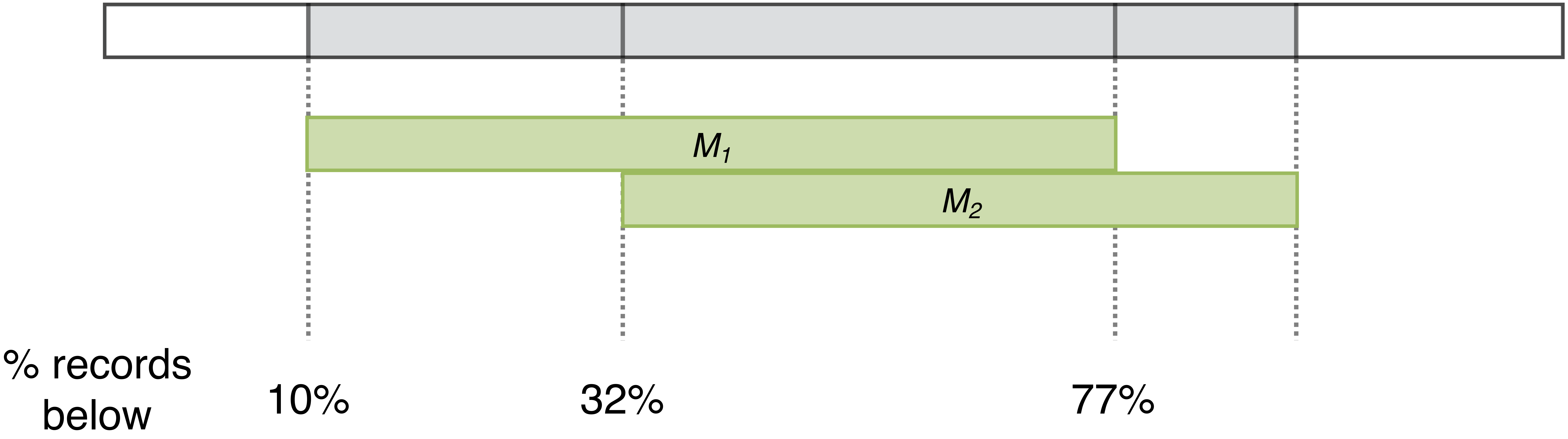
% records below

10%

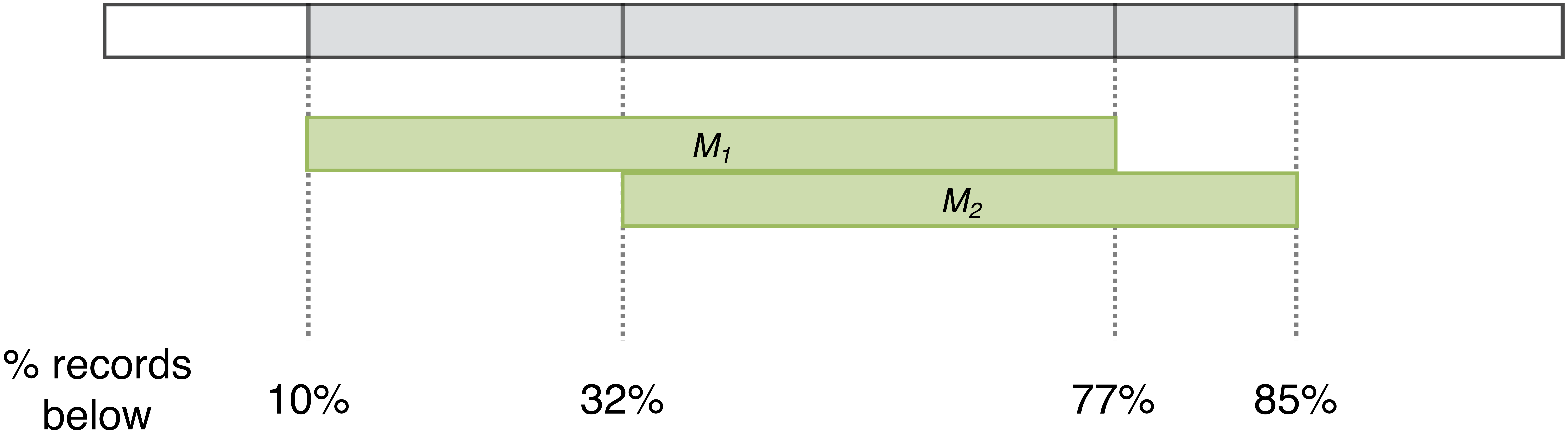
Partitioning and Matching



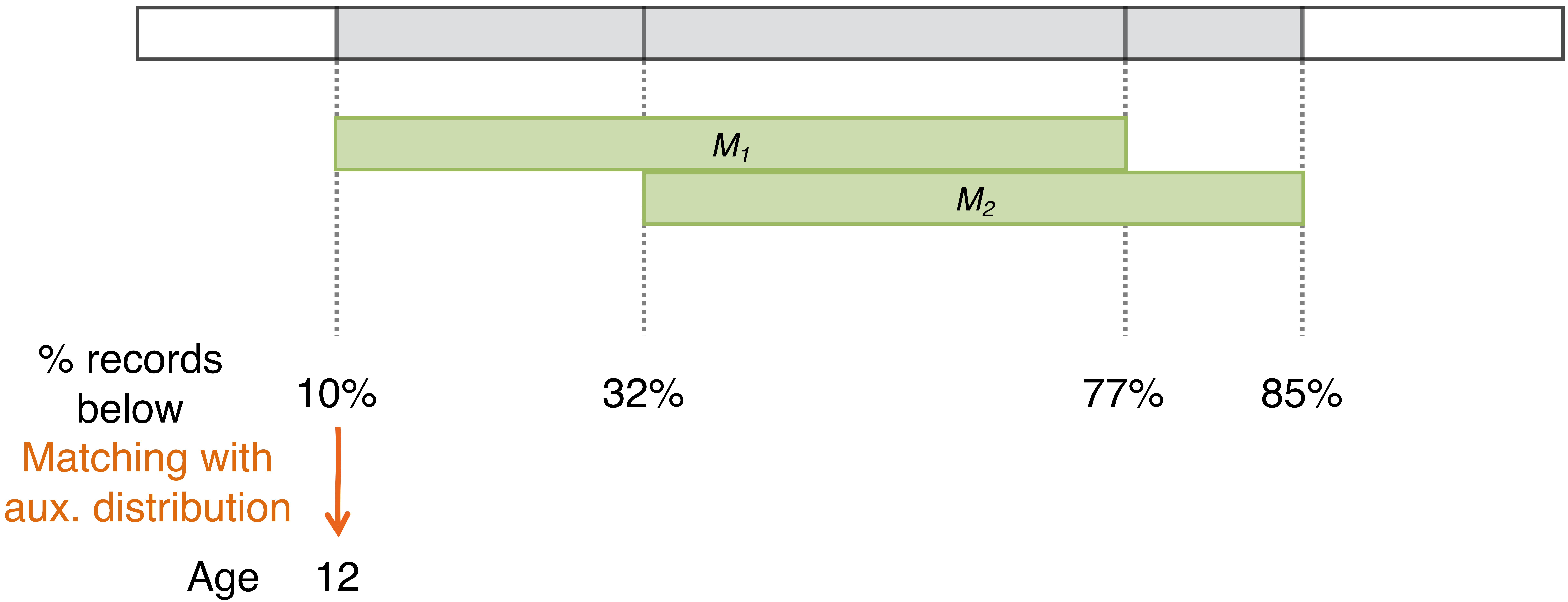
Partitioning and Matching



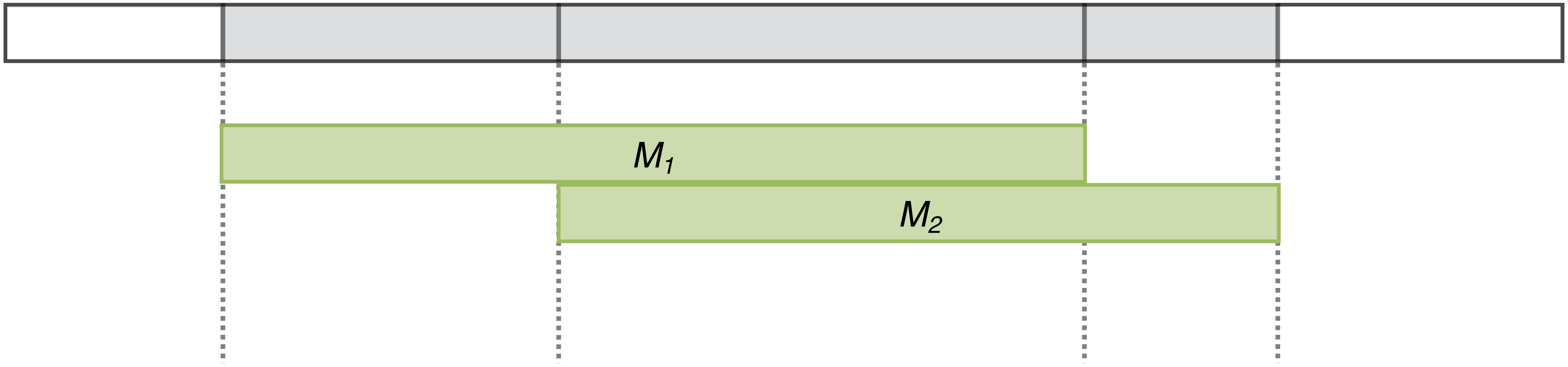
Partitioning and Matching



Partitioning and Matching



Partitioning and Matching

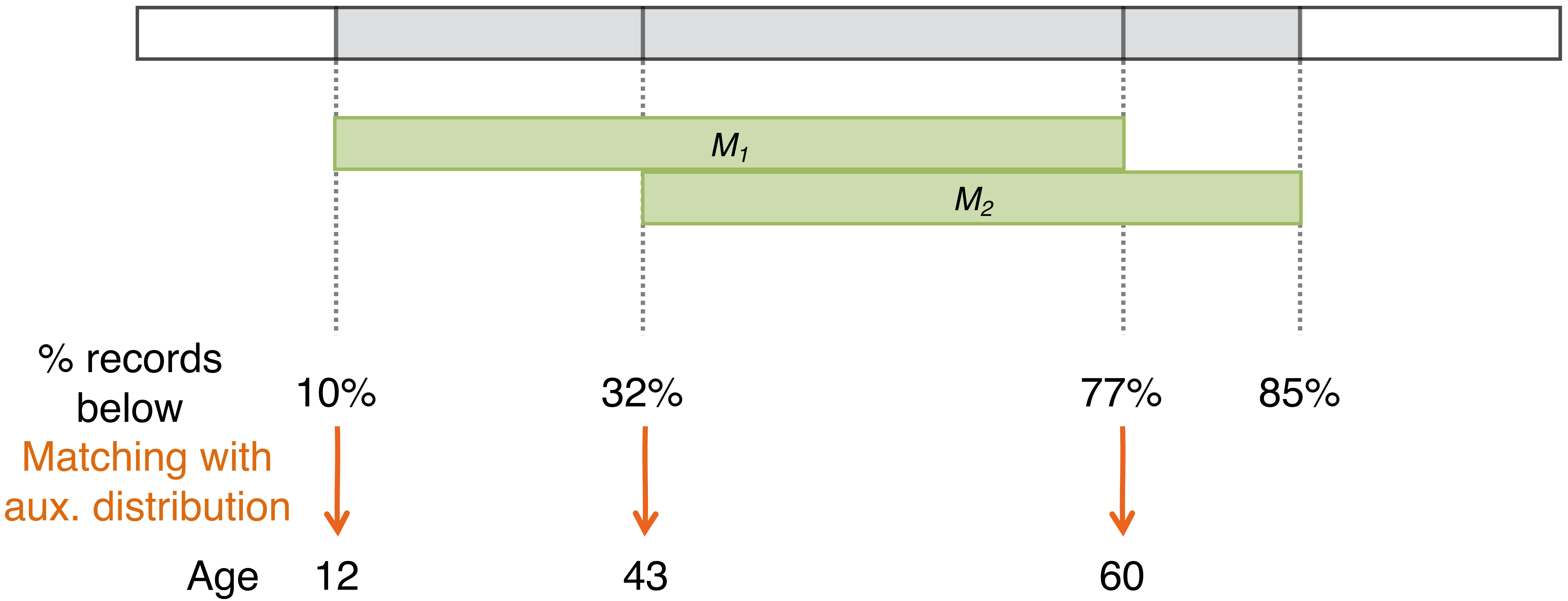


% records below 10% 32% 77% 85%

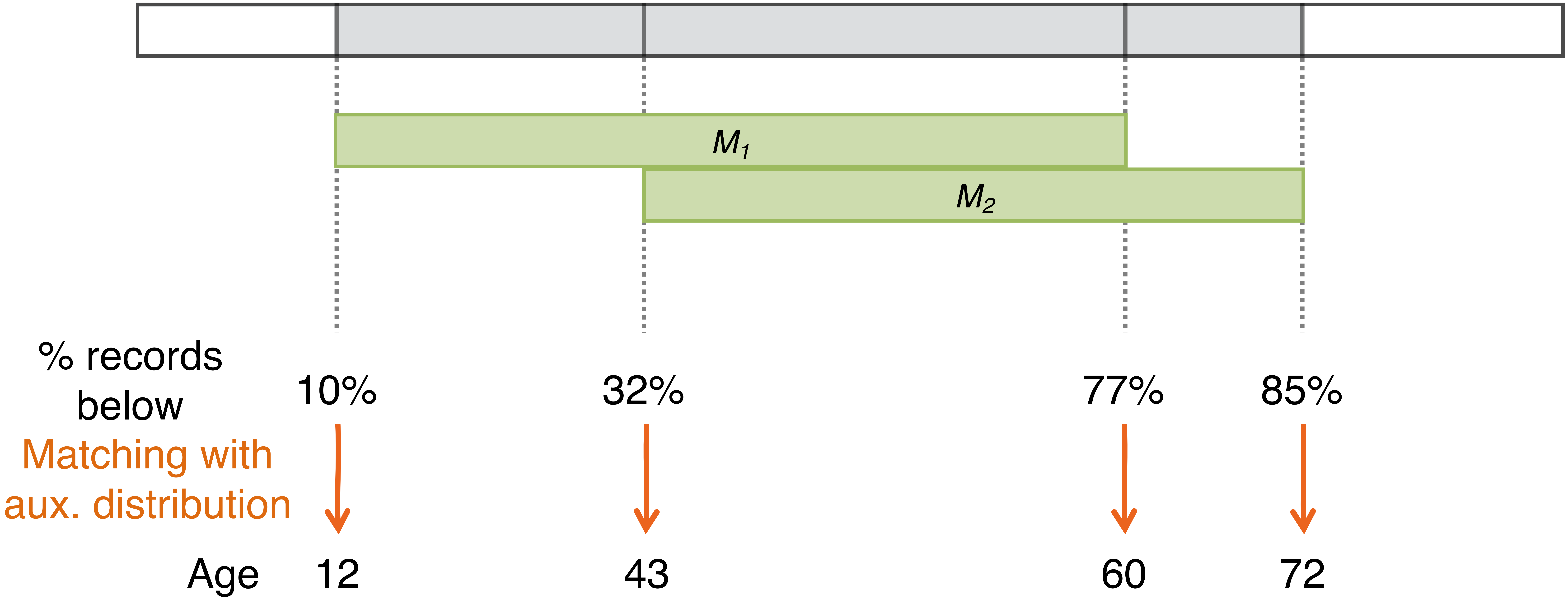
Matching with aux. distribution

Age 12 43

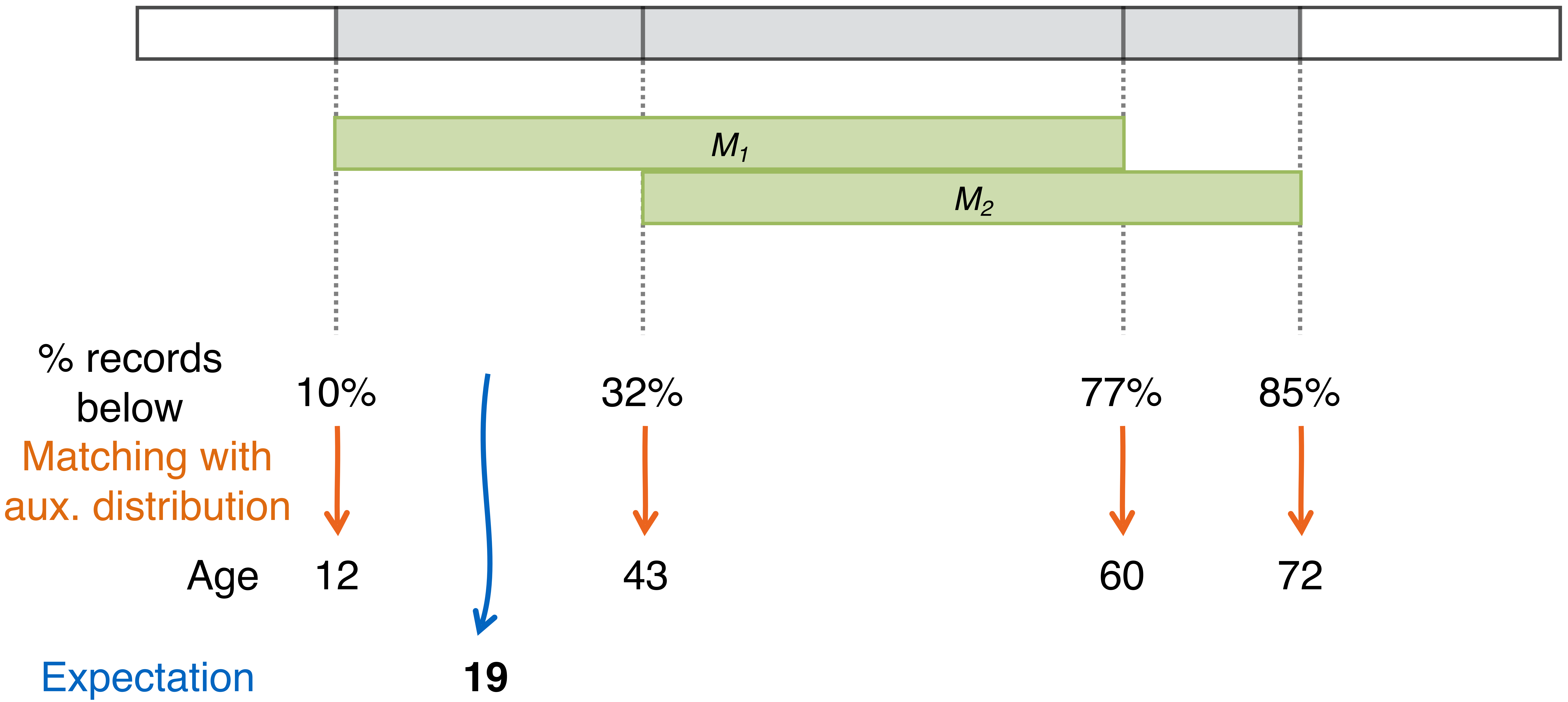
Partitioning and Matching



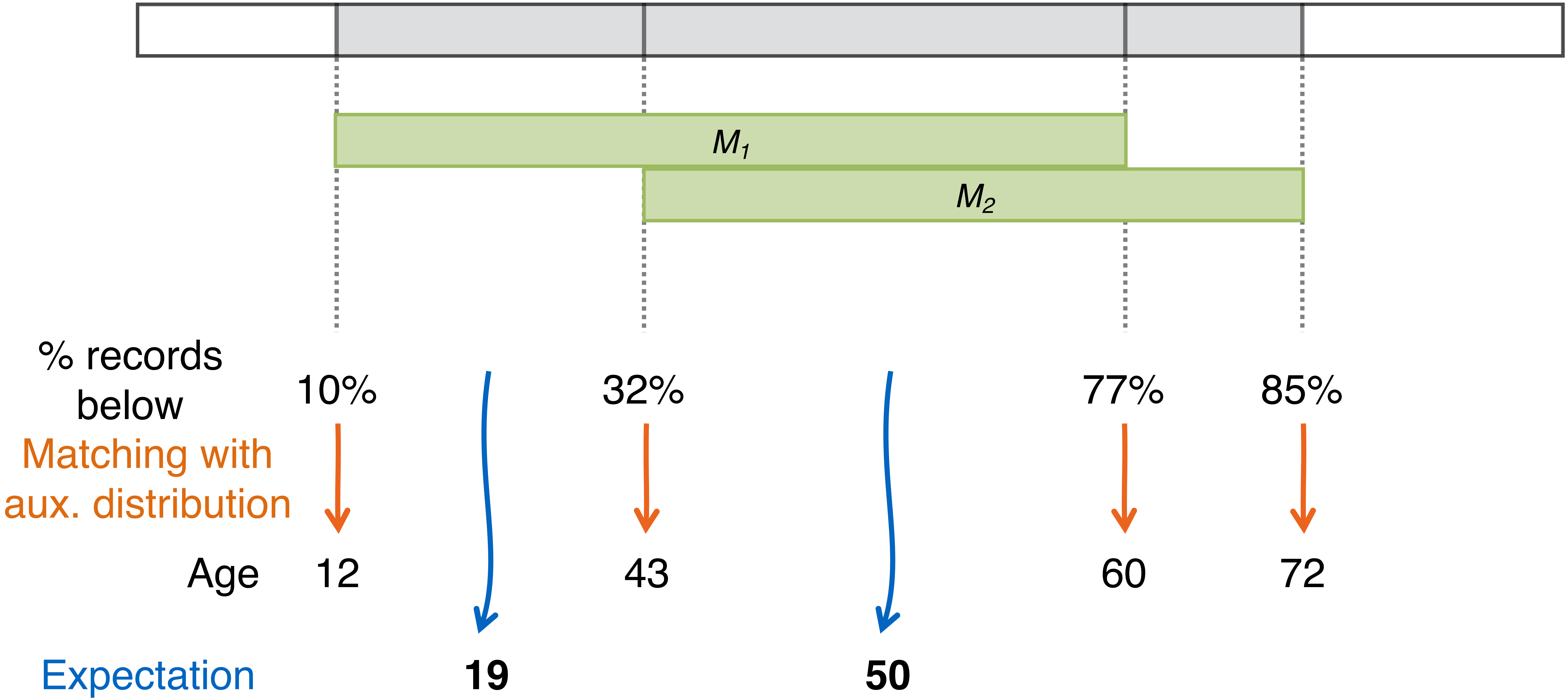
Partitioning and Matching



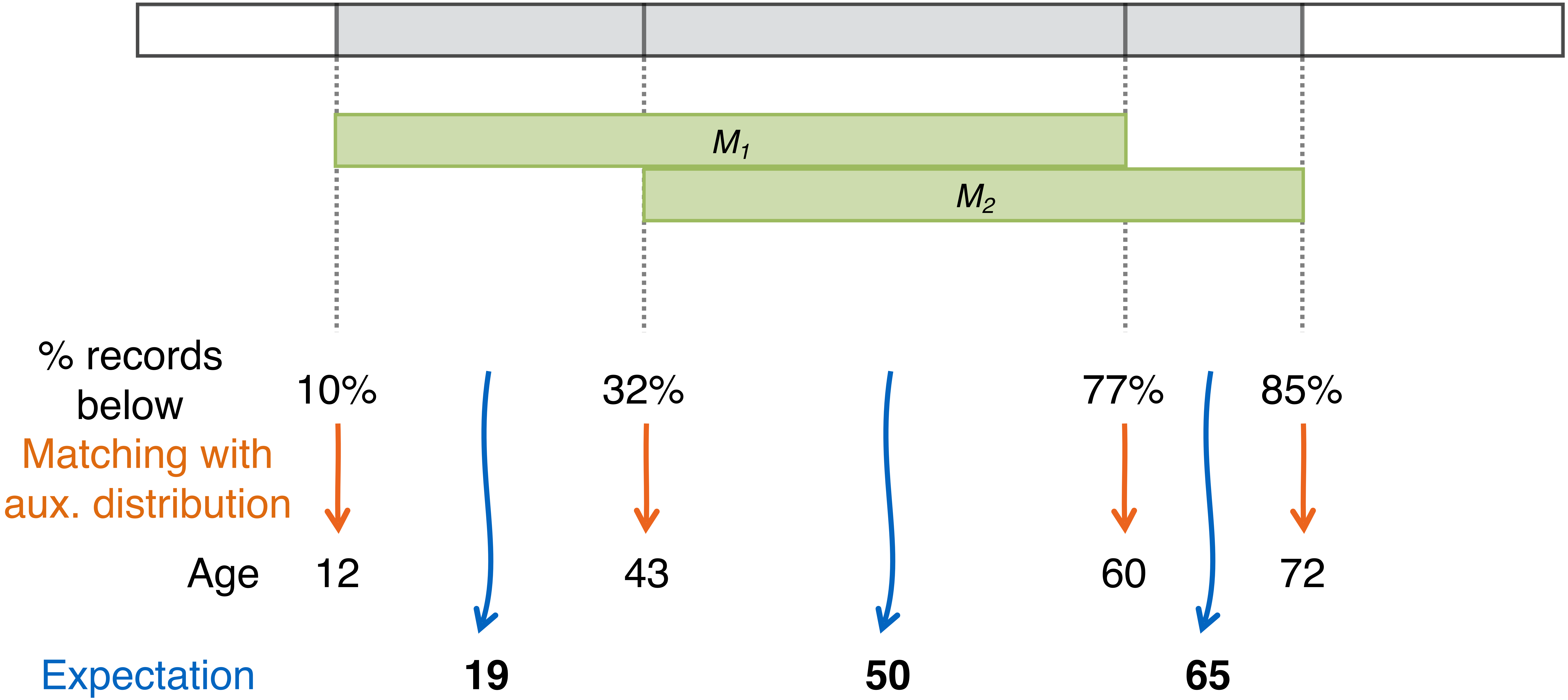
Partitioning and Matching



Partitioning and Matching



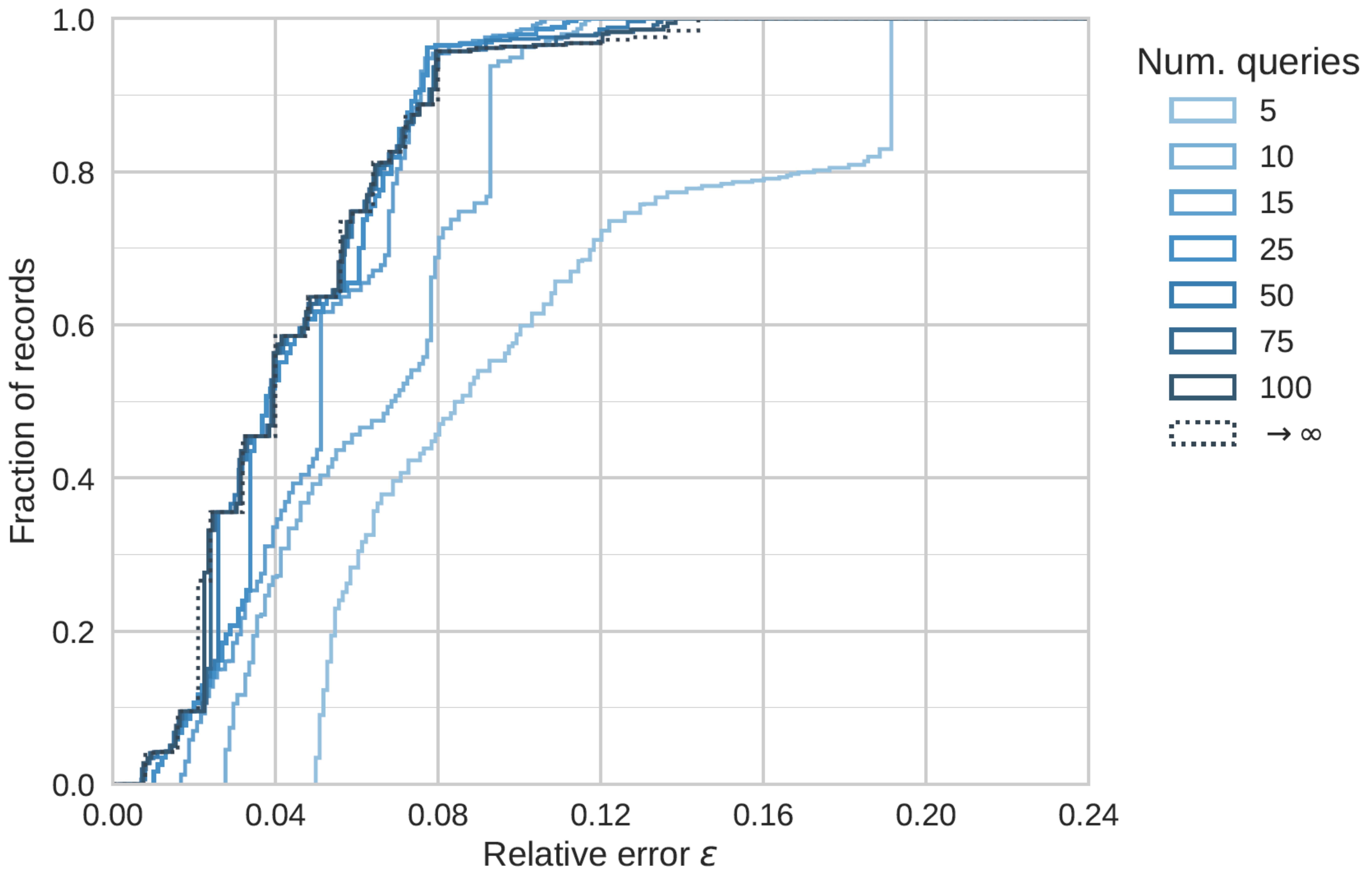
Partitioning and Matching



Auxiliary Data Attack: Experimental Evaluation

- Ages, $N = 125$.
- Health records from US hospitals (NIS HCUP 2009).
- **Target:** age of individual hospitals' records.
- **Auxiliary data:** aggregate of 200 hospitals' records.
- **Measure of success:** proportion of records with value guessed within ε.

Results with Imperfect Auxiliary Data



Conclusions

Reconstruction Attacks: Conclusions

Attack	Leakage	Other req'ts	Suff. # queries
KKNO16	AP	Density	$O(N^2 \log N)$
Full	AP + rank	Density	$N \cdot (\log N + 2)$
	AP	Density	$N \cdot (\log N + 3)$
ε-approx.	AP	Density	$5/4\ N \cdot (\log 1/\varepsilon) + O(N)$
Auxiliary	AP + rank	Auxiliary dist.	Experimental

- Full reconstruction ≈ $N \log N$ queries with only access pattern!

  Efficient, data-optimal algorithms + matching lower bound.

- For $N = 125$:

  800 queries → full reconstruction.

  25 queries → majority of records within 5%, using *auxiliary distribution* + rank.

Reconstruction Attacks: Conclusions

- Many clever schemes have been designed, enabling range queries on encrypted data.

OPE, ORE schemes, POPE, [HK16], BlindSeer, [Lu12], [FJKNRS15], FH-OPE, Lewi-Wu, Arx, Cipherbase, EncKV,…

- Second-generation schemes **defeat the snapshot adversary** (with caveats).
- But as our attacks show, **no known scheme offers meaningful privacy** vs. a **persistent adversary** (including server itself).
- More research needed!