Dynamic Anonymization for Marginal Publication

Xianmang He,Yanghua Xiao,
Wei Wang, etc
Fudan University

Privacy preserving data publishing

	Age	Zip	Sex	Disease
Bob	20	12k	M	bronchitis
Alex	19	20k	M	flu
Jane	20	13k	F	pneumonia
Lily	24	16k	F	gastritis
Jame	29	21k	F	flu
Linda	34	24k	F	gastritis
Sarah	39	19k	M	bronchitis
Mary	45	14k	M	flu
Andy	34	21k	F	pneumonia

Microdata *T*

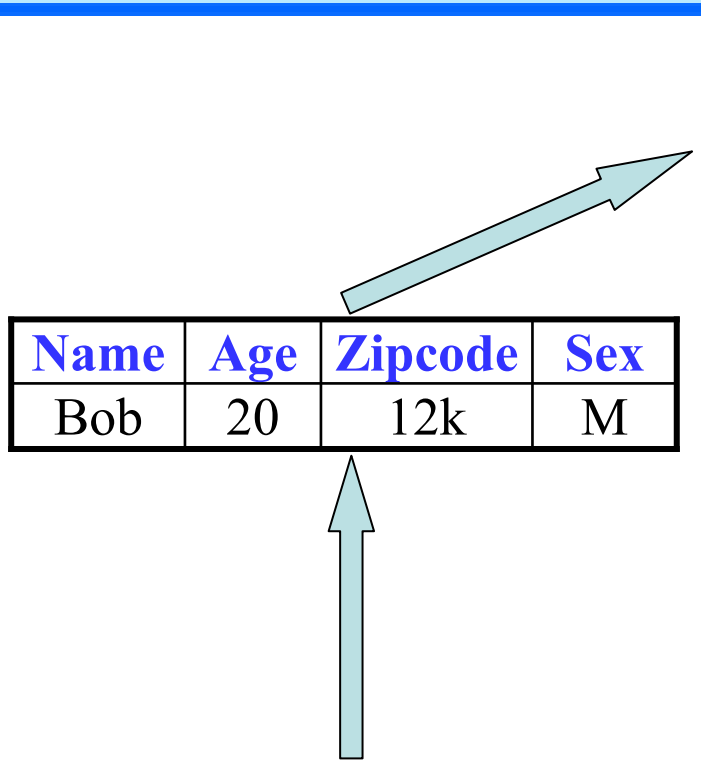
GID	Age	Zip	Sex	Disease
1	[19-20]	[12k-20k]	M	bronchitis
1	[19-20]	[12k-20k]	M	flu
2	[20-24]	[13k-16k]	F	pneumonia
2	[20-24]	[13k-16k]	F	gastritis
3	[29-34]	[21k-24k]	F	flu
3	[29-34]	[21k-24k]	F	gastritis
4	[34-45]	[14k-21k]	*	bronchitis
4	[34-45]	[14k-21k]	*	flu
4	[34-45]	[14k-21k]	*	pneumonia

Marginal <Age, Zip, Sex, Disease>

Name	Age	Zipcode	Sex
Bob	20	12k	M

An adversary

After several weeks,



Name	Age	Zipcode	Sex
Bob	20	12k	M

An adversary

Zip	Disease
[12k-13k]	bronchitis
[12k-13k]	pneumonia
[14k-16k]	gastritis
[14k-16k]	flu
[19k-20k]	flu
[19k-20k]	bronchitis
[21k-24k]	gastritis
[21k-24k]	flu
[21k-24k]	pneumonia

Marginal <Zip, Disease>

Motivating Example

- What the adversary learns from *Marginal<Age,Zip,Sex,Disease>*.

Name	Age	Zipcode	Sex
Bob	21	12k	M

G. ID	Age	Zipcode	Sex	Disease
1	[19, 20]	[12k, 20k]	M	bronchitis
1	[19, 20]	[12k, 20k]	M	flu
.....				

- What the adversary learns from *Marginal<Zip,Disease>*

Name	Age	Zipcode	Sex
Bob	21	12k	M

G. ID	Zipcode	Disease
1	[12k, 13k]	bronchitis
1	[12k, 13k]	penumonia

- So Bob must have contracted bronchitis!

ANGEL

GID	Zip	Batch-ID
1	[12k-13k]	1
1	[12k-13k]	2
2	[14k-16k]	4
2	[14k-16k]	2
3	[19k-20k]	1
3	[19k-20k]	4
4	[21k-24k]	3
4	[21k-24k]	3
4	[21k-24k]	4

BT

Batch-ID	Disease	Count
1	bronchitis	1
1	flu	1
2	pneumonia	1
2	gastritis	1
3	flu	1
3	gastritis	1
4	bronchitis	1
4	flu	1
4	pneumonia	1

GT

Select Count(*) From Table GT and BT Where Zip-Code □ [12k, 24k] And Disease='Pneumonia'.

The answer is 4, larger than the actual result 2

Contributions

- We propose a solution for marginal publication
- Our solution includes
  - a dynamic anonymization technique, whose effectiveness is verified by extensive experiments.;
  - We systematically explored the theoretic properties of marginal publication;

Related Work

- Y. Tao, H. Chen, etc Angel: Enhancing the utility of generalization for privacy preserving publication, *TKDE'09*
- D. Kifer and J. Gehrke, Injecting utility into anonymized datasets, *SIGMOD'06*
- C. Yao, X. S. Wang, and S. Jajodia, Checking for k-anonymity violation by views, *VLDB '05*
- K. Wang and B. C. M. Fung, Anonymizing sequential releases, in *KDD '06*

Outline

- The *m*-invariance Principle
- Problem Definition
- Existence of *m*-Invariant marginals
- Experimental Results
- Conclusion

The *m*-invariance principle

- *A set S of partitions is m-invariant if*

(1) Each partition in S is m-unique;

(2) For any partitions P_1, P_2 □ *S, and any tuple t* □ *T, t has the same signature in* P_1 *and* P_2

The *m*-invariance principle

- Lemma: if a sequence of generalized tables {*T**(1), …, *T**(*n*)} is *m*-invariant, then for any individual *o* involved in any of these tables, we have

$$risk(o) <= 1/m$$

Outline

- Counterfeited Generalization
- Problem Definition
- Evaluation of Disclosure Risk
- The *m*-invariance Principle
- Experimental Results
- Conclusion

Problem Definition

Given a table *T* and an integer *m*, we need to anonymize it to be a set of marginals $M_j (1 \leq j \leq r)$ such that :

- Existence: these marginals are m-invariant
- Optimality: the information loss measured by NCP is minimized

Outline

- The *m*-invariance Principle
- Problem Definition
- Existence of m-Invariant marginals
- Experimental Results
- Conclusion

Existence of m-Invariant marginals

- Theorem : *If a table T is m-eligible, then there exists a set of marginals* $\{M_1, M_2, \ldots, M_r\}$ *that is m-invariant.*
- Theorem: *A table T is m-eligible, if and only if the number of tuples that have the same sensitive attribute values is at most* $|T|/m$ *, where* $|T|$ *is the number of tuples in table T.*

Outline

- Counterfeited Generalization
- Problem Definition
- Evaluation of Disclosure Risk
- Algorithm
  - The partition step
  - The Assign Step
  - Dynamic Anonymization Technique
- Experimental Results
- Conclusion

The *partition* step

- Input: A microdata *T*, integers *k* and *m*
- Output: A set *S* consisting of sub-tables of *T*;
- /* the parameter *k* is number of rounds to partition *G**/
- 1. *S* = {*T* };
- 2. While(exist *G* □ *S* that has not been partitioned){
- 3. For *i* = 1 to *k*
- 4. Randomly shuffle the tuples of *G*;
- 5. Set *G*1 = *G*2 =Φ;
- 6. Add tuple *t*1 (*t*2) of extremely maximal (minimal) value to *G*1(*G*2);
- 7. For any *w,*compute □1=NCP(G1□□w) and □ 2=NCP(G2□□w) .
- If(□ 1 < □ 2) then Add *w* to *G*1, else add *w* to *G*2;
- 8. If both *G*1 and *G*2 are *m*-eligible
- remove *G* from *S*, and add *G*1-{t1},*G*2-{t2} to *S*, break;
- 9.Return *S*;

The *Assign* step

Given a set of sub-tables *Ti* passed from the previous phase, the assigning step is to divide each *Ti* into buckets such that each bucket constitutes a bucketization.

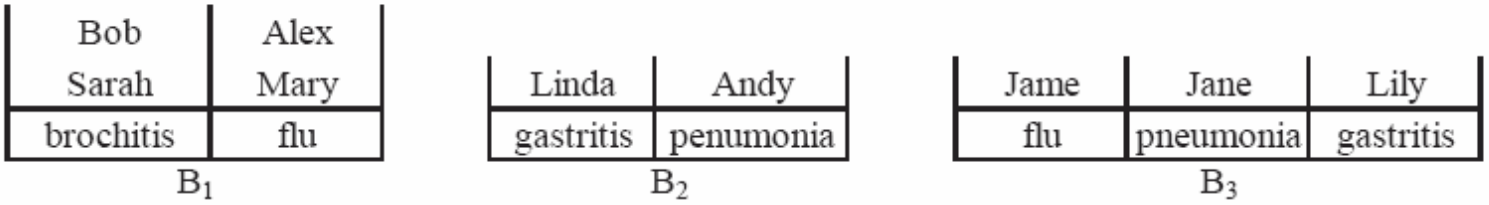


Fig. 3. Illustration of bucketization

Dynamic Anonymization Technique

- In real applications, a publisher may want to release a set of marginals {M1,M2,…, Mr} that overlap with each other in an arbitrarily manner. To help publishers accomplish this, we use the third step: decomposing step to produce a set of marginals {M1, M2, …, Mr} that are m-invariant.

- A bucketization U obtained from the previous two steps and a required schema of marginal Mj specified by publishers are given as the input of this step. Depending on marginals of different attribute sets, the bucketization U is decomposed differently. Each decomposition of a bucketization U is essentially a partition of the microdata T. All of the partitions constitute an m-invariant set while offering strong privacy guarantees.

Example

Bob	Alex
brochitis	flu

Sarah	Mary
brochitis	flu

B_1

Linda	Andy
gastritis	penumonia

B_2

Jame	Jane	Lily
flu	pneumonia	gastritis

B3

Age	Sex	Disease
[19-20]	M	bronchitis
[19-20]	M	flu
[20-29]	F	pneumonia
[20-29]	F	gastritis
[20-39]	F	flu
[34-34]	F	gastritis
[34-34]	F	pneumonia
[39-45]	M	flu
[39-45]	M	bronchitis

Marginal{Age,Sex,Diesease}

Sarah	Alex
brochitis	flu

Bob	Mary
brochitis	flu

B_1

Linda	Andy
gastritis	penumonia

B_2

Jame	Jane	Lily
flu	pneumonia	gastritis

B3

Zip	Disease
[12k-14k]	bronchitis
[12k-14k]	flu
[19k-20k]	bronchitis
[19k-20k]	flu
[13k-21k]	pneumonia
[13k-21k]	gastritis
[13k-21k]	flu
[21k-24k]	gastritis
[21k-24k]	pneumonia

Marginal{Zip,Disease}

Outline

- The *m*-invariance Principle
- Problem Definition
- Existence of m-Invariant marginals
- Experimental Results
- Conclusion

Experiment Settings

- Real dataset with 600k tuples and 6 attributes:
  - Age, Gender, Education, Martial,Race,
  - Sensitive Attribute(SA):Salary-Class
- Marginals: M1<Age, SA>, M2<Age,Gender,SA>, M3<Age,Gender,Education,SA>, M4<Age,Gender,Education,Martial,SA> M5<Age,Gender,Education,Martial,Race,SA>

Query Accuracy

SELECT COUNT(*) FROM $T_x(j)$
WHERE $pred(A_1^{qi})$ AND ... AND $pred(A_4^{qi})$ AND $pred(A^s)$

- SELECT COUNT(*) FROM *SAL*
  WHERE *Age* < 30 AND *SA* = *Manager*

- Query error = |*act* – *est*| / *act*
- Each workload contains 1000 queries.
- We measure the median error of each workload.

Utility

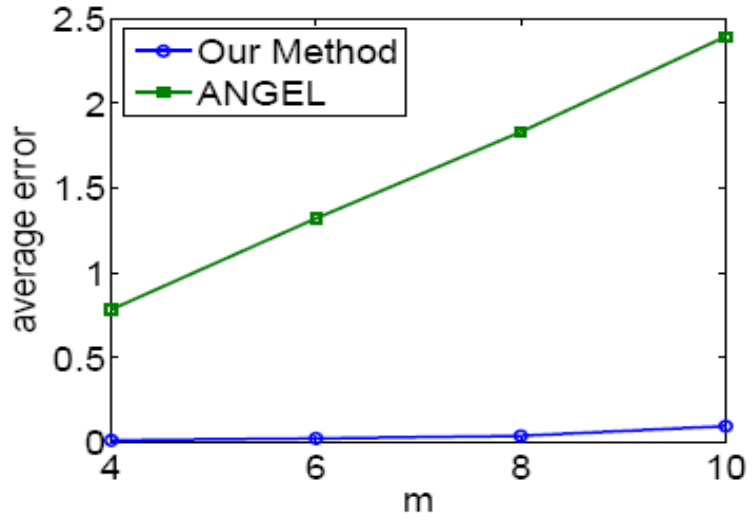


Fig. 5. Query accuracy vs. m ($s = 0.1$, M_5)

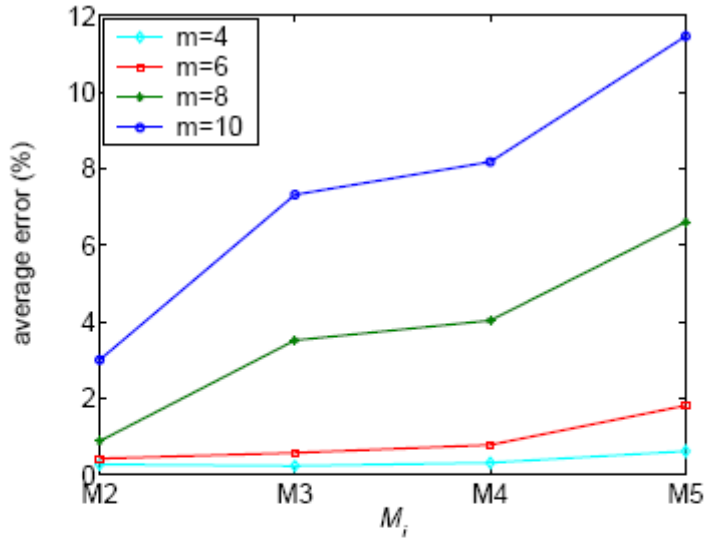
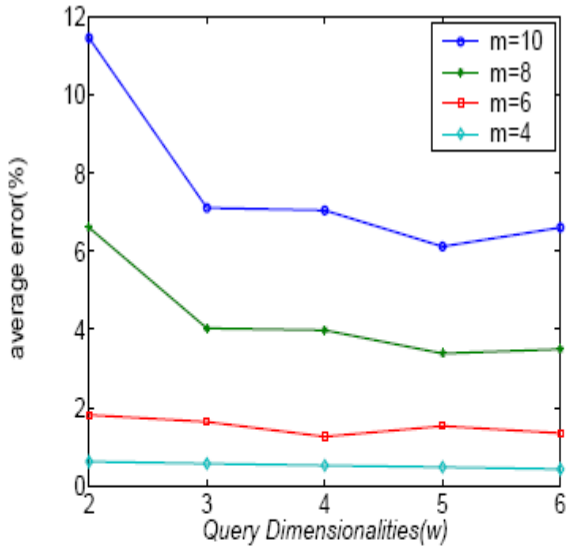


Fig. 6. Query accuracy vs. M_i($s = 0.1$)

Query Accuracy: w , s, n



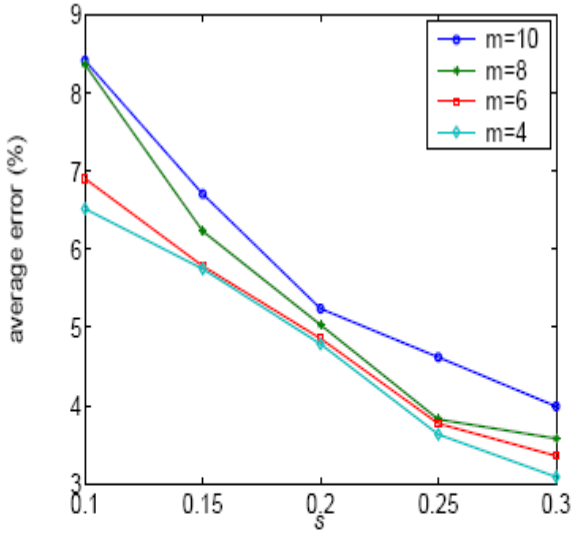


Fig. 7. Query accuracy vs. Query Dimensionalities w $(s = 0.1)$

Fig. 8. Query accuracy vs. s (M_5)

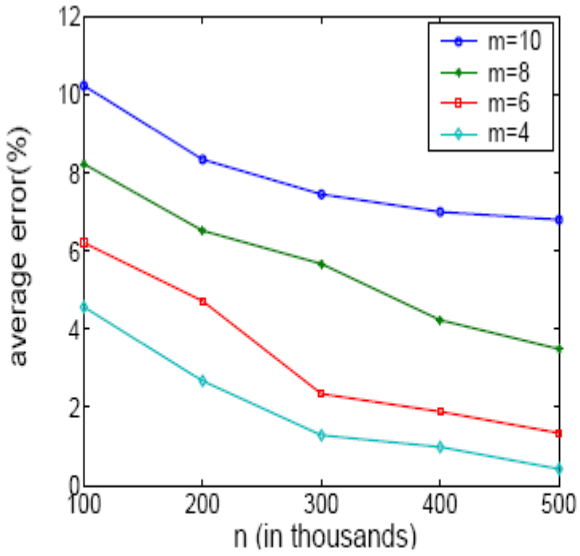


Fig. 9. Query accuracy vs. Cardinality n (M_5)

Efficiency of Our Algorithm

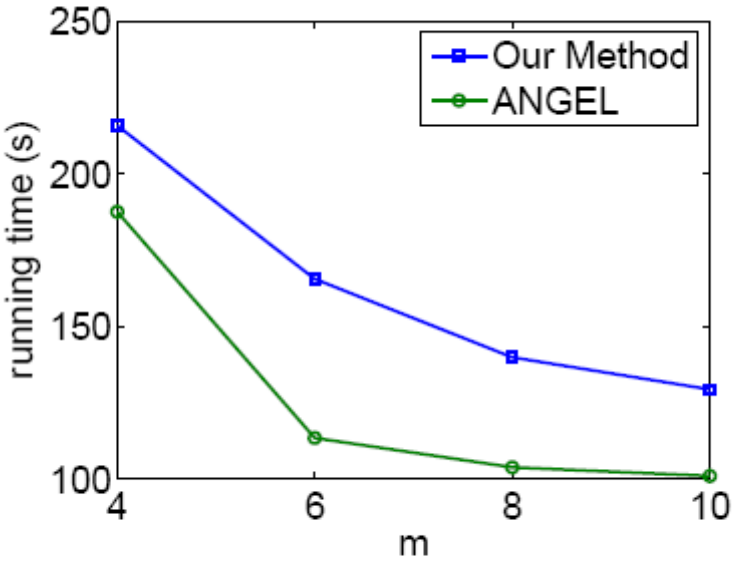


Fig. 10. Running time vs. m

Conclusions

- Existing solutions do not support marginal publication well!
- We devise a framework for analyzing marginal publication.
- We develop an efficient algorithm for computing marginals that are *m*-invariant.

Thank you for your attention!