Privacy-Preserving Data Publishing

Masooma Iftikhar
masooma.iftikhar@anu.edu.au

School of Computing
College of Engineering and Computer Science
The Australian National University, Canberra, Australia

January 27, 2022

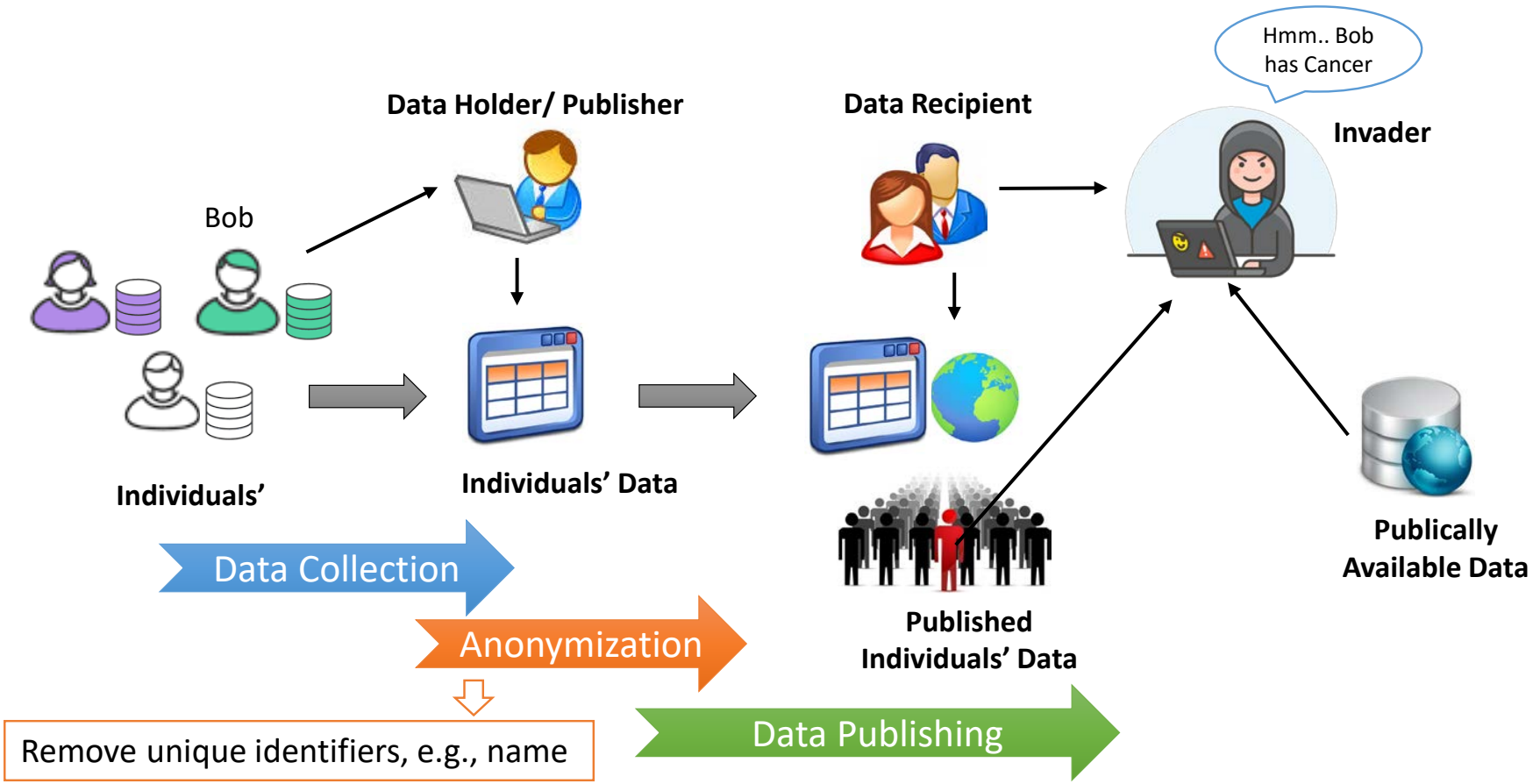
Outline

- Introduction
- Related Work
- Research Goals and Challenges
- Contributions and Publications
- Privacy-Preserving Data Publishing (PPDP) Mechanisms
  - Relational Data Publishing
  - Graph Data Publishing

Introduction

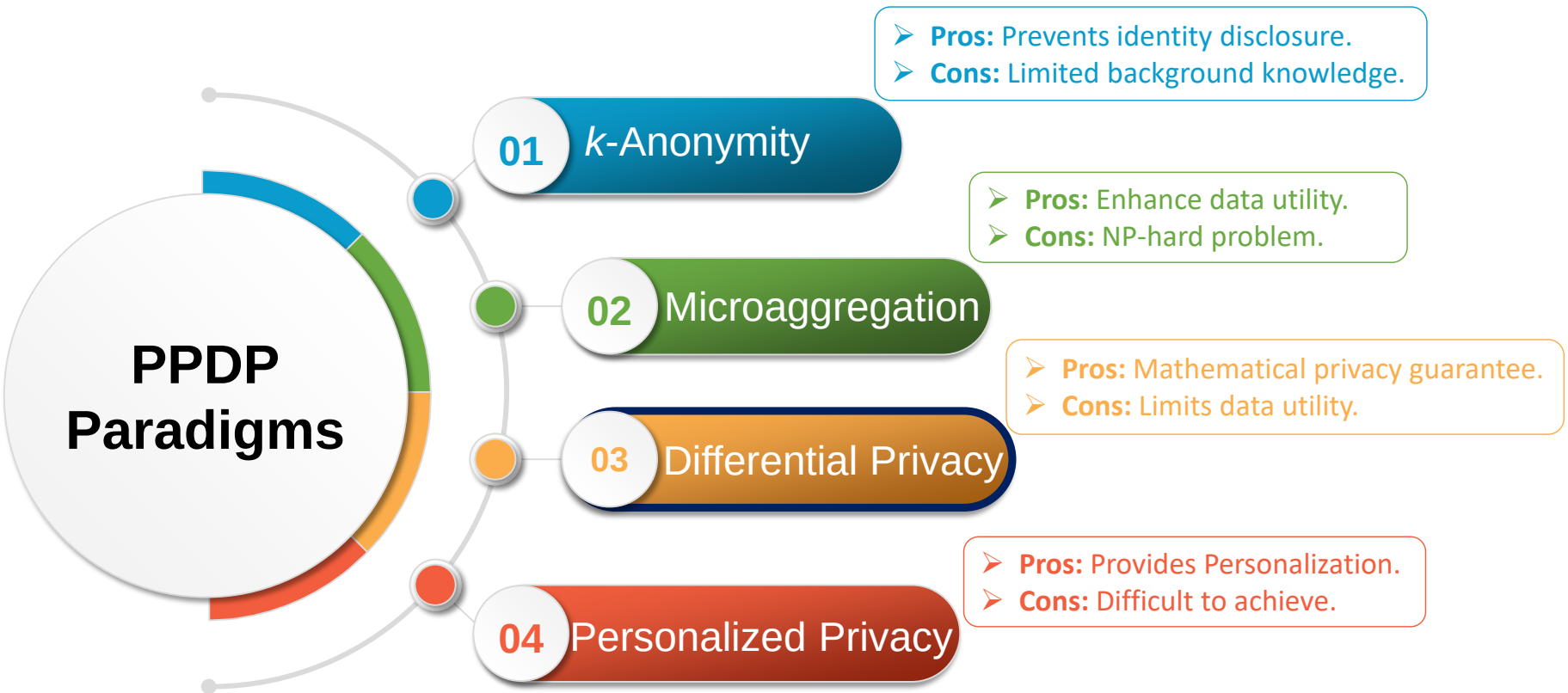
Data Publishing Process

- Publishing data about individuals poses a privacy threat.



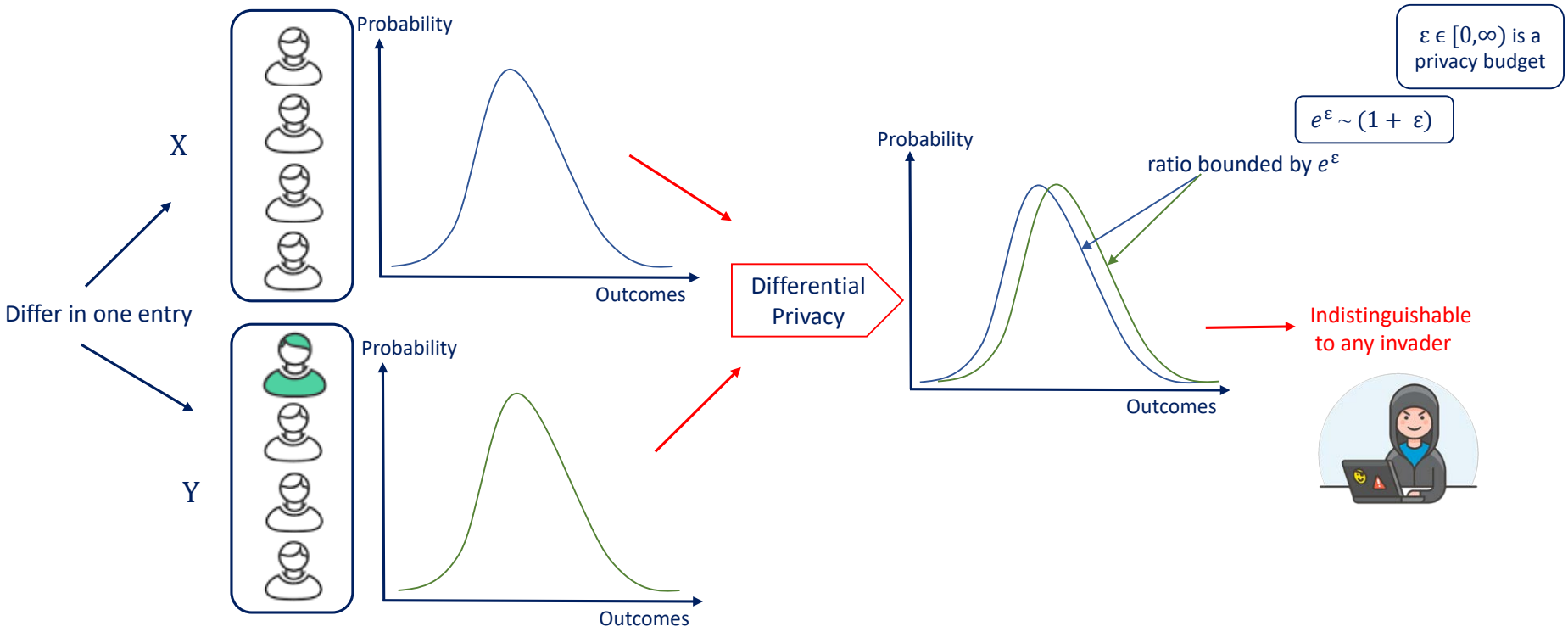
Related Work

PPDP Paradigms



Differential Privacy (DP)

DP bounds a shift in the output distribution of a randomized mechanism $\mathcal{K}$ that can be caused by a small change in its input.

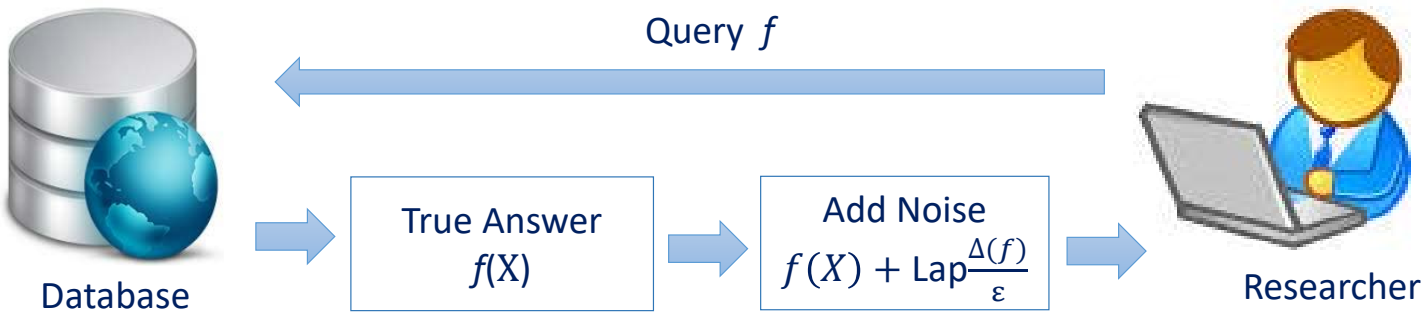


If mechanism $\mathcal{K}$ satisfies ε-DP then

$$Pr[\mathcal{K}(X) \in \mathcal{D}] \leq e^{\varepsilon} Pr[\mathcal{K}(Y) \in \mathcal{D}]$$

Standard way of Achieving DP

One way to satisfy DP is to add controlled noise to the output of a query f using a random distribution (e.g., Laplace).



Noise is calibrated according to sensitivity (Δ) of f and ε.

- Δ is maximum variation in f between X and Y.
- $\varepsilon > 0$ is the privacy parameter.

Research Goals and Challenges

Goals

To develop privacy-preserving data publishing (PPDP) mechanisms.

- **Privacy:** Provide privacy guarantee of differential privacy.
- **Utility:** Enhance the accuracy of published data while providing differential privacy.

Key Challenges

- How to determine the right amount of noise that guarantees both privacy and accuracy under DP?
- How to reduce the amount of noise needed to achieve DP by controlling sensitivity?
- How to enhance data utility under different data structures while providing DP guarantee?

Contributions and Publications

Contributions

Australian National University

- **Part I:** Relational Data Publishing
  - Reduce sensitivity by incorporating microaggregation into DP.
  - Enhance overall utility by reducing both error produced during microaggregation and DP.
- **Part II:** Graph Data Publishing
  - Preserve topological structure of a graph through adding controlled perturbation to its edges.
  - Publish higher-order network statistics while providing DP guarantee to nodes in a network.
  - Consider personalization to achieve personal data protection under DP while enhancing utility.

Publications

- **Part I:** Relational Data Publishing
  - M.Iftikhar, Q.Wang, Y.Lin. **Publishing Differentially Private Datasets via Stable Microaggregation**. EDBT 2019.
  - M.Iftikhar, Q.Wang, Y.Li, Y.Lin, J.Shao. **Differentially Private Data Release via α-Stable Microaggregation**. Under journal preparation.
- **Part II:** Graph Data Publishing
  - M.Iftikhar, Q.Wang, Y.Lin. **dK-Microaggregation: Anonymizing Graphs with Differential Privacy Guarantees**. PAKDD 2020.
  - M.Iftikhar, Q.Wang. **dK-Projection: Publishing Graph Joint Degree Distribution with Node Differential Privacy**. PAKDD 2021.
  - M.Iftikhar, Q.Wang, Y.Li. **dK-Personalization: Publishing Network Statistics with Personalized Differential Privacy**. Accepted by PAKDD 2022.

Relational Data Publishing

Neighboring Datasets

Name	Zip	Age	Nationality	Disease
Eve	13053	28	Russian	Heart
Alice	13068	29	American	Heart
Bob	13068	21	Japanese	Flu
Amy	14853	50	Indian	Cancer

X

Name	Zip	Age	Nationality	Disease
Eve	13053	28	Russian	Heart
Alice	13068	29	American	Heart
Bob	13068	21	Japanese	Flu
Farhan	26001	29	Pakistan	Covid-19

Y

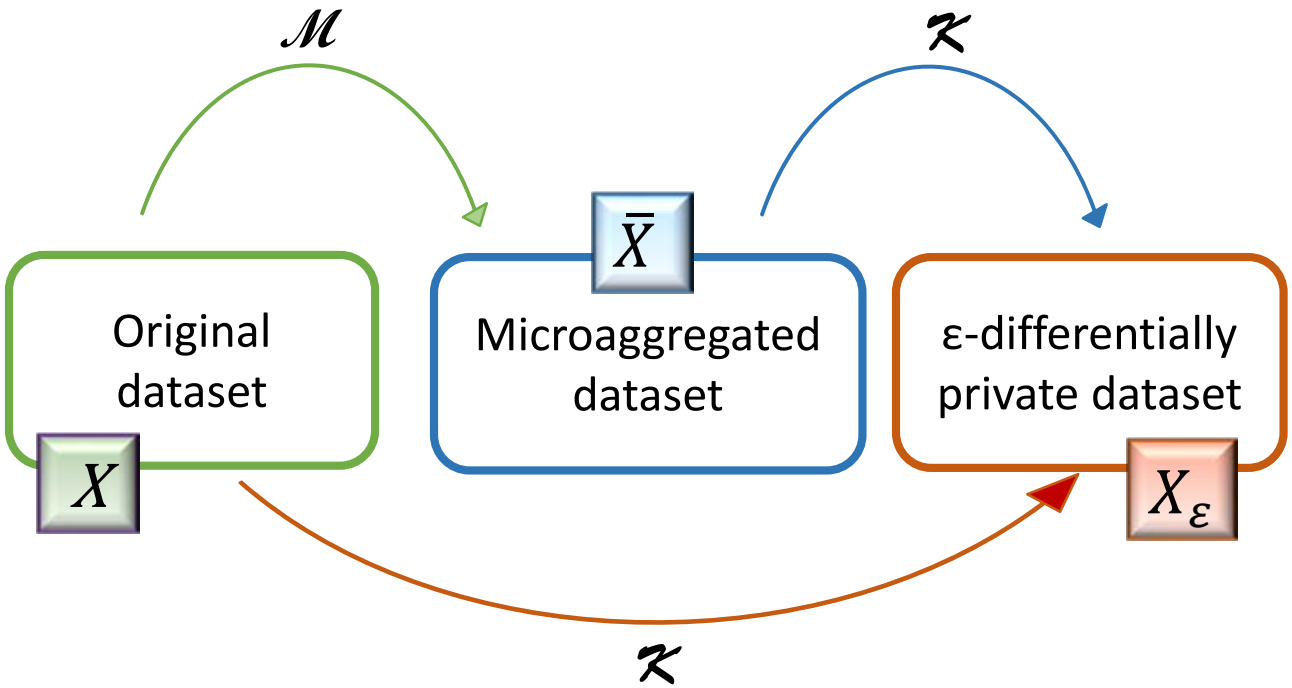
X and *Y* differ in one record

Problem Statement

- **Problem:** Given a dataset X, we want to generate X_ϵ that can provide DP guarantee.
- **Key Observation:** Microaggregation can help to reduce sensitivity for improving data utility.

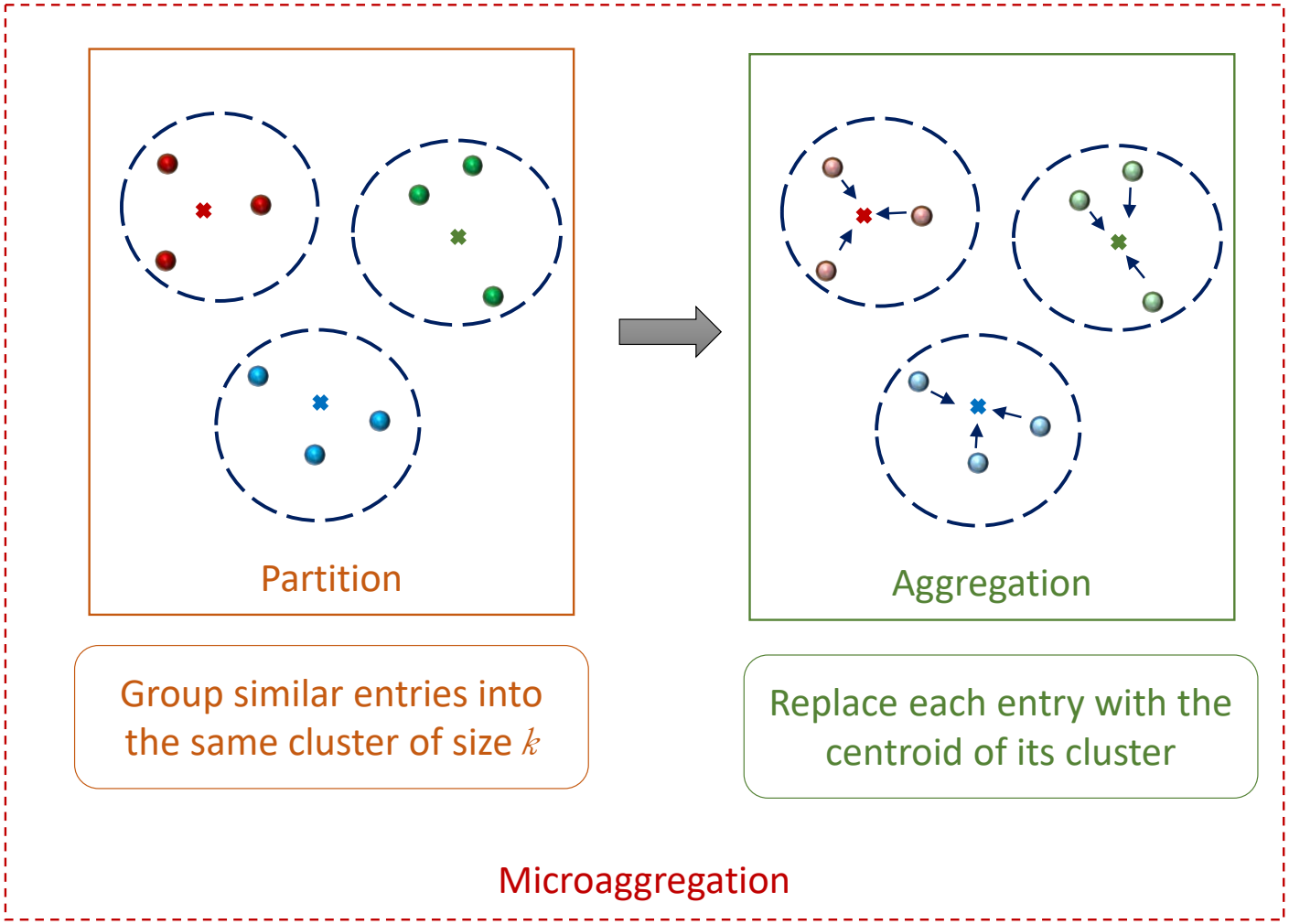
Proposed Solution

- A microaggregated dataset $\overline{X}$ is added between X and X_ε.

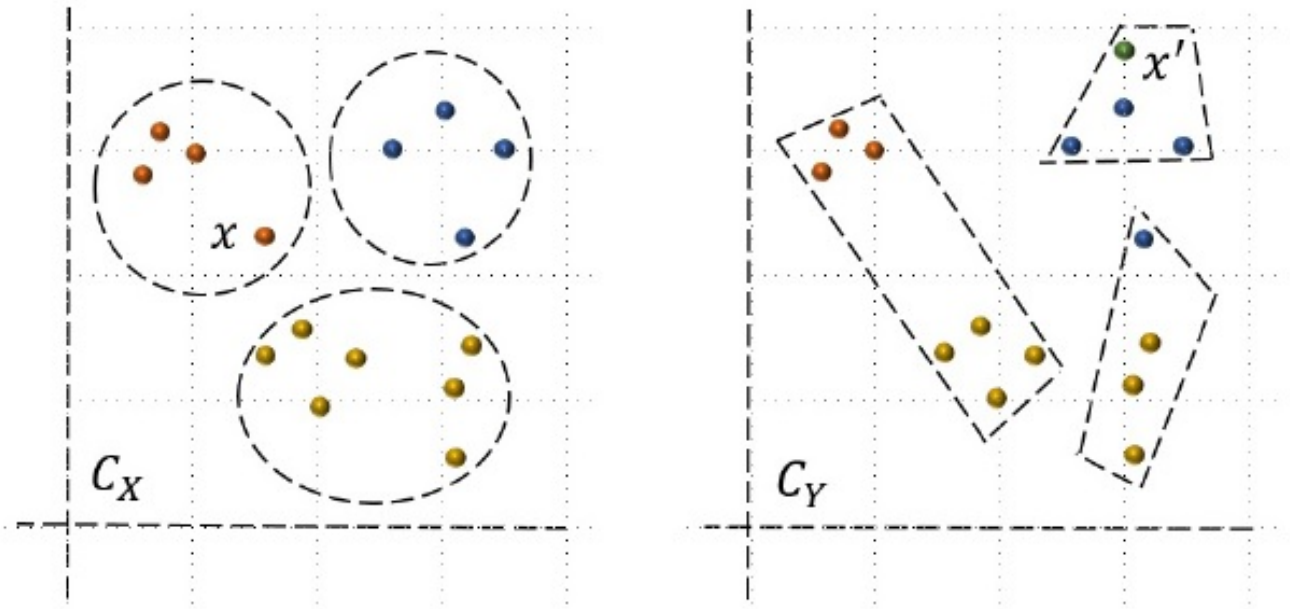


a query f $\xrightarrow{\text{approximated}}$ a query $f \circ \mathcal{M}$

A microaggregation algorithm $\mathcal{M}$ consists of two phases:



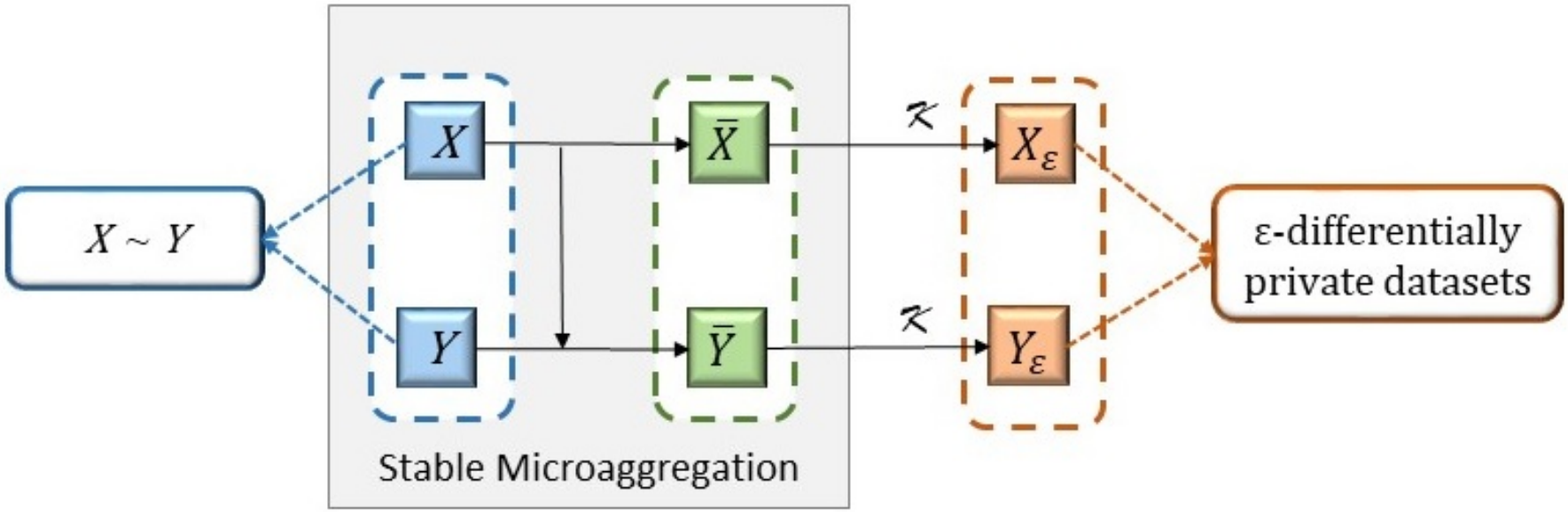
An arbitrary $\mathcal{M}$ could not reduce sensitivity when incorporated into DP.



- $\mathcal{M}$ may generate considerably different clusters.
- Leading to a much larger $\Delta(f \circ \mathcal{M})$ than $\Delta(f)$.

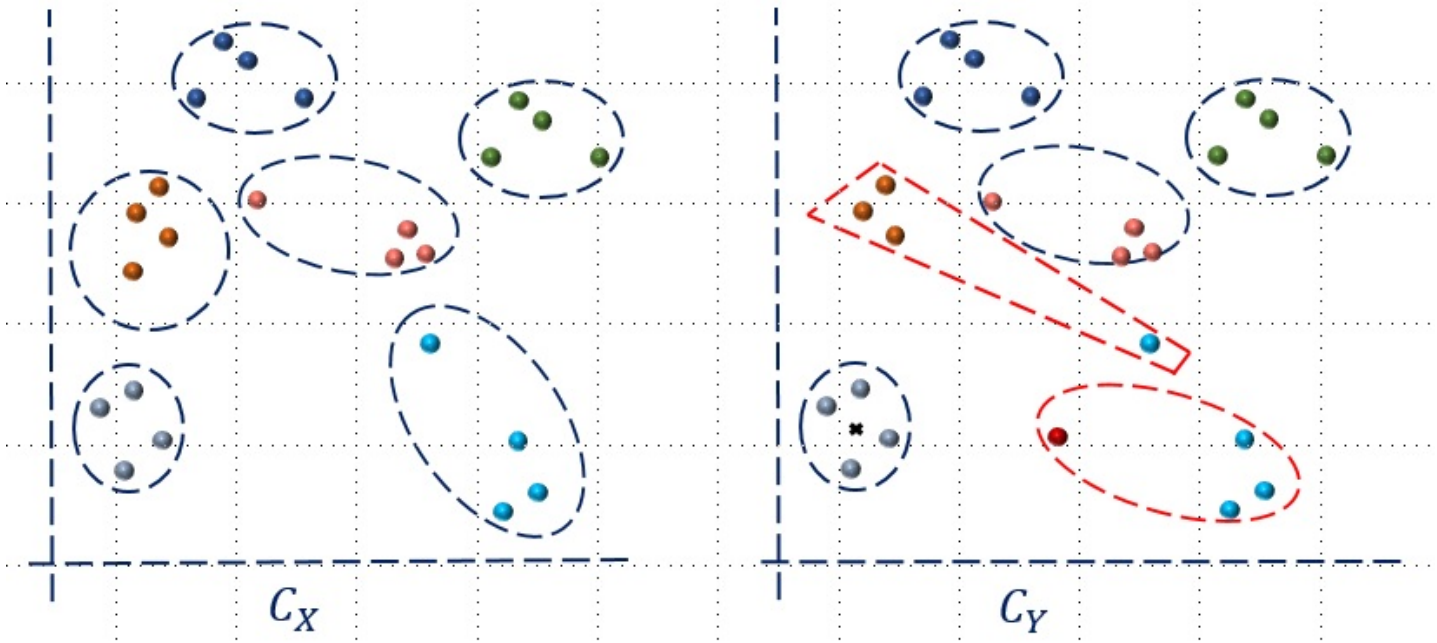
Proposed Framework

Stable microaggregation characterizes a certain correspondence of clusters in microaggregated datasets to control sensitivity.



Stable microaggregation

There is a bijection between $\mathcal{C}_X$ and $\mathcal{C}_Y$ such that at most two pairs of corresponding clusters differ in a single record.



- $\Delta(f \circ \mathcal{M})$ is reduced to $(2 \times \Delta(f)/k)$ from $(n/k \times \Delta(f))$.
- Needs $k \geq 2$ to reduce noise regardless of dataset size n.

α-STABLE MICROAGGREGATION

Australian National University

We proposed a unified framework called α-stable microaggregation which generalized stable microaggregation:

- At most α pairs of corresponding clusters in $\mathcal{C}_X$ and $\mathcal{C}_Y$ differ in a single record to enhance within cluster homogeneity.
- α indicates the trade-off between error introduced during microaggregation and achieving DP.
- $\Delta(f \circ \mathcal{M})$ is $(\alpha \times \Delta(f)/k)$.

α-STABLE MICROAGGREGATION ALGORITHMS

Australian National University

1. **Sequential α-stable microaggregation algorithm:** Performs record-level search such that swapping records leads to enhance within cluster homogeneity.

2. **Decisional α-stable microaggregation algorithm:** Performs cluster-level search followed by record-level within the selected cluster to enhance within cluster homogeneity.

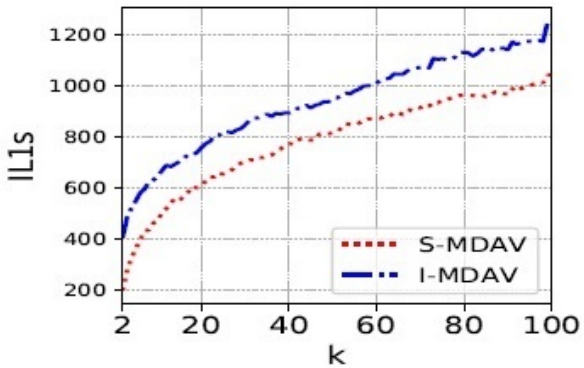
Experimental Setup

Australian National University

- Three datasets:
  (1) *CENSUS* contains 1,080 records.
  (2) *EIA* contains 4,092 records.
  (2) *Tarragona* contains 834 records.
- Two measures:
  - *IL1s* measures *information loss* between the original and differentially private datasets [8].
  - *RL* measures the percentage of record linkage between the original and differentially private datasets [7].

Experiments I

Does the proposed framework yield less *IL1s* and RL in microaggregated datasets?

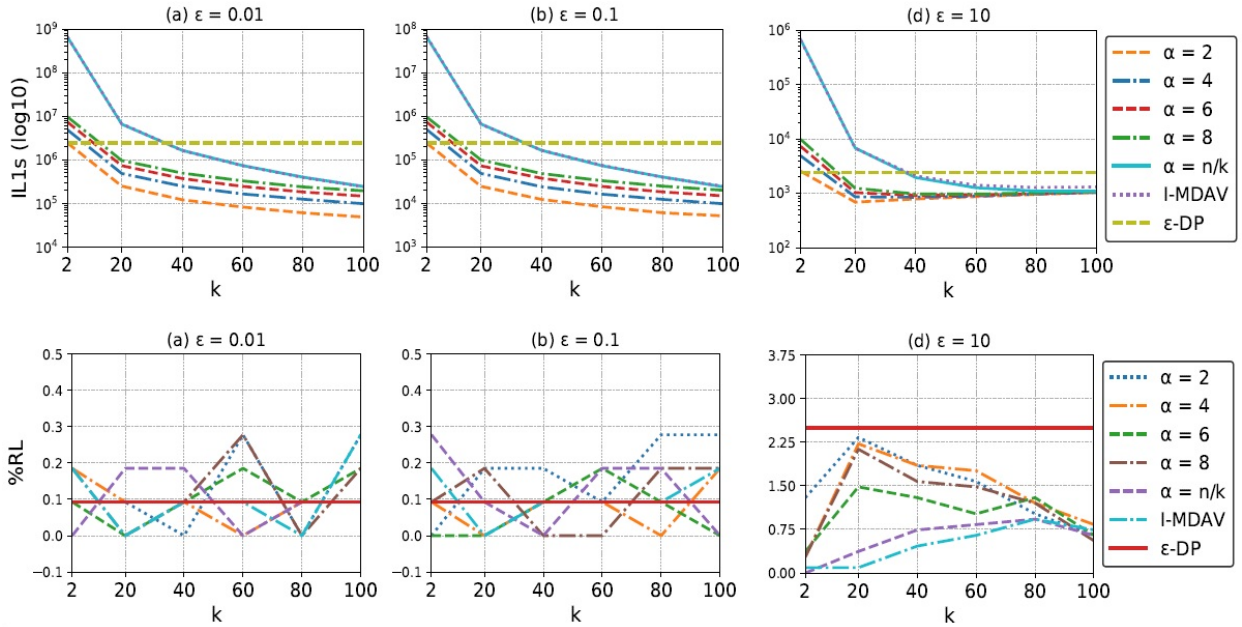


k	$\alpha = 2$				$\alpha = n/k$				IM	
	SαS		DαS		SαS		DαS			
	IL1s	%RL	IL1s	%RL	IL1s	%RL	IL1s	%RL	IL1s	%RL
2	191.04	34.7	190.33	34.5	189.51	35	190.22	34.4	398.08	14.3
20	611.08	4.44	610.165	4.44	611.08	4.44	610.37	4.44	757.86	**4.26**
40	770.08	2.31	771.062	2.31	770.16	2.31	771.06	2.31	893.37	2.31
60	864.82	1.57	868.051	1.57	868.94	1.57	868.47	1.57	1010.7	1.67
80	958.34	1.2	960.153	1.2	959.25	1.2	960.4	1.2	1130.6	1.11
100	1038.5	0.93	1040.51	0.93	1039.3	0.83	1040.5	0.93	1243.9	0.93

- Significantly reduced *IL1s* during microaggregation by providing better within cluster homogeneity.
- Attain RL below 5% for $k \geq 20$.

Experiments II

Does the proposed framework yield less *IL1s* and RL in differentially private datasets?



- Reduced *IL1s* as $\Delta(f \circ \mathcal{M})$ is $\alpha \times \Delta(f)/k$.
- When $k \geq \alpha$, noise is reduced regardless of n.
- Attain RL below 5%

Discussion

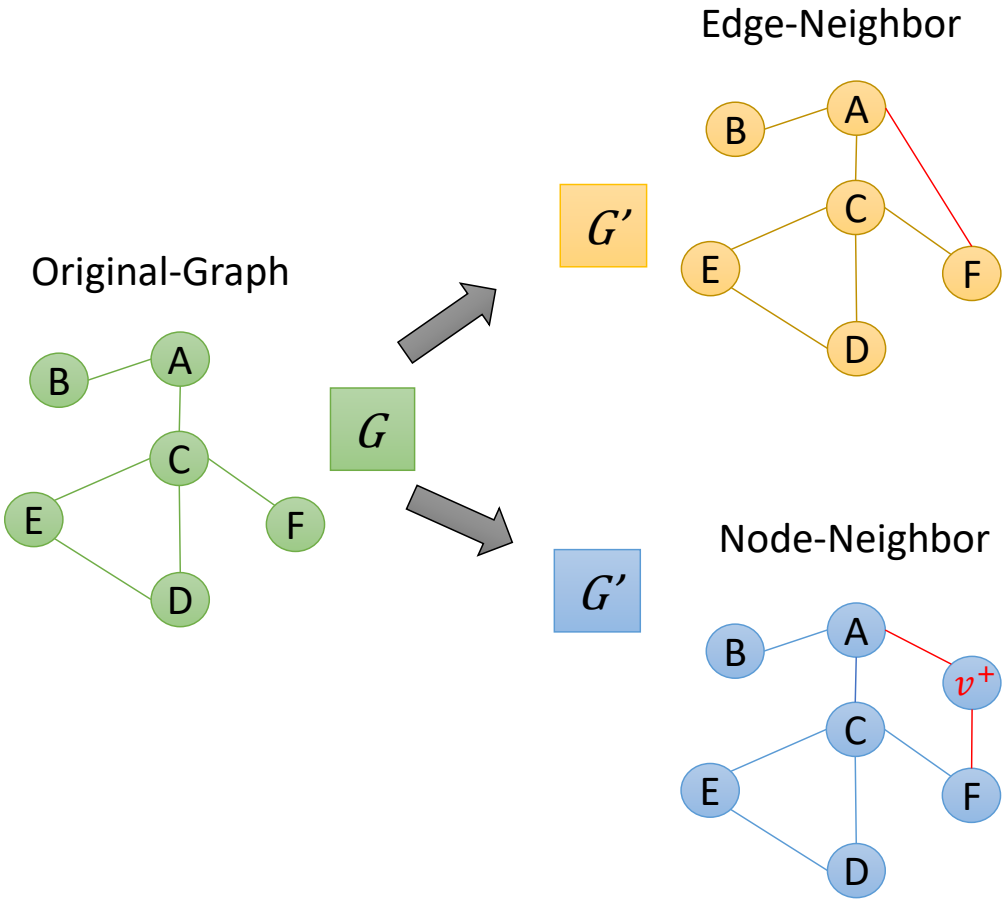
Australian National University

What kind of trade-off exists between utility and privacy while generating differentially private datasets?

- Error caused by DP that depend on $\Delta(f \circ \mathcal{M})$ dominates the impact on data utility as compared to microaggregation error.
- Reducing sensitivity can increase the data utility but it is not straightforward.
- Adding more noise provides better privacy but less utility and vise versa.

Graph Data Publishing

Neighboring Graphs



G and *G'* differ in one edge.

G and *G'* differ in one node and the set of edges incident to that node.

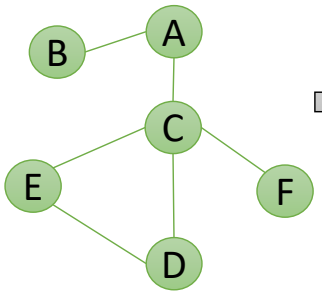
Key Challenges

- Graph data is highly sensitive to structural changes.
- Directly perturbing graph data for achieving DP often leads to inject a large amount of noise.
- Preserving topological structures of an original graph while achieving DP is not straightforward.

Problem Statement

- **Problem:** Given a graph G, we want to publish graph statistics under the guarantee of DP.
- **Key Observation:** dK-distributions can serve as a good basis for representing graph statistics. The dK-graph model [5] provides a systematic way of extracting dK-distributions from G.

dK-distributions are a set of reproducible graph properties, which capture degree correlations within *d*-sized subgraphs of a graph.



$1K(G)$

1	2
2	3
4	1

Degree Distribution

$2K(G)$

<1,2> = 1 (B-A)
<1,4> = 1 (F-C)
<2,2> = 1 (E-D)
<2,4> = 3 (A-C), (D-C), (E-C)

Joint Degree Distribution

$3K(G)$

<1,2,4> = 2 (B-A-C), (F-C-A)
<2,4,2> = 3 (A-C-D), (A-C-E), (D-C-E)

Clustering Coefficient Distribution

- When $d = |V|$, a dK-distribution specifies the entire graph.
- $\gamma^{dK}(G)$ queries the *dK*-distribution of *G*.

Variants of DP

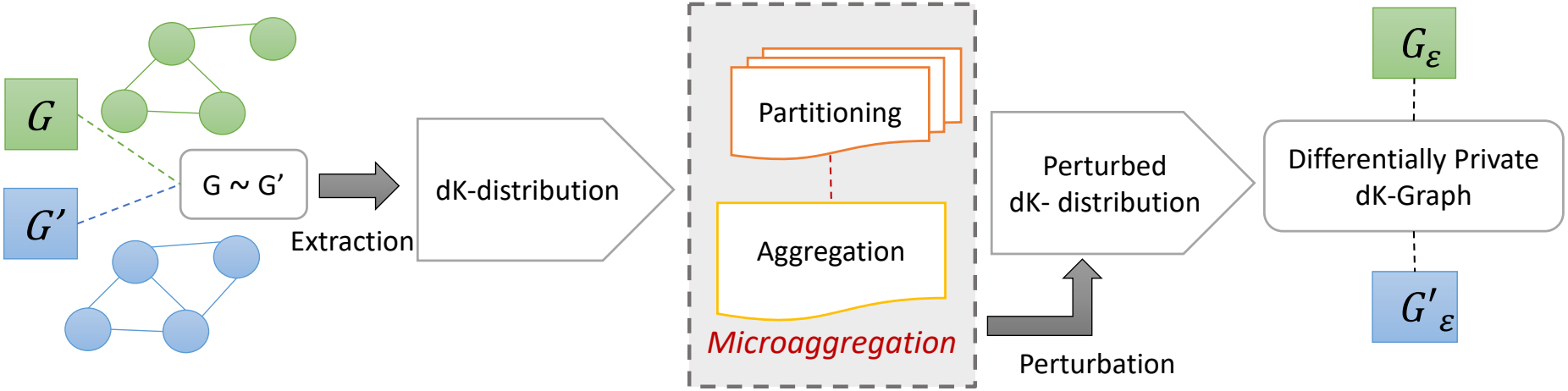
DP has two variants when applying to graph data:

- **Edge-DP:** Hide the presence and absence of a single edge in a graph.
- **Node-DP:** Hide the presence and absence of a single node and the set of edges incident to that node.

Node-DP can provide stronger privacy protection than edge-DP.

Achieving node-DP is more challenging than for edge-DP as graph data is highly sensitive under node-DP.

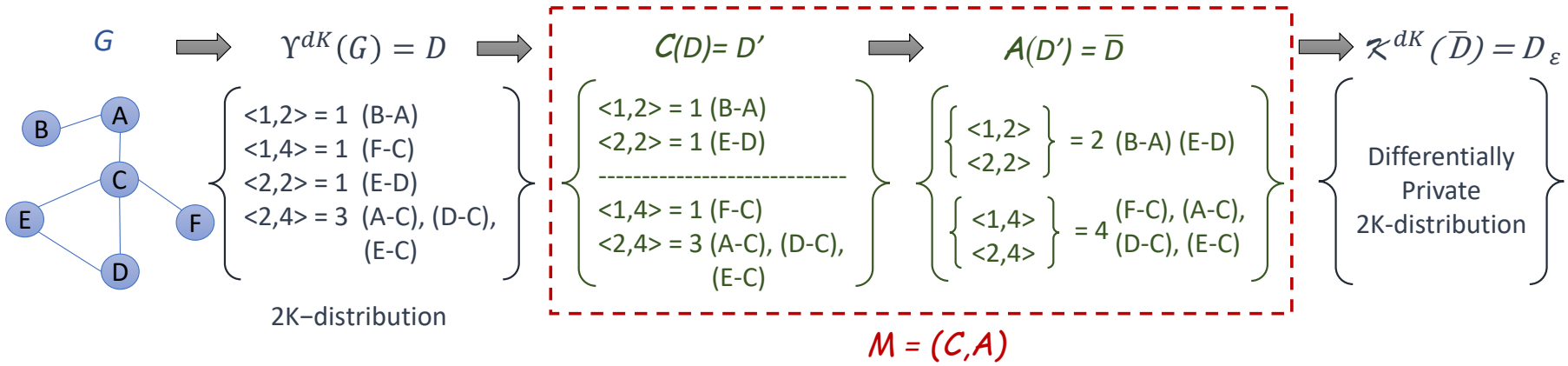
dK-Microaggregation Framework: Microaggregation helps to reduce the overall noise needed to achieve edge-DP.



- $\gamma^{dK} \circ \mathcal{M}$ is $(4 \times g + 1) \times n$, where $g = max(\{deg(G), deg(G')\})$, and n is the number of clusters generated by $\mathcal{M}$.

Proposed Microaggregation Algorithms

A microaggregation algorithm partitions *dK*-distribution into clusters and then aggregates frequency values of tuples in each cluster.



1. **MDAV-dK algorithm:** Partition *dK*-distribution with a fixed-size constraint such that each cluster has at least *k* tuples.
2. **MPDC-dK algorithm:** Partition *dK*-distribution such that every pair of tuples in a cluster satisfies a distance constraint.

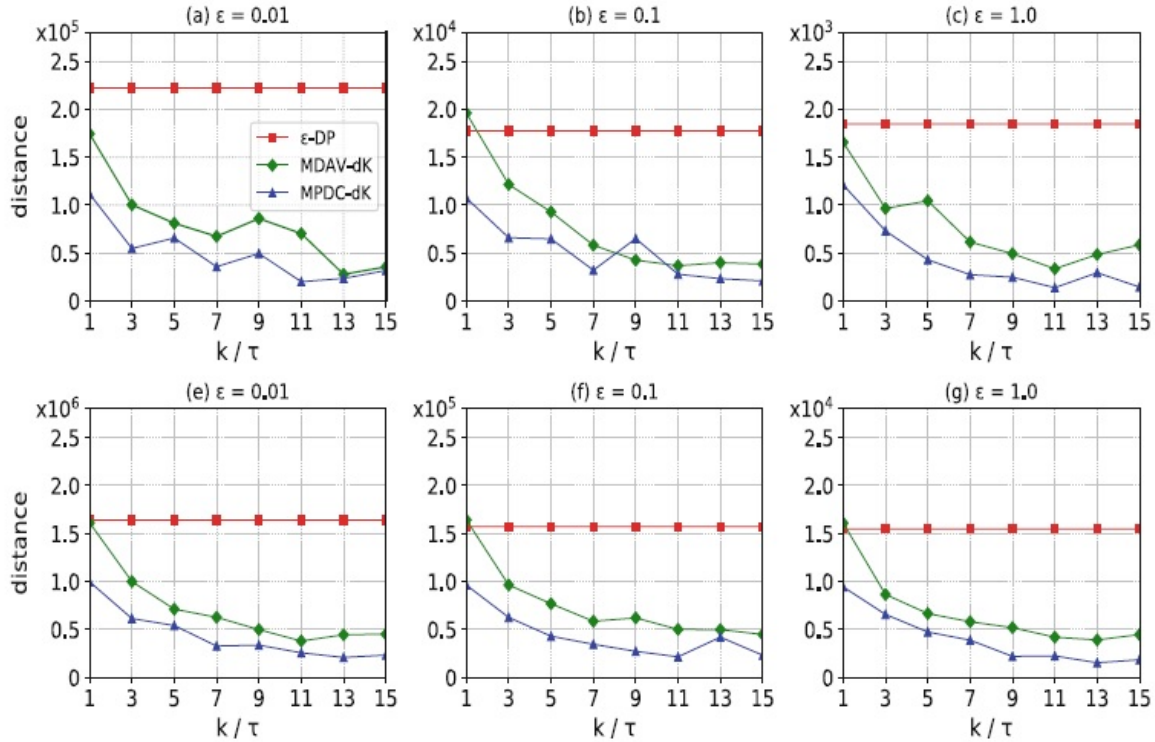
Experimental Setup

Australian National University

- Three network datasets:
  (1) *polbooks* contains 105 nodes and 441 edges.
  (2) *ca-GrQc* contains 5,242 nodes and 14,496 edges.
  (3) *ca-HepTh* contains 9,877 nodes and 25,998 edges.
- Two measures:
  - Euclidean distance measures network structural error between original and perturbed dK-distributions [6].
  - Sum of absolute error measures within-cluster homogeneity of clustering algorithms [3] .

Experiments I

Does the proposed framework reduce the amount of noise added into dK-distributions while still providing edge-DP guarantee?



- Lead to less structural error.
- Introduce overall less noise to achieve edge-DP.

Experiments II

How do our microaggregation algorithms perform in providing better within cluster homogeneity for dK-distributions?

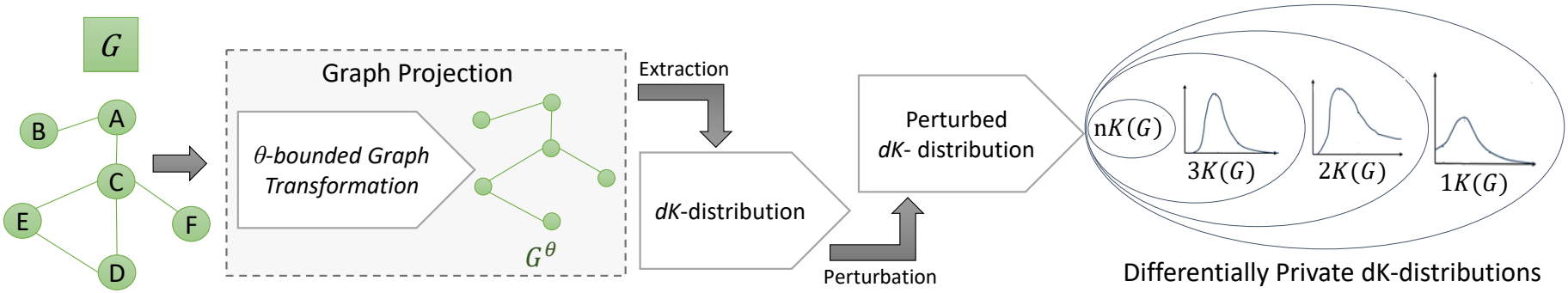
Datasets	Measures	$k=1$	$k=3$	$k=5$	$k=7$
polbooks	SAE	0	144.6	184.67	**224.84**
	# Clusters	161	53	32	**23**
ca-GrQc	SAE	0	1073.3	1476	**1810.5**
	# Clusters	1233	411	246	**176**
ca-HepTh	SAE	0	968.72	1304	1599.8
	# Clusters	1295	431	259	185

Datasets	Measures	$\tau=1$	$\tau=3$	$\tau=5$	$\tau=7$
polbooks	SAE	90.72	**192.15**	328.96	424.2
	# Clusters	68	**25**	13	8
ca-GrQc	SAE	725.38	**1732.1**	2630.6	3470.6
	# Clusters	483	**178**	98	61
ca-HepTh	SAE	841.87	**1761.8**	2773.3	3721.4
	# Clusters	412	**140**	73	37

- Produce clusters with less sum of absolute error.
- Reduce error due to microaggregation.

Achieving Node-DP

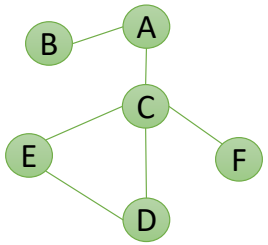
dK-Projection Framework: Projection helps to reduce sensitivity by bounding maximum degree in G.



- $\gamma^{2K} \circ \mathcal{P}$ is reduced to $(2 \times \theta + 1) \times \theta$ from $(2 \times deg(G) + 1) \times |E^{+}|$

The algorithm projects a graph using a two-level ordering: (i) global node ordering, and (ii) local neighborhood ordering.

- Assume a sequence of edges ordered by two-level ordering, and let $\theta = 1$.



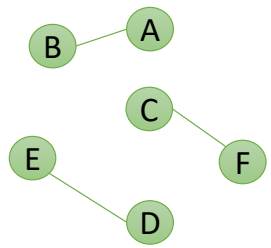
Original Graph

v	deg(v)	N(v)
C	4	{A, D, E, F}
A	2	{C, B}
D	2	{C, E}
E	2	{C, D}
B	1	{A}
F	1	{C}

v	deg(v)	N(v)
C	3	{D, E, F}
A	1	{B}
D	2	{C, E}
E	2	{C, D}
B	1	{A}
F	1	{C}

v	deg(v)	N(v)
C	2	{E, F}
A	1	{B}
D	1	{E}
E	2	{C, D}
B	1	{A}
F	1	{C}

v	deg(v)	N(v)
C	1	{F}
A	1	{B}
D	1	{E}
E	1	{D}
B	1	{A}
F	1	{C}



After Stable-Edge-Removal

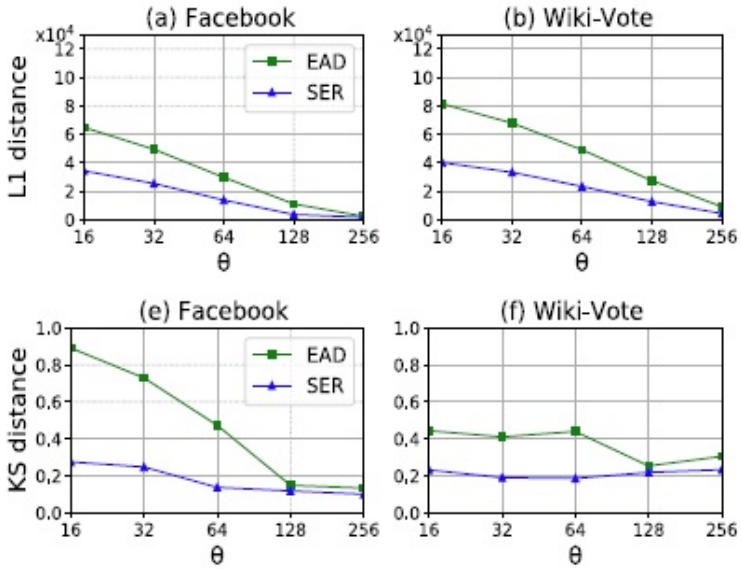
Experimental Setup

Australian National University

- Four network datasets:
  (1) *Facebook* contains 4,039 nodes and 88,234 edges.
  (2) *Wiki-Vote* contains 7,115 nodes and 103,689 edges.
  (3) *Ca-HepPh* contains 12,008 nodes and 118,521 edges.
  (4) *Email-Enron* contains 36,692 nodes and 183,831 edges.
- Three utility metrics [1]:
  - *Preserved edge ratio* measures the ratio of edges being preserved by graph projection.
  - *L1 distance* measures the network structural error between an original dK-distribution and its perturbed dK-distribution.
  - *KS distance* quantifies the closeness between an original dK-distribution and its perturbed dK-distribution.

Australian National University

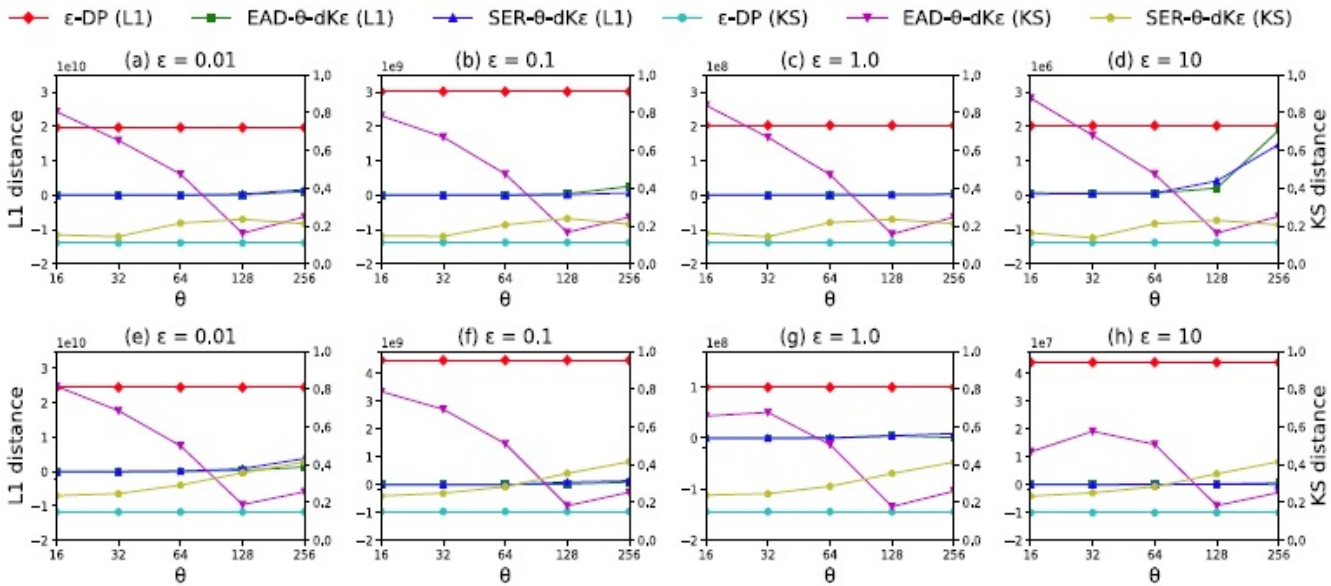
Does the proposed graph projection algorithm yield more utility in projected graphs?



Dataset	$\theta = 16$		$\theta = 32$		$\theta = 64$	
	EAD	SER	EAD	SER	EAD	SER
Facebook	0.27	0.61	0.44	0.71	0.66	0.84
Wiki-Vote	0.19	0.59	0.32	0.66	0.50	0.76
Ca-HepPh	0.16	0.61	0.24	0.68	0.31	0.77
Email-Enron	0.17	0.52	0.22	0.61	0.29	0.71

- Preserves more edges.
- Leads to less network structural error.
- Generates dK-distributions that are more similar to their original dK-distributions.

Does the proposed graph projection algorithm yield more utility in differentially private datasets?



- Yields less network structural error.
- For smaller values of θ, differentially private dK-distributions are more similar to their original dK-distributions.

- **Limitations of DP:**
  - Uniform privacy level (i.e., ε) is assigned to each individual while performing perturbation.
  - DP may lead to provide insufficient protection for some individuals, while over-protecting others.

Personalized differential privacy provides freedom to individuals to set their own privacy parameter ε.

Problem Statement

- **Problem:** Given a graph G, we want to publish graph statistics under the guarantee of DP while considering personalization.
- **Key Challenges:**
  - Graph is a structure of connections between nodes, thus publishing data about one node may leads to violate privacy of others under personalization.
  - Each individual (node) has its own privacy preference whereas each entry in data distribution reflects information about more than one node.

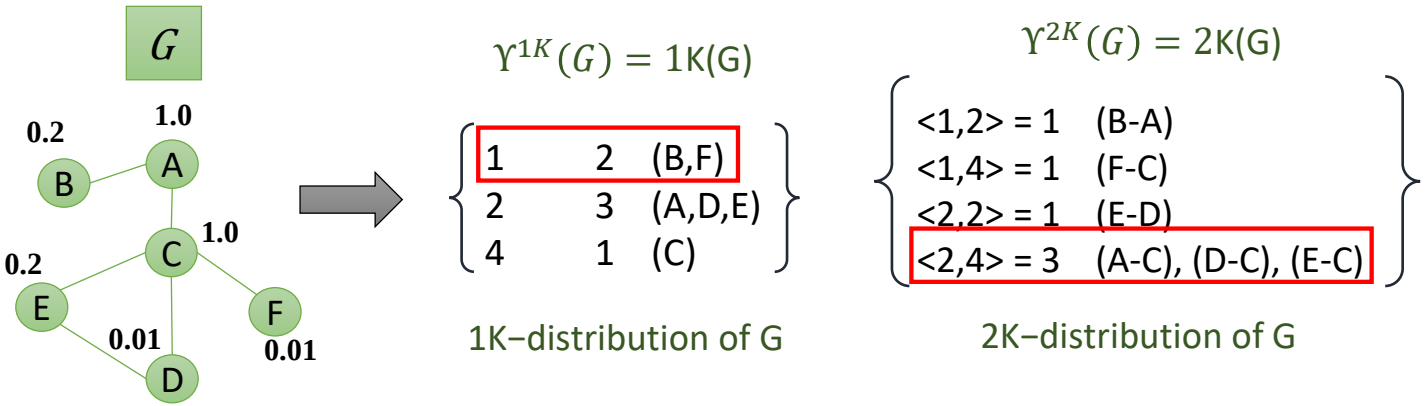
Sensitivity Analysis

We analyze the sensitivity (Δ) of a single dK-distribution entry, i.e., degree query γ_q rather than the entire dK-distribution γ^{dK}.

- $\Delta(\gamma_q)$ of is $|E^+| + 1$ over 1K(G) under node-DP.
- $\Delta(\gamma_q)$ of $(deg(G) + 1) \times |E^+|$ over 2K(G) under node-DP.
- $\Delta(\gamma_q)$ of is 2 over 1K(G) under Edge-DP.
- $\Delta(\gamma_q)$ of is $2 \times deg(G) + 1$ over 2K(G) under Edge-DP.

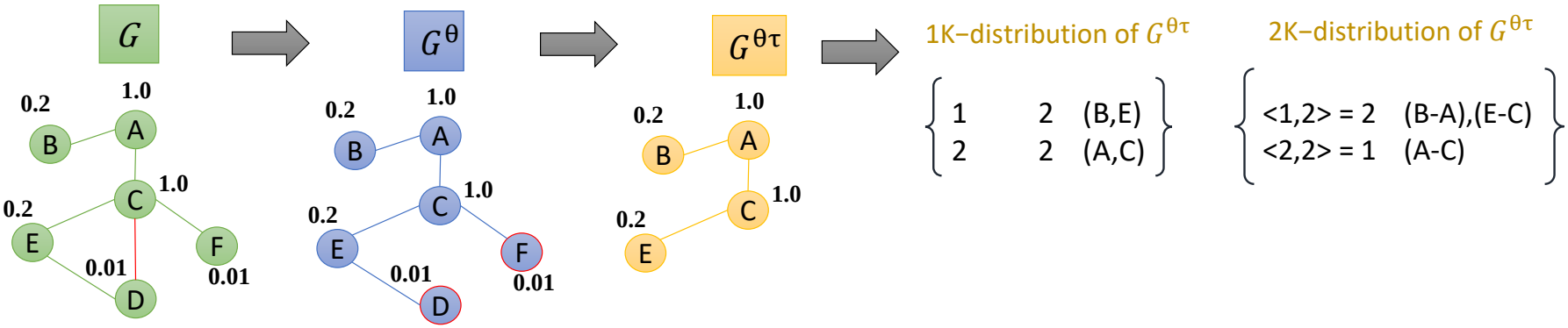
We observe that the sensitivity of γ_q is half as compared to γ^{dK}.

Local Least Based Personalized Perturbation: LL-dK perturbs entries with the strongest local ε.



The frequency value 2 in $1K(G)$, and the frequency value 3 in $2K(G)$ are perturbed with $\varepsilon = min(\Phi^B, \Phi^F)$, and $\varepsilon = min(\Phi^A, \Phi^C, \Phi^D, \Phi^E)$, respectively.

Threshold Projection Based Personalized Perturbation: TP-dK transforms a graph into a θ-bounded graph then removes all nodes with $\varepsilon < \tau$.

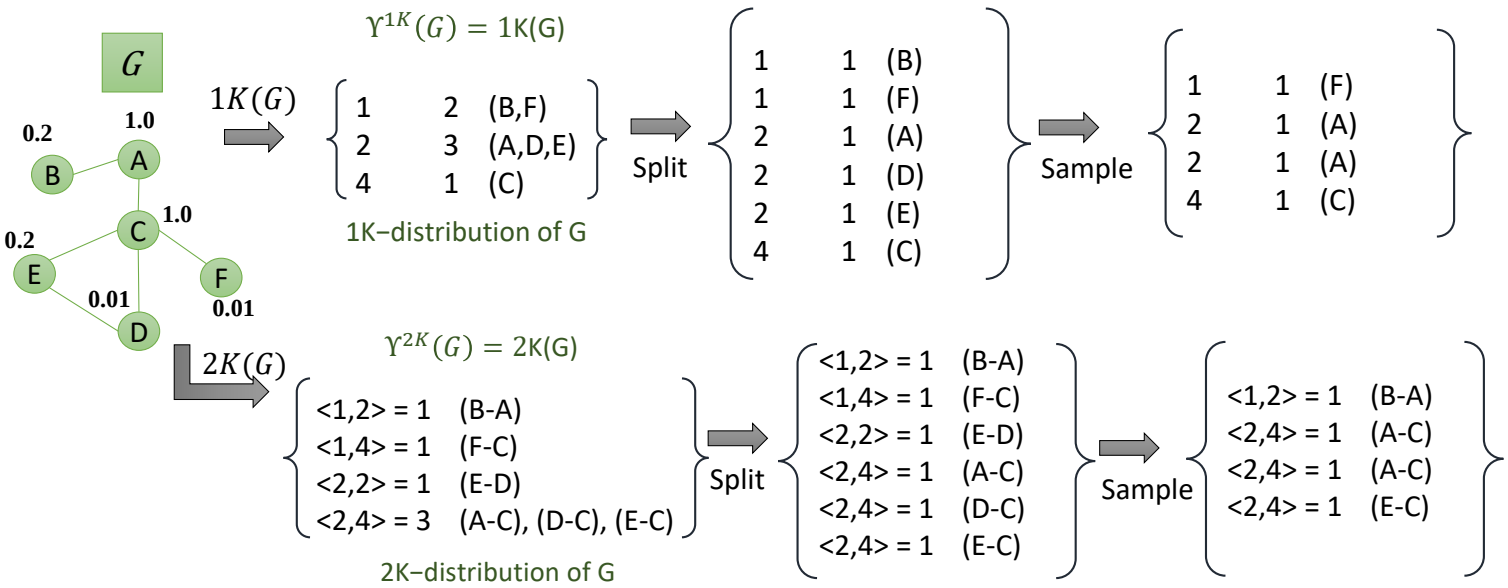


Since $deg(G) \leq \theta$, the sensitivity of γ_q is reduced.

With threshold τ all nodes with high privacy are removed.

Proposed Personalized Approaches III

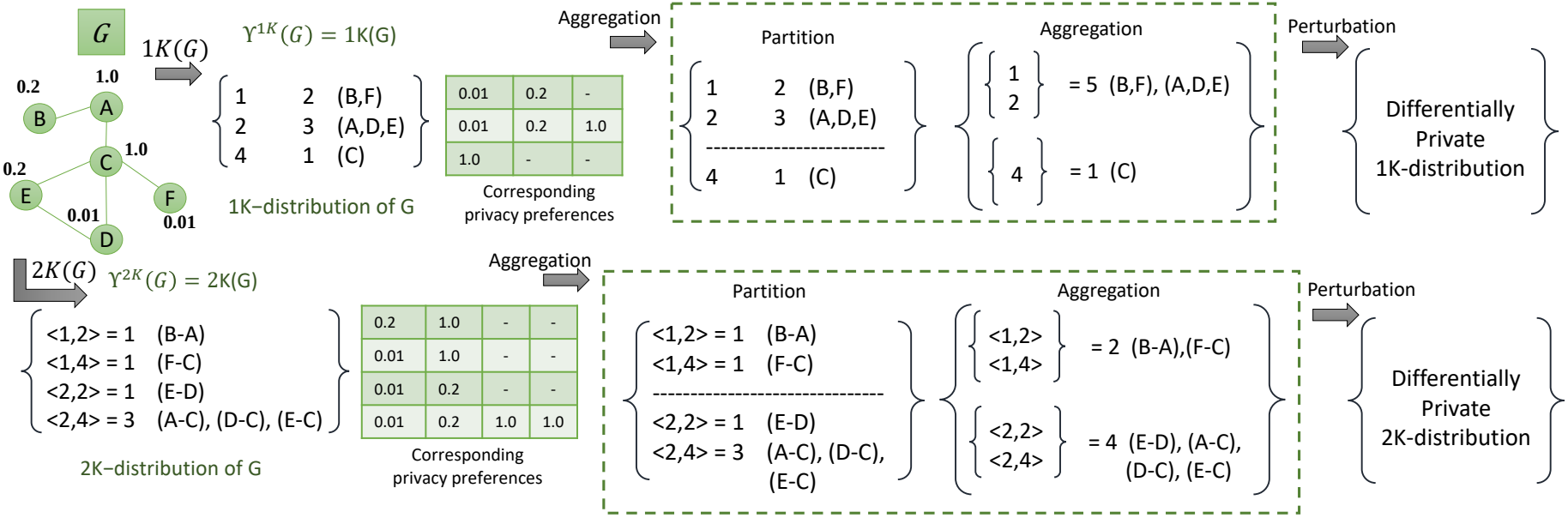
Sampling Based Personalized Perturbation: ST-dK first splits entries, and then samples them with non-uniform probabilities.



Inclusion probability for each entry depends on corresponding ε and global threshold τ.

Sampled dK-distribution is perturbed with τ.

Aggregation Based Personalized Perturbation: AG-dK computes corresponding ε values to performs aggregation over dK-distribution.



Entries are perturbed with the strongest local ε corresponding to each partition.

γ_q is approximated to $\gamma_q \circ \mathcal{M}$.

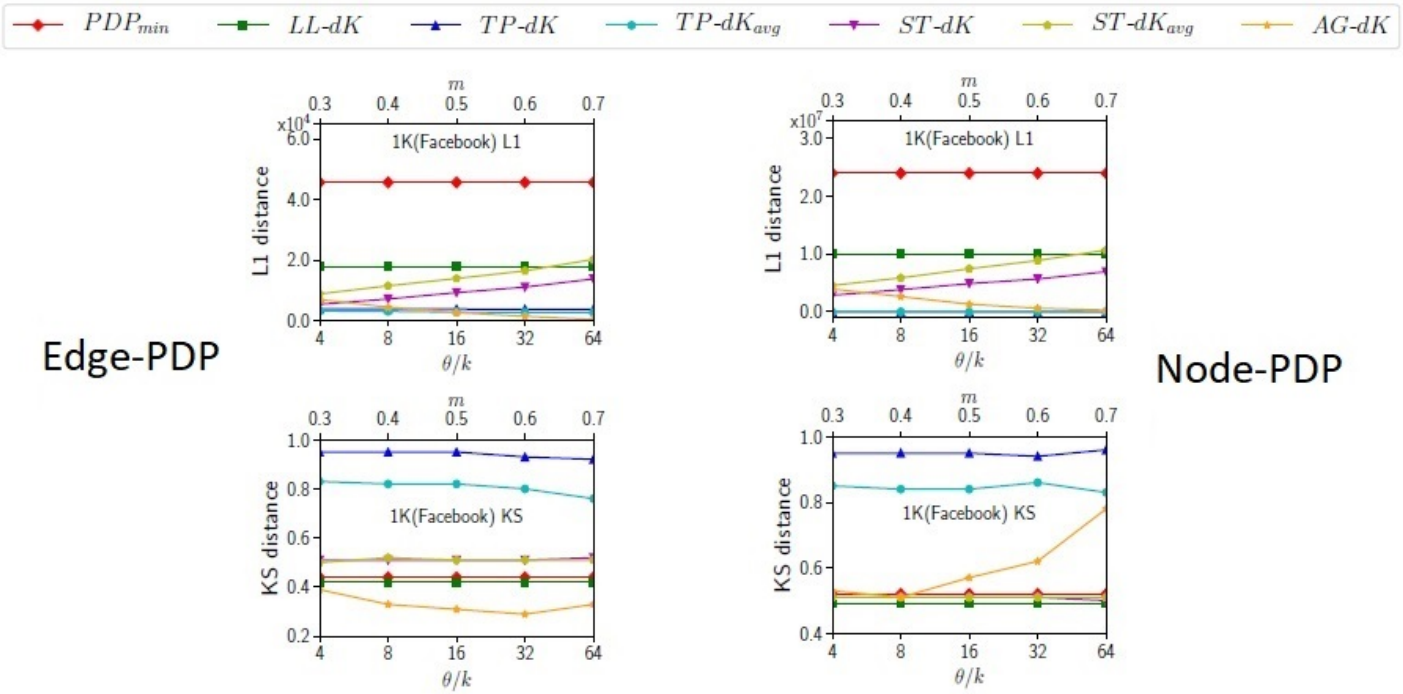
Experimental Setup

Australian National University

- Four network datasets:
  (1) *Facebook* contains 4,039 nodes and 88,234 edges.
  (2) *Wiki-Vote* contains 7,115 nodes and 103,689 edges.
  (3) *Ca-HepPh* contains 12,008 nodes and 118,521 edges.
  (4) *Email-Enron* contains 36,692 nodes and 183,831 edges.
- Two utility metrics [1]:
  - *L1 distance* measures the network structural error between an original dK-distribution and its perturbed dK-distribution.
  - *KS distance* quantifies the closeness between an original dK-distribution and its perturbed dK-distribution.

Experiments I

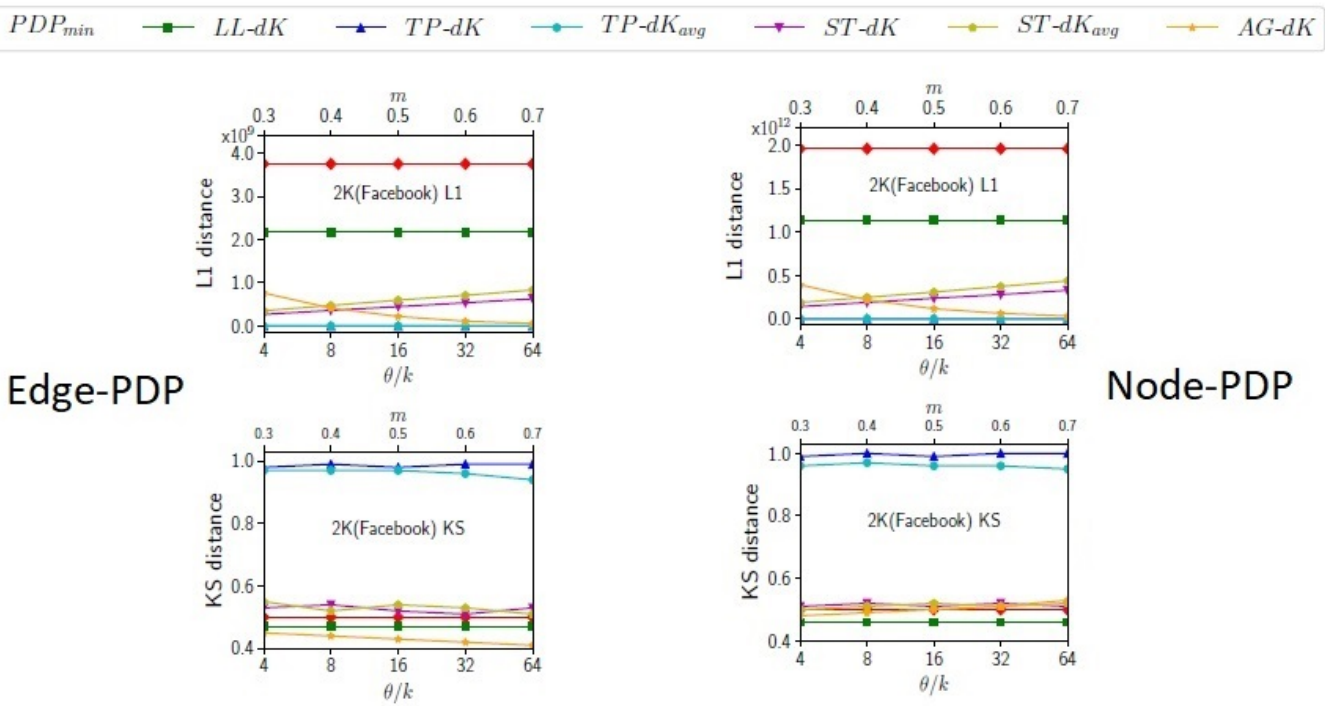
Does the proposed personalized approaches yield more utility in 1K-distribution under edge-PDP and node-PDP?



- Our methods yield less network structural error.
- AG-dK outperforms under edge-PDP and LL-dK outperforms under node-PDP by generating more similar 1K-distributions.

Experiments II

Does the proposed personalized approaches yield yield more utility in 2K-distribution under edge-PDP and node-PDP?



- Our methods yield less network structural error.
- AG-dK outperforms under edge-PDP and LL-dK outperforms under node-PDP by generating more similar 2K-distributions.

Discussion

What kind of trade-off exists between utility and privacy while generating personalized differentially private dK-distributions?

- The error caused by sensitivity (Δ) and the privacy preference ε dominates the impact on output utility.
- Increasing ε and decreasing Δ can help to reduce error.
- Reducing sensitivity is more challenging under node-PDP than for edge-PDP as graph data is highly sensitive under node-DP.

Conclusion

PPDP mechanisms for publishing differentially private data:

- **Relational Data:** We present novel framework that outperforms the state-of-the-art methods in terms of preserving output utility while guarantee DP.
- **Graph Data:** We present novel PPDP mechanisms to publish higher-order graph statistics under edge, node and personalized DP while enhancing output utility.

Future work

- To publish graph statistics under local differential privacy while considering personalization.
- To develop differentially private mechanisms for continual release of graph statistics in dynamic graphs.
- To release statistics about social groups in a network while protecting privacy of individuals under zero knowledge privacy (ZKP).

References

Wei-Yen Day, Ninghui Li, and Min Lyu.
Publishing graph degree distribution with node differential privacy.
In *SIGMOD*, pages 123–138, 2016.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith.
Calibrating noise to sensitivity in private data analysis.
In *TCC*, pages 265–284, 2006.

Vladimir Estivill-Castro and Jianhua Yang.
Fast and robust general purpose clustering algorithms.
In *PRICAI*, pages 208–218, 2000.

Priya Mahadevan, Calvin Hubble, Dmitri Krioukov, Bradley Huffaker, and Amin Vahdat.
Orbis: rescaling degree correlations to generate annotated internet topologies.
In *SIGCOMM*, pages 325–336, 2007.

Priya Mahadevan, Dmitri Krioukov, Kevin Fall, and Amin Vahdat.
Systematic topology analysis and generation using degree correlations.
In *SIGCOMM*, pages 135–146, 2006.

Alessandra Sala, Xiaohan Zhao, Christo Wilson, Haitao Zheng, and Ben Y Zhao.
Sharing graphs using differentially private graph models.
In *SIGCOMM*, pages 81–98, 2011.

Jordi Soria-Comas, Josep Domingo-Ferrer, David Sánchez, and Sergio Martínez.
Enhancing data utility in differential privacy via microaggregation-based k-anonymity.
The VLDB Journal, 23(5):771–794, 2014.

William E Yancey, William E Winkler, and Robert H Creecy.
Disclosure risk assessment in perturbative microdata protection.
In *Inference control in statistical databases*, pages 135–152. 2002.

Australian National University

Thanks for your attention!

Any **Questions**