dK-Projection: Publishing Graph Joint degree distribution with Node Differential Privacy

Masooma Iftikhar
Qing Wang

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

May 11-14, 2021

Agenda

- Introduction
- Problem Formulation
- Sensitivity Analysis
- dK-Projection Framework
- Proposed Approach
- Experiments and Results
- Conclusion and Future Work

Introduction

- Publishing network data may reveal sensitive information of an individual even if the graph is anonymized, thereby requiring *privacy-preserving mechanisms*.

Australian National University

- Publishing network data may reveal sensitive information of an individual even if the graph is anonymized, thereby requiring *privacy-preserving mechanisms*.
- Differential privacy (DP) [3] bounds a shift in the output distribution of a randomized mechanism that can be induced by a small change in its input, preserving individual's privacy.

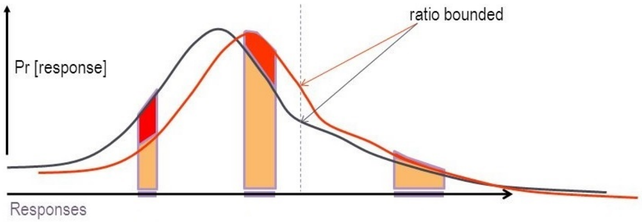


Figure 1: $\mathcal{K}$ gives ε-DP if for all neighboring datasets (differing in just one entry) D_1 and D_2, and all $C \subseteq range(\mathcal{K})$:
$Pr[\mathcal{K}(D_1) \in C] \leq e^{\varepsilon} \, Pr[\mathcal{K}(D_2) \in C]$

- **Aim:** To develop a framework for publishing higher-order network statistics, such as joint degree distribution, under guarantees of node-DP, while enhancing network data utility.

- **Aim:** To develop a framework for publishing higher-order network statistics, such as joint degree distribution, under guarantees of node-DP, while enhancing network data utility.
- **Key Challenge:** To enhance the overall utility of published network statistics, the key challenge is how to reduce the magnitude of noise needed to achieve node-DP by controlling sensitivity effectively.

Australian National University

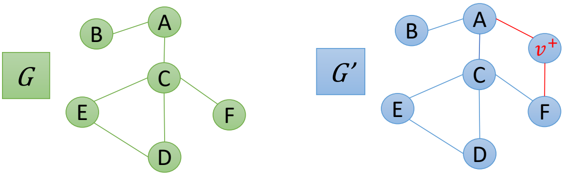
- **Aim:** To develop a framework for publishing higher-order network statistics, such as joint degree distribution, under guarantees of node-DP, while enhancing network data utility.
- **Key Challenge:** To enhance the overall utility of published network statistics, the key challenge is how to reduce the magnitude of noise needed to achieve node-DP by controlling sensitivity effectively.
- **Key Observation:** We observe that dK-distributions [5] can serve as a good basis for representing higher-order network statistics.

Problem Formulation

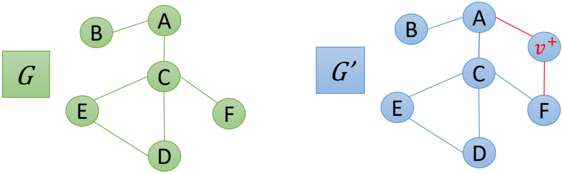
Neighboring graphs

- We define the notion of neighboring graphs under node-DP.

- We define the notion of neighboring graphs under node-DP.



- We define the notion of neighboring graphs under node-DP.



NEIGHBORING GRAPHS

Two graphs $G = (V, E)$ and $G' = (V', E')$ are said to be *neighboring graphs*, denoted as $G \sim G'$, iff $V' = V \cup \{v^+\}$, $E' = E \cup E^+$, and E^+ is the set of all edges incident to v^+ *in* G'.

- Given a graph, we represent its topology properties as *dK*-distributions [5].

- Given a graph, we represent its topology properties as *dK-distributions* [5].

dK-DISTRIBUTION

A *dK-distribution* over a graph $G = (V, E)$, denoted as $dK(G)$, is a probability distribution $p : D^d \rightarrow \mathbb{N}$ such that $p(a_1, \ldots, a_d)$ refers to the total number of connected subgraphs of size d in G with the nodes $\{v_1, \ldots, v_d\}$ and $a_i = deg(v_i)$ for $i = 1, \ldots, d$.

- Given a graph, we represent its topology properties as *dK-distributions* [5].

DK-DISTRIBUTION

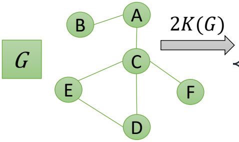
A *dK-distribution* over a graph $G = (V, E)$, denoted as $dK(G)$, is a probability distribution $p : D^d \rightarrow \mathbb{N}$ such that $p(a_1, \dots, a_d)$ refers to the total number of connected subgraphs of size d in G with the nodes $\{v_1, \dots, v_d\}$ and $a_i = deg(v_i)$ for $i = 1, \dots, d$.

- For a graph, 1K-distribution captures the degree distribution, 2K-distribution captures the joint degree distribution. When $d = |V|$, a dK-distribution specifies the entire graph.

- A dK-distribution is extracted from a graph, by using *dK function* (s.t. $\gamma^{dK}(G) = dK(G)$).

- A dK-distribution is extracted from a graph, by using *dK function* (s.t. $\gamma^{dK}(G) = dK(G)$).
- $\gamma^{2K}(G)$ returns the joint degree distribution of G, i.e., $p(i,j)$ is a frequency value, referring to the number of edges connecting nodes of degrees i and j.

$\Upsilon^{2K}(G) =$ 2K(G)

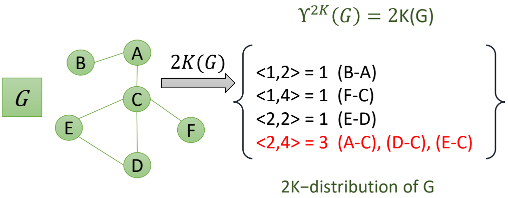


<1,2> = 1 (B-A)
<1,4> = 1 (F-C)
<2,2> = 1 (E-D)
<2,4> = 3 (A-C), (D-C), (E-C)

2K–distribution of G

Australian National University

- A dK-distribution is extracted from a graph, by using *dK function* (s.t. $\gamma^{dK}(G) = dK(G)$).
- $\gamma^{2K}(G)$ returns the joint degree distribution of G, i.e., $p(i, j)$ is a frequency value, referring to the number of edges connecting nodes of degrees i and j.



- For instance, $p(2, 4) = 3$ because G contains 3 edges between 2 degree nodes (i.e., A, D, and E) and 4 degree node (i.e., C)

- To release *dK*-distribution under the guarantees of node-DP, we perturb *dK*-distribution by adding controlled noise from Laplace stochastic process [3].

$$\mathcal{K}(G) = \gamma^{dK}(G) + Lap\left(\frac{\Delta\gamma}{\varepsilon}\right)^{|V|^d}$$

- $\varepsilon > 0$ is the *privacy parameter* (smaller values provide stronger privacy guarantees).
- $\Delta\gamma$ refers to the *sensitivity* of the *dK*-function γ^{dK}, which is the maximum variation in its output, i.e., *dK*-distribution, over two neighboring graphs $G \sim G'$.

- We define the notion of ε-differentially private dK-distribution (i.e., an anonymized version of $\gamma^{dK}(G)$ satisfying differential privacy).

DIFFERENTIALLY PRIVATE dK-DISTRIBUTION

A randomized mechanism $\mathcal{K}$ is ε-differentially private, if for each pair of neighboring graphs $G \sim G'$ and all possible perturbed dK-distributions $\mathcal{D} \subseteq range(\mathcal{K})$, we have:

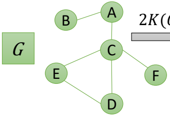
$$Pr[\mathcal{K}(G) \in \mathcal{D}] \leq e^{\varepsilon} \times Pr[\mathcal{K}(G') \in \mathcal{D}]. \quad (1)$$

- The challenge of releasing differentially private dK-distributions is to determine how much noise should be added to perturb dK-distributions.

Sensitivity Analysis

SENSITIVITY ANALYSIS

- Suppose that a node v^+ is added to G with a set E^+ of edges.

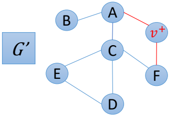


$2K(G)$

$\Upsilon^{2K}(G) = \text{2K(G)}$

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

2K–distribution of G



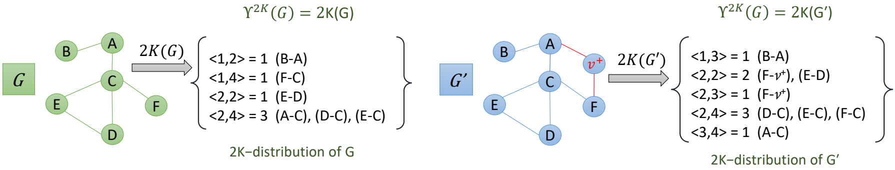
$2K(G')$

$\Upsilon^{2K}(G) = \text{2K(G}')$

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

2K–distribution of G′

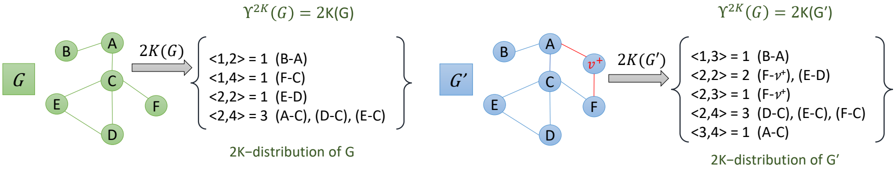
- Suppose that a node v^+ is added to G with a set E^+ of edges.



- **Each edge** $(v^+, v_i) \in E^+$ may cause at most $2 \times deg(G) + 1$ entries of $\gamma^{2K}(G)$ being changed.

Sensitivity Analysis

- Suppose that a node v^+ is added to G with a set E^+ of edges.



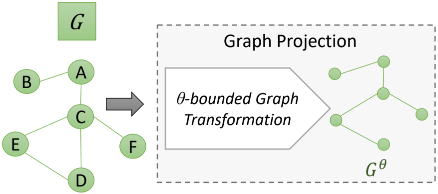
- Each edge $(v^+, v_i) \in E^+$ may cause at most $2 \times deg(G) + 1$ entries of $\gamma^{2K}(G)$ being changed.
- Thus, the total number of entries of $\gamma^{2K}(G)$ being changed by all edges in E^+ is upper bounded by $(2 \times deg(G) + 1) \times |E^+|$.

dK-Projection Framework

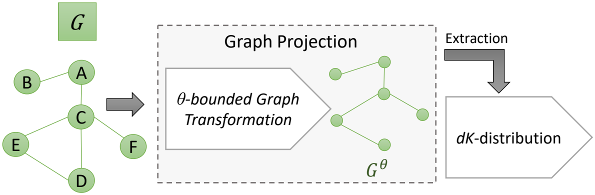
- dK-projection works in the following steps:

- dK-projection works in the following steps:

  (1) Given a graph G, a graph projection algorithm transforms G into a θ-bounded graph G^θ.



- dK-projection works in the following steps:
  (1) Given a graph G, a graph projection algorithm transforms G into a θ-bounded graph G^{θ}.
  (2) Then higher-order network statistics such as dK-distributions [5] are extracted from G^{θ}.



- dK-projection works in the following steps:
  (1) Given a graph G, a graph projection algorithm transforms G into a θ-bounded graph G^{θ}.
  (2) Then higher-order network statistics such as dK-distributions [5] are extracted from G^{θ}.
  (3) Finally extracted dK-distributions are perturbed yielding ε- differentially private dK-distributions.

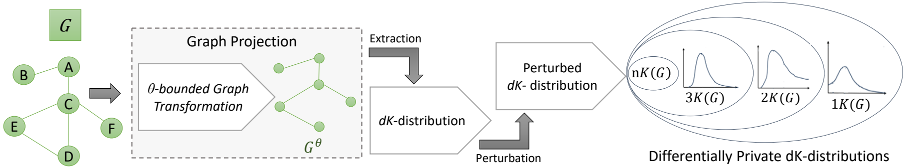


Figure 2: A high-level overview of the proposed framework (*dK-Projection*)

Proposed Approach

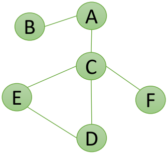
- We propose *Stable-Edge-Removal* (SER) that transform a graph G to a θ-bounded graph G^{θ} with $\theta < deg(G)$ based on a two-level ordering strategy on G.

Two-Level Ordering

A *two-level ordering* over $G = (V, E)$ is a pair $\Gamma = (\succ_N, \succ_V)$ where $\succ_N$ is a *local neighbour ordering* such that, for each $v \in V$, there is a bijection: $N_G(v) \rightarrow \{1, \ldots, |N_G(v)|\}$; $\succ_V$ is a *global node ordering* such that there is a bijection: $V \rightarrow \{1, \ldots, |V|\}$.

- Given a two-level ordering Γ, an edge ordering is defined.

- Assume that a two-level ordering $\Gamma = (\succ_N, \succ_V)$ on a graph G obtained by sorting nodes based on degrees from highest to lowest ($\succ_V$), and for each node v sorting their neighbours in $N_G(v)$ in a similar manner ($\succ_N$).

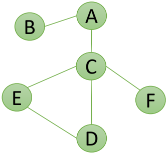


Original Graph

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}

Australian National University

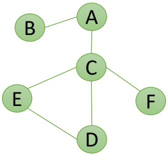
- Thus, we have a sequence of edges ordered by $\succ_\Gamma$, i.e., $\langle (C, A), (C, D), (C, E), (C, F), \ldots, (F, C) \rangle$. Let $\theta = 1$.



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}

- Then, following this sequence, by checking whether $deg(C) > \theta$, *SER* first removes edge (C, A) and decreases the degree counts of nodes C and A by 1.



Original Graph

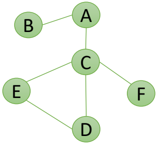
v	deg(v)	N(v)
C	4	{A, D, E, F}
A	2	{C, B}
D	2	{C, E}
E	2	{C, D}
B	1	{A}
F	1	{C}

→

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}

Stable-Edge-Removal Algorithm

- Similarly, *SER* removes edge (C, D) and decreases the degree counts of nodes C and D by 1.



Original Graph

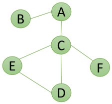
v	deg(v)	N(v)
C	4	{A, D, E, F}
A	2	{C, B}
D	2	{C, E}
E	2	{C, D}
B	1	{A}
F	1	{C}

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}

STABLE-EDGE-REMOVAL ALGORITHM

- *SER* keeps on removing edges, following the edge ordering $\succ_\Gamma$, and decreases the degree counts of nodes $v \in V$ by 1, until G^θ is obtained.



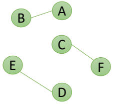
Original Graph

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

- Given a graph G, instead of extracting a dK-distribution from G directly, we extract a dK-distribution from a θ-bounded graph G^{θ} generated by a graph projection algorithm $\mathcal{P}$, here $\mathcal{P}$ refers to our *SER* algorithm.

- Given a graph G, instead of extracting a dK-distribution from G directly, we extract a dK-distribution from a θ-bounded graph G^θ generated by a graph projection algorithm $\mathcal{P}$, here $\mathcal{P}$ refers to our *SER* algorithm.
- Then based on the sensitivity of $\gamma^{dK} \circ \mathcal{P}$, i.e., $(2\theta + 1) \times \theta$ the perturbation is performed over the dK-distribution being extracted from G^θ to generate a ε-differentially private joint degree distribution.

Experiments and Results

- 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.

- 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 [2, 4, 6]:
  - *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 p and its perturbed dK-distribution p'.
  - *KS distance* quantifies the closeness between an original dK-distribution p and its perturbed dK-distribution p'.

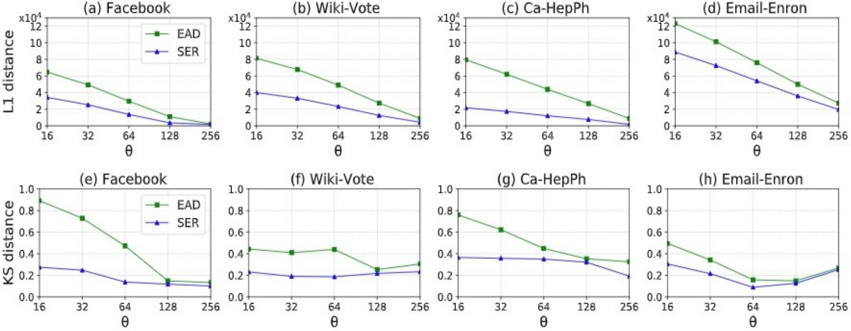
Australian National University

- We first compare our method *SER* with the state-of-the-art graph projection method *EAD* [2], in terms of *preserved edge ratio*. For every value of θ, *SER* outperforms *EAD* by preserving more edges over all four datasets.

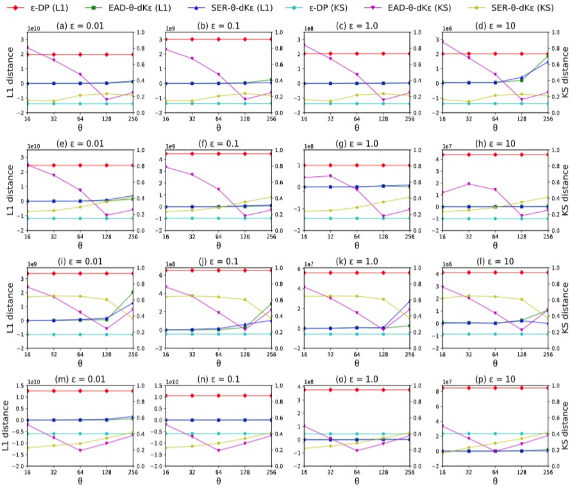
Table 1: Comparison on the preserved edge ratio $|E^{\theta}|/|E|$ of EAD and our proposed SER graph projection approach under different values of θ.

Dataset	$\theta = 16$		$\theta = 32$		$\theta = 64$		$\theta = 128$		$\theta = 256$	
	EAD	SER	EAD	SER	EAD	SER	EAD	SER	EAD	SER
Facebook	0.27	0.61	0.44	0.71	0.66	0.84	0.88	0.96	0.97	0.98
Wiki-Vote	0.19	0.59	0.32	0.66	0.50	0.76	0.71	0.87	0.88	0.96
Ca-HepPh	0.16	0.61	0.24	0.68	0.31	0.77	0.39	0.84	0.46	0.96
Email-Enron	0.17	0.52	0.22	0.61	0.29	0.71	0.36	0.80	0.43	0.89

- We also compare our method *SER* with graph projection method *EAD* [2], in terms of *L1 distance* and *KS distance*. For all four datasets, our projection method *SER* leads to less network structural error and generates *dK*-distributions which are more similar to their original *dK*-distributions for every value of θ as compared to *EAD*.



- We compare the overall utility of differentially private *dK*- distributions generated by our method against the baseline methods.



Conclusion and Future work

- **Conclusion:**
  - Developed a novel framework, called *dK*-Projection to publish higher-order network statistics such as *joint degree distribution* under node-DP.

- **Conclusion:**
  - Developed a novel framework, called *dK*-Projection to publish higher-order network statistics such as *joint degree distribution* under node-DP.
  - Analysed the sensitivity of publishing *joint degree distribution* in the proposed framework.

- **Conclusion:**
  - Developed a novel framework, called *dK*-Projection to publish higher-order network statistics such as *joint degree distribution* under node-DP.
  - Analysed the sensitivity of publishing *joint degree distribution* in the proposed framework.
  - Introduced a new graph projection algorithm to reduce sensitivity of publishing network statistics under node-DP.

- **Conclusion:**
  - Developed a novel framework, called *dK*-Projection to publish higher-order network statistics such as *joint degree distribution* under node-DP.
  - Analysed the sensitivity of publishing *joint degree distribution* in the proposed framework.
  - Introduced a new graph projection algorithm to reduce sensitivity of publishing network statistics under node-DP.
  - Conducted experiments to verify the utility enhancement and privacy guarantee of our proposed framework on four real-world networks.

- **Conclusion:**
  - Developed a novel framework, called *dK*-Projection to publish higher-order network statistics such as *joint degree distribution* under node-DP.
  - Analysed the sensitivity of publishing *joint degree distribution* in the proposed framework.
  - Introduced a new graph projection algorithm to reduce sensitivity of publishing network statistics under node-DP.
  - Conducted experiments to verify the utility enhancement and privacy guarantee of our proposed framework on four real-world networks.

- **Future work:** Future extensions to this work will consider personalized differential privacy to release statistics about social networks while protecting privacy of individuals based on individuals preferences.

REFERENCES

- JEREMIAH BLOCKI, AVRIM BLUM, ANUPAM DATTA, AND OR SHEFFET.
DIFFERENTIALLY PRIVATE DATA ANALYSIS OF SOCIAL NETWORKS VIA RESTRICTED SENSITIVITY.
In *ITCS*, pages 87–96, 2013.

- 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.

- SHIVA PRASAD KASIVISWANATHAN, KOBBI NISSIM, SOFYA RASKHODNIKOVA, AND ADAM SMITH.
ANALYZING GRAPHS WITH NODE DIFFERENTIAL PRIVACY.
In *TCC*, pages 457–476. Springer, 2013.

- PRIYA MAHADEVAN, DMITRI KRIOUKOV, KEVIN FALL, AND AMIN VAHDAT.
SYSTEMATIC TOPOLOGY ANALYSIS AND GENERATION USING DEGREE CORRELATIONS.
In *SIGCOMM*, pages 135–146, 2006.

- SOFYA RASKHODNIKOVA AND ADAM SMITH.
EFFICIENT LIPSCHITZ EXTENSIONS FOR HIGH-DIMENSIONAL GRAPH STATISTICS AND NODE PRIVATE DEGREE DISTRIBUTIONS.
CoRR/1504.07912, 2015.

Australian National University

Thanks for your attention!

Any **Questions**