Analyzing Private Network Data

Gerome Miklau

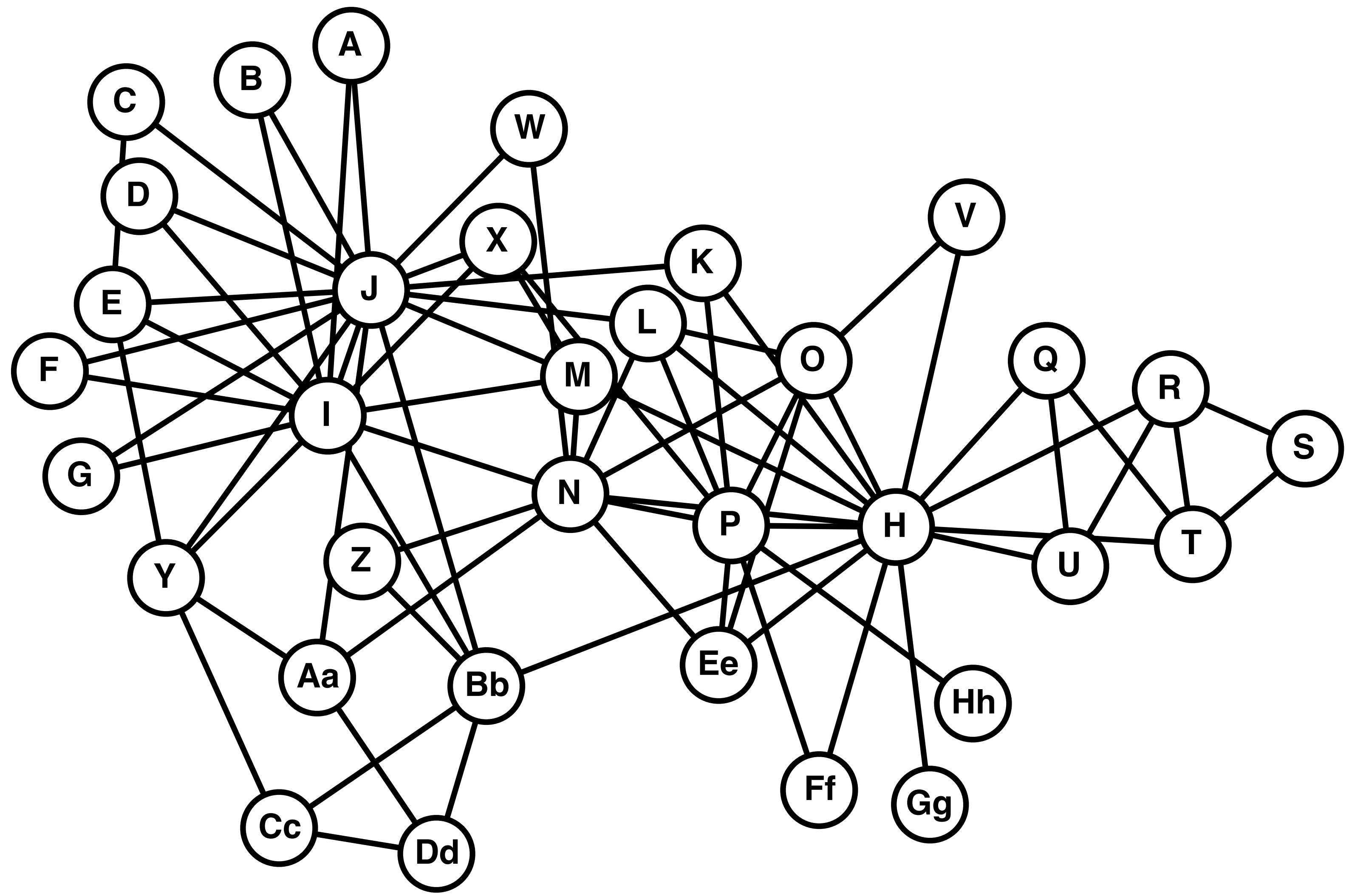
Joint work with

Michael Hay, Chao Li, David Jensen, Don Towsley
University of Massachusetts, Amherst

Vibhor Rastogi, Dan Suciu
University of Washington

February 2010

Friendship in a karate club



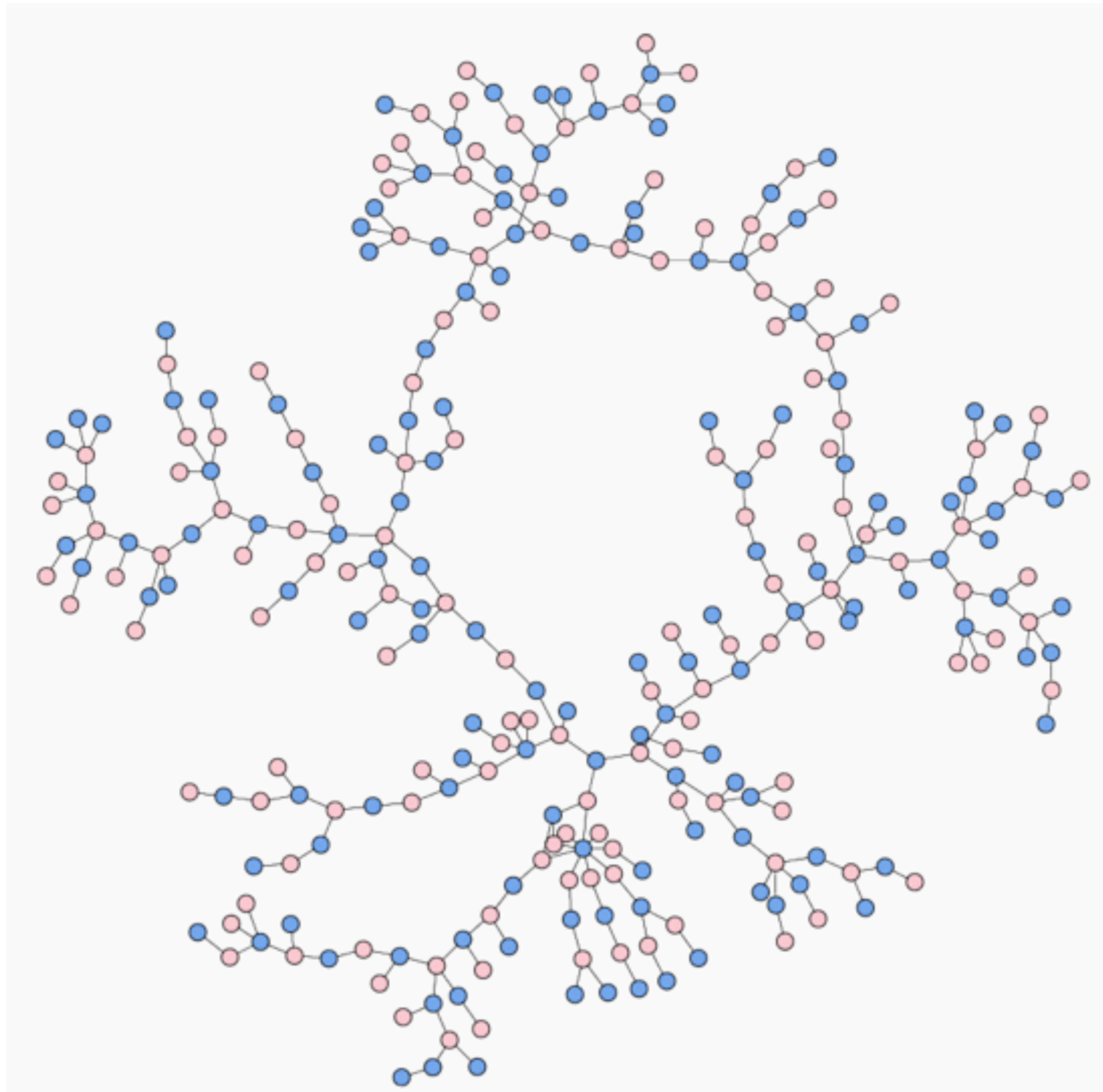
“Zachary’s Karate Club”

W. W. Zachary

An information flow model for conflict and fission in small groups

Journal of Anthropological Research, 1977

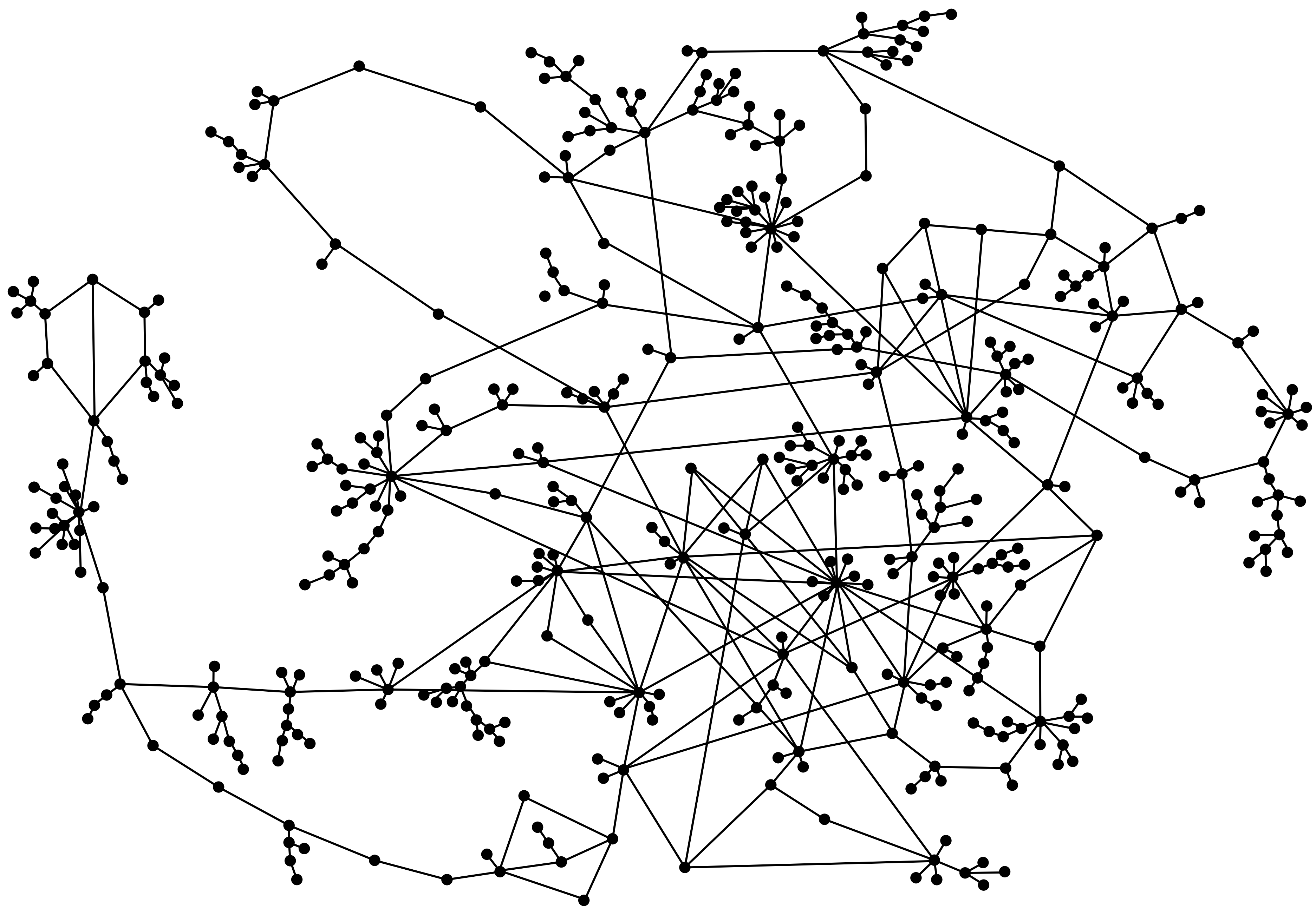
Romantic connections in a high school



Bearman, et al.
The structure of adolescent romantic and sexual networks.
American Journal of Sociology, 2004.

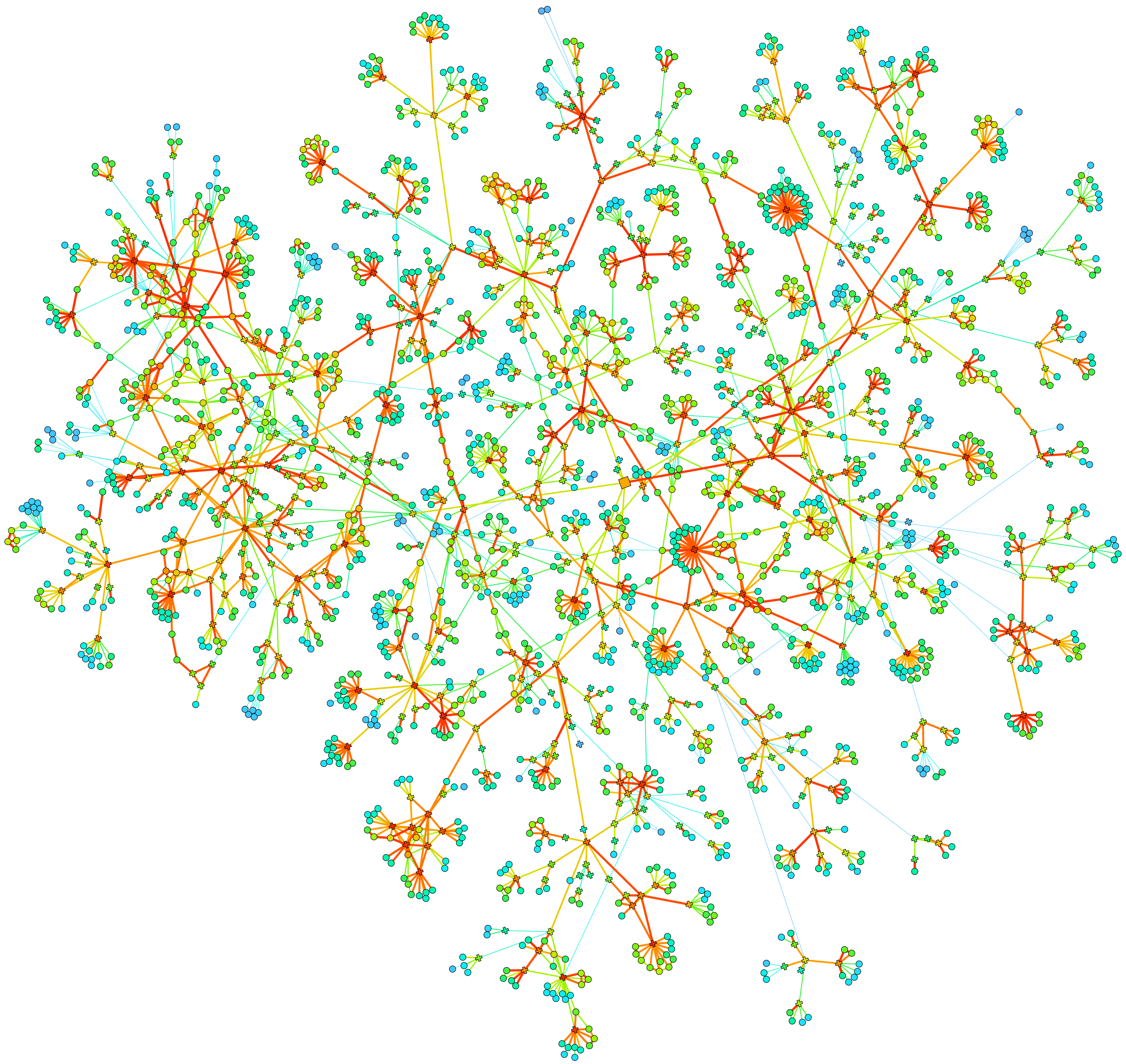
(Image drawn by Newman)

Sexual and injecting drug partners



Potterat, et al.
Risk network structure in the early epidemic phase of hiv transmission in colorado springs.
Sexually Transmitted Infectections, 2002.

Social ties derived from a mobile phone network



J. Onnela et al.
Structure and tie strengths in mobile communication networks,
Proceedings of the National Academy of Sciences, *2007*

Global instant messaging network

180 million nodes
1.3 billion edges

Leskovec, et al.
Planetary-scale views on a large instant-messaging network.
Conference on the World Wide Web, 2008.

Privacy risk a major obstacle to network analysis

Common outcomes include:

- No availability
- Limited availability:
  - Only within institutions who own the data, or among limited set of researchers who have negotiated access.
- Availability, at a cost:
  - Privacy of participants may be violated, bias or inaccuracy in released data.

Analysis of private networks

Can we permit analysts to study networks without revealing sensitive information about participants?

Example analyses based on network topology:

- **Properties of the degree distribution**
- **Motif analysis**
- Community structure
- Processes on networks: routing, rumors, infection
- Resiliency / robustness

Outline of the talk

1. Existing approaches to protecting network data

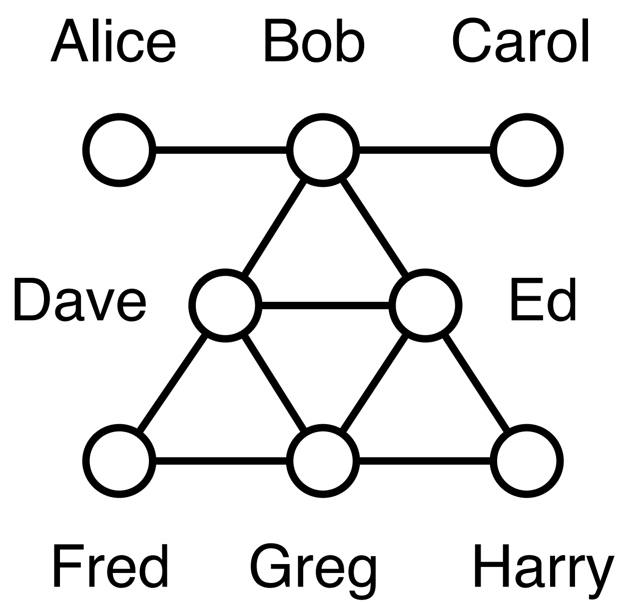
2. Background on differential privacy

3. Privately estimating the degree distribution

4. Privately counting motifs

5. Future goals and open questions

Sensitive information in networks



Nodes

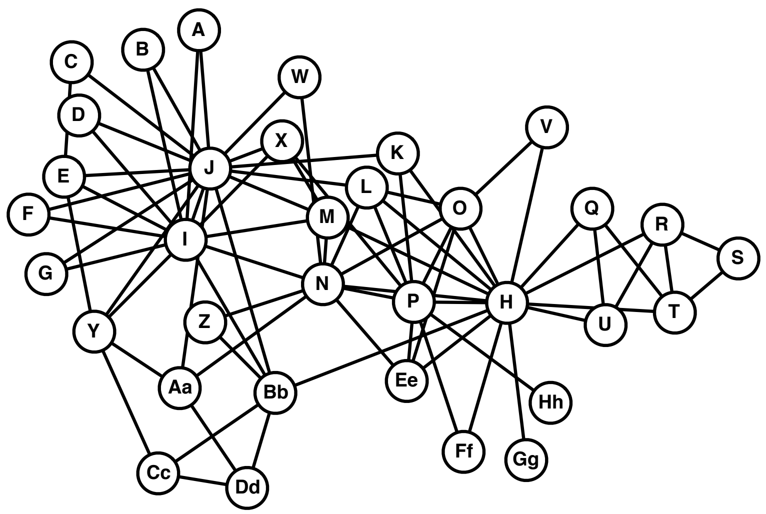
ID	Age	HIV
Alice	25	Pos
Bob	19	Neg
Carol	34	Pos
Dave	45	Pos
Ed	32	Neg
Fred	28	Neg
Greg	54	Pos
Harry	49	Neg

Edges

ID1	ID2
Alice	Bob
Bob	Carol
Bob	Dave
Bob	Ed
Dave	Ed
Dave	Fred
Dave	Greg
Ed	Greg
Ed	Harry
Fred	Greg
Greg	Harry

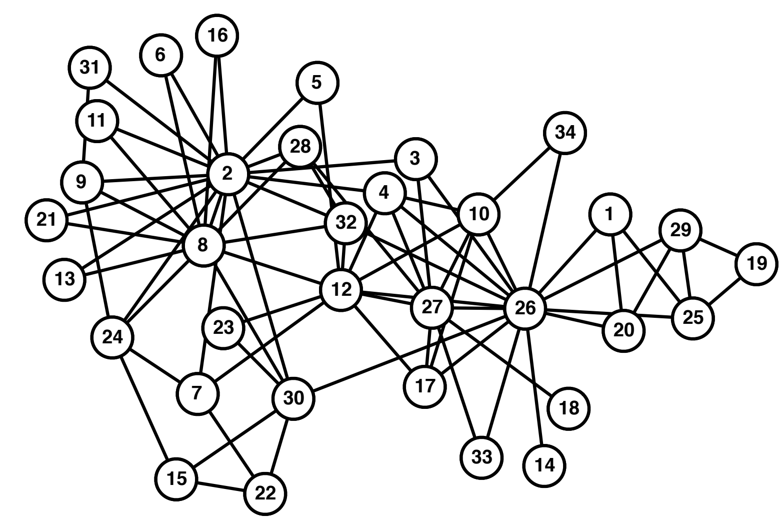
Naive anonymization

DATA OWNER | **ANALYST**



Original network

Naive Anonymization

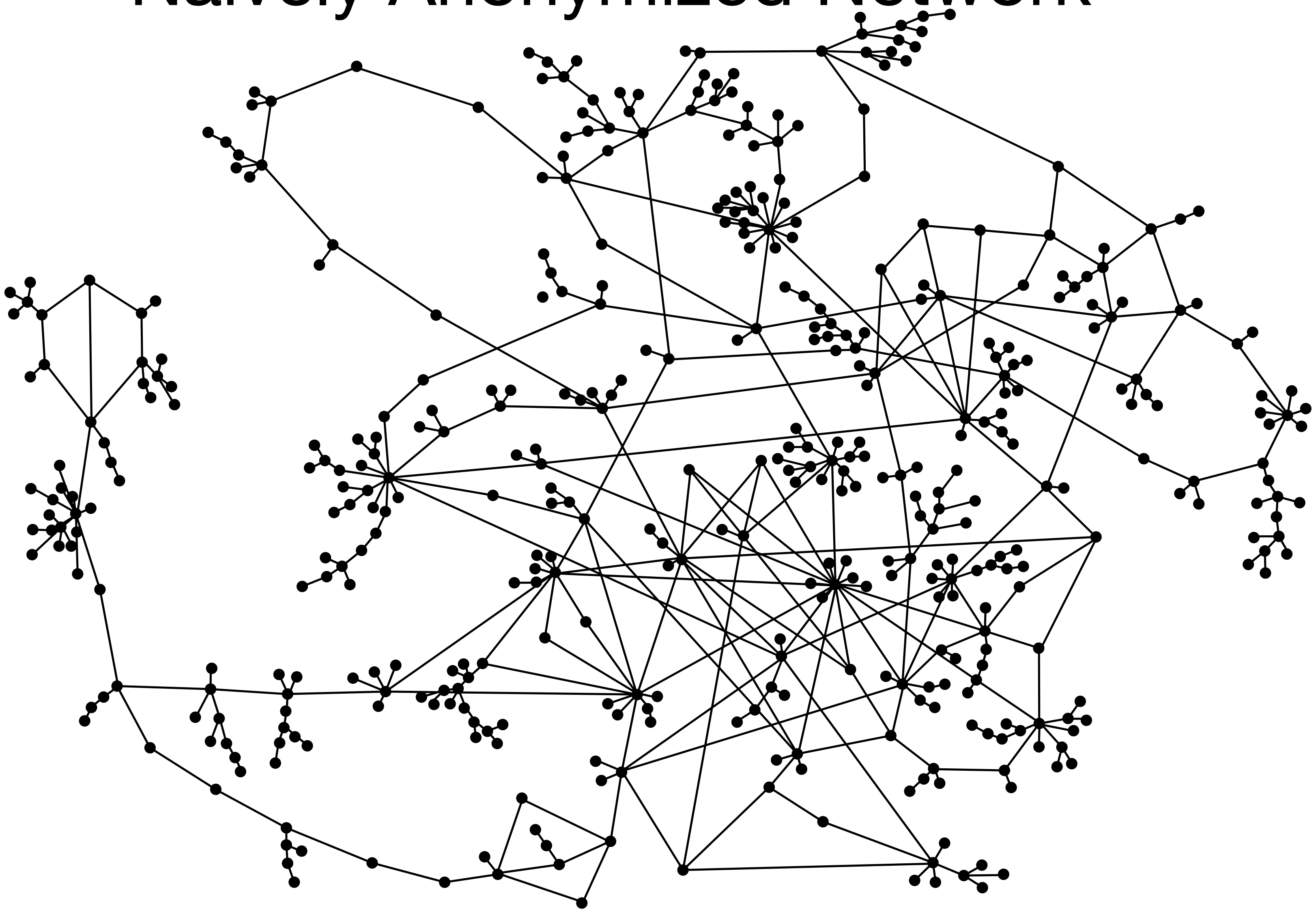


Naive anonymization

- Naive anonymization replaces identifiers with random numbers, releasing an isomorphic copy of the graph.
- Allows very accurate analysis of the topology... but not secure.

Threat of re-identification

Naively Anonymized Network

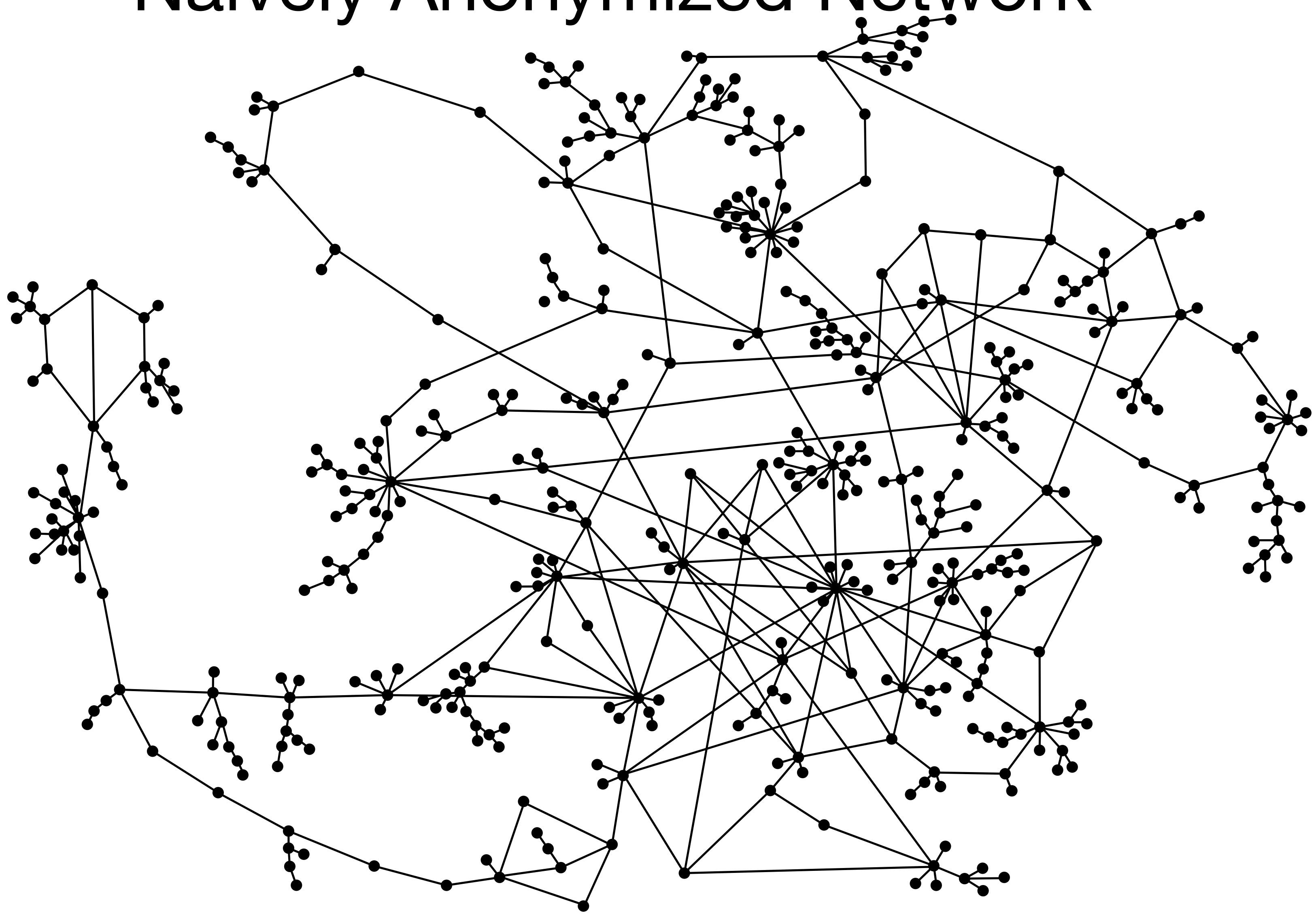


Re-identification

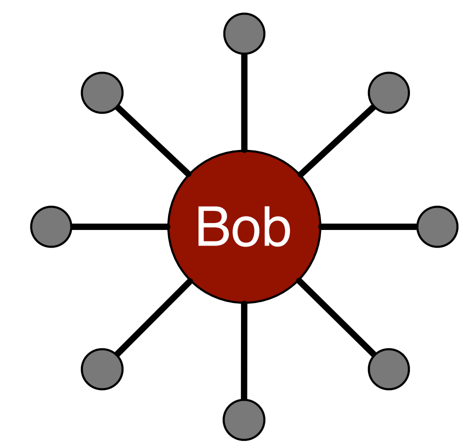
Adversary acquires knowledge of network structure and uses it to re-identify individual

Threat of re-identification

Naively Anonymized Network



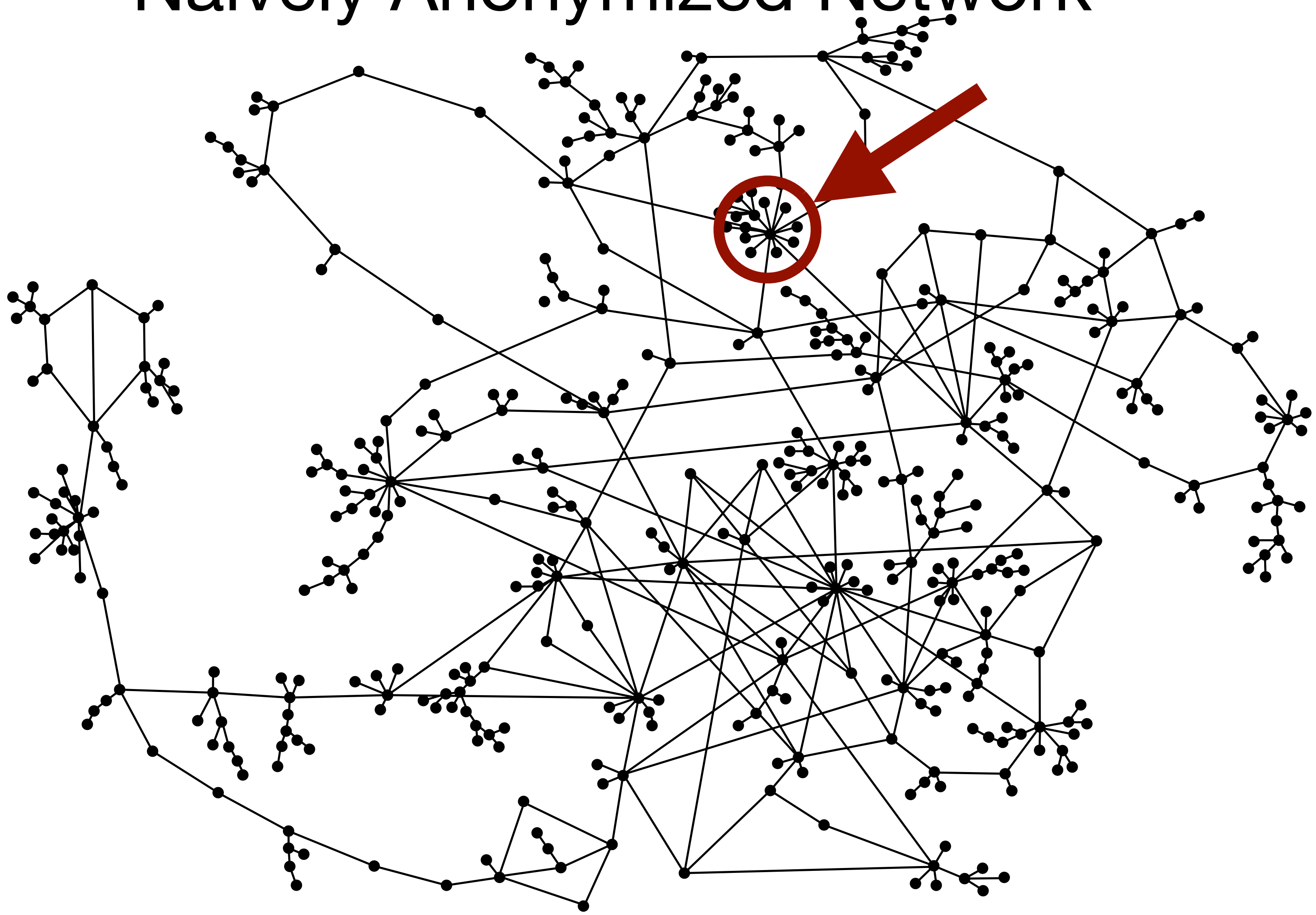
External information



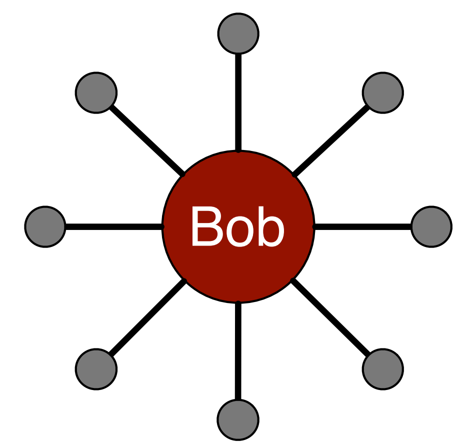
Re-identification Adversary acquires knowledge of network structure and uses it to re-identify individual

Threat of re-identification

Naively Anonymized Network



External information

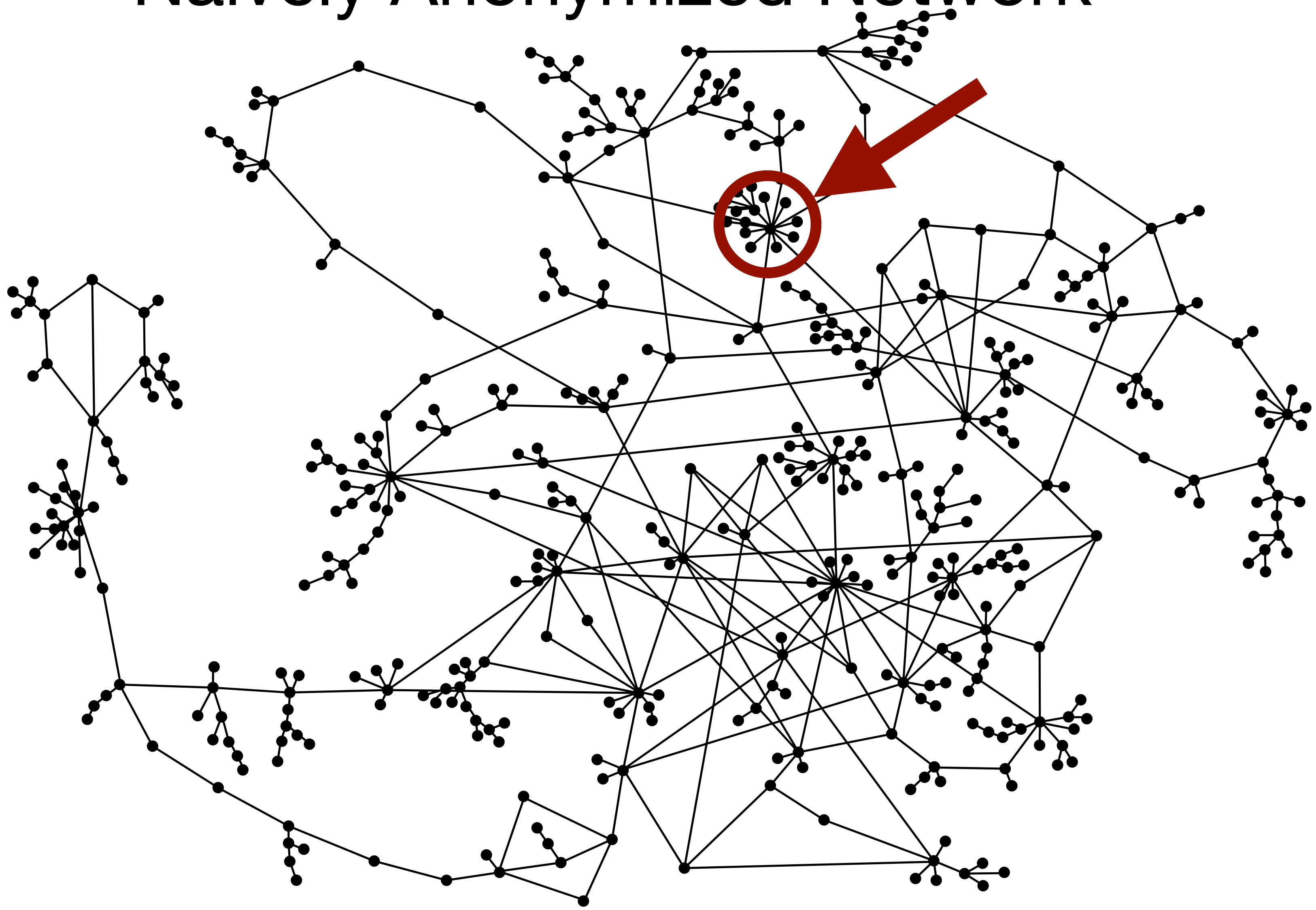


Re-identification

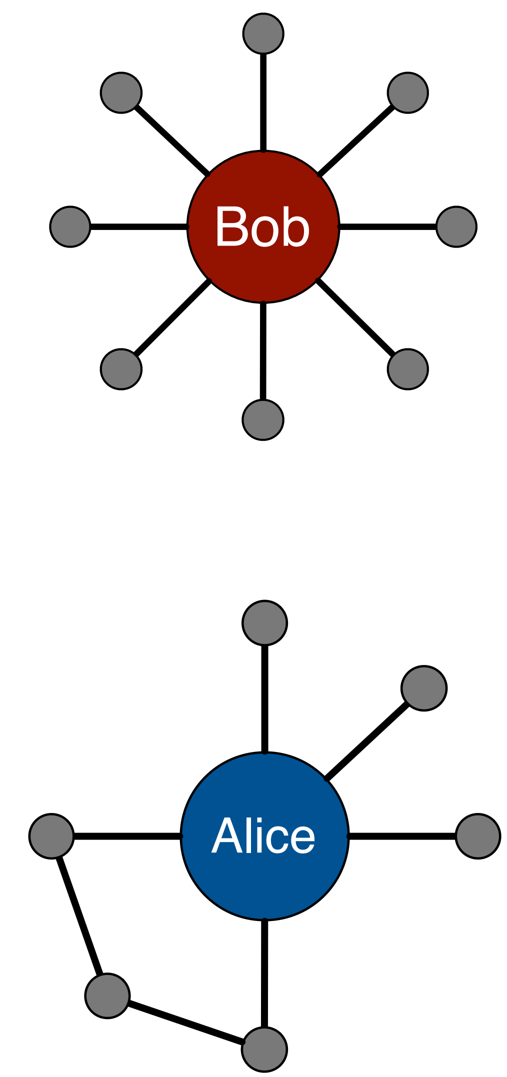
Adversary acquires knowledge of network structure and uses it to re-identify individual

Threat of re-identification

Naively Anonymized Network



External information

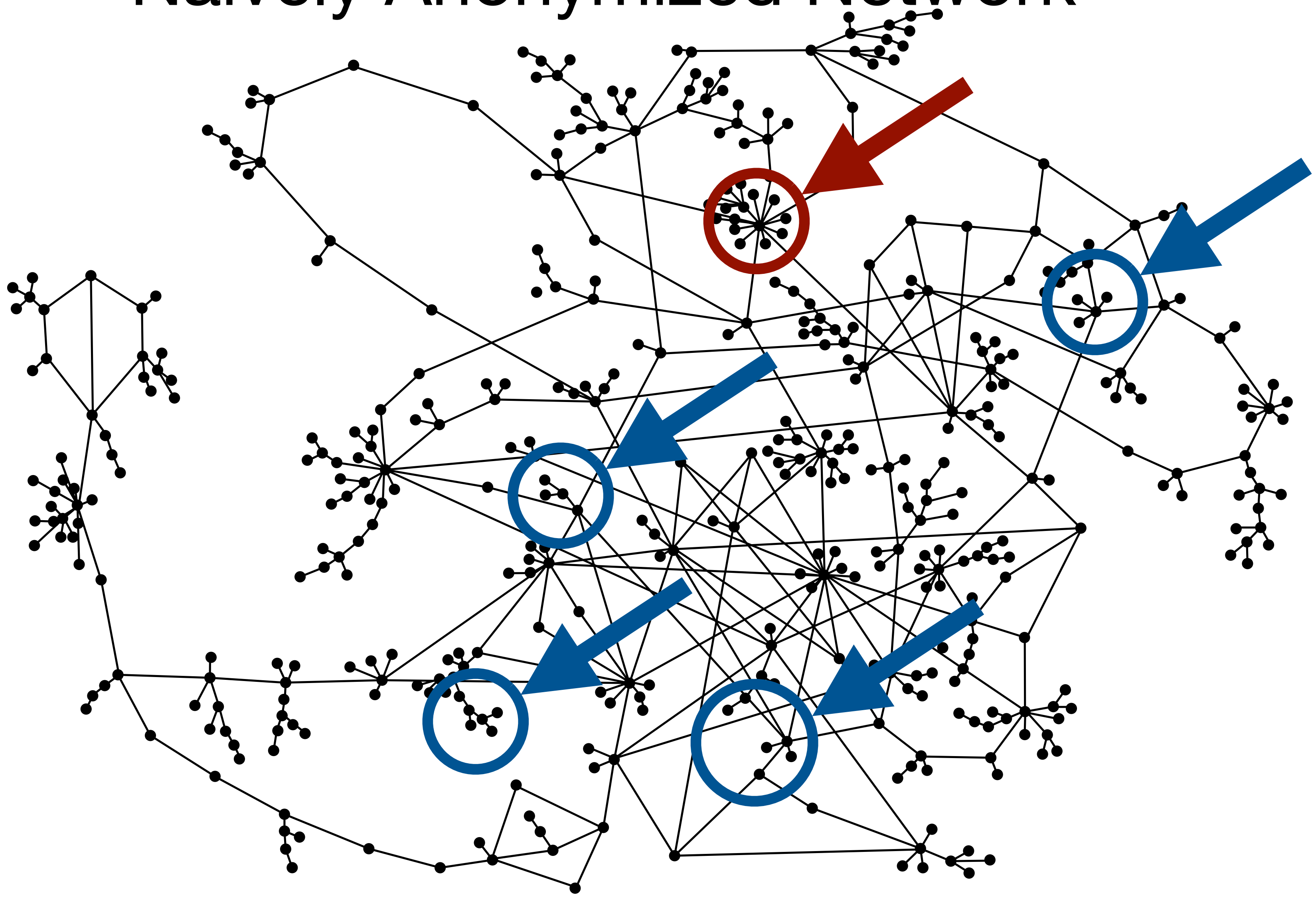


Re-identification

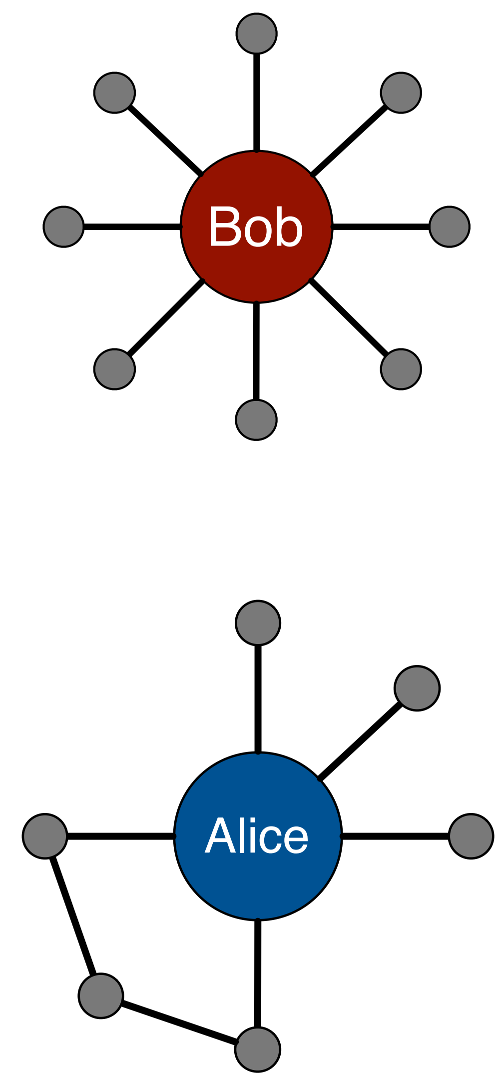
Adversary acquires knowledge of network structure and uses it to re-identify individual

Threat of re-identification

Naively Anonymized Network



External information

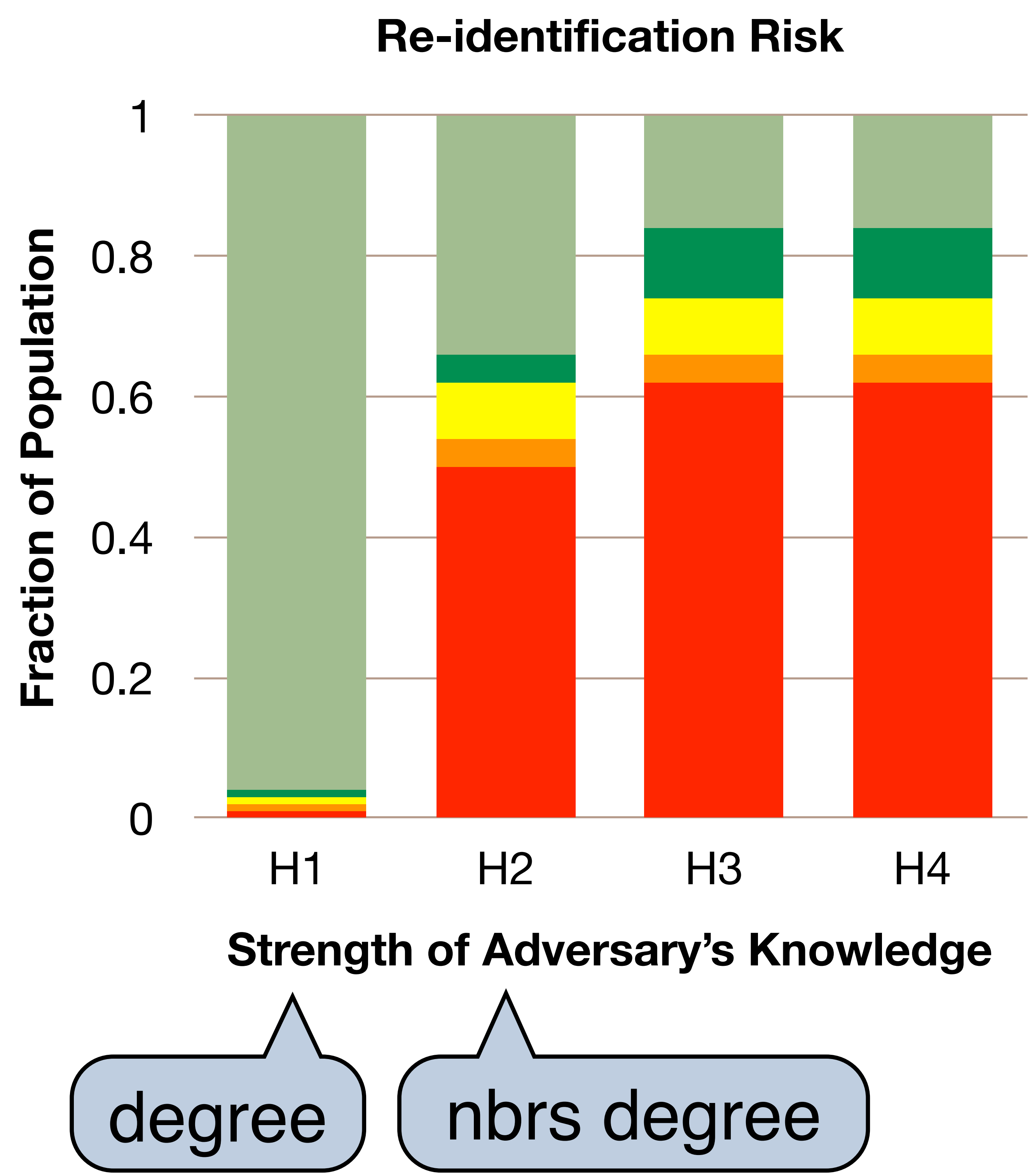


Re-identification

Adversary acquires knowledge of network structure and uses it to re-identify individual

Local structure is highly identifying

Friendster network
~4.5 million nodes



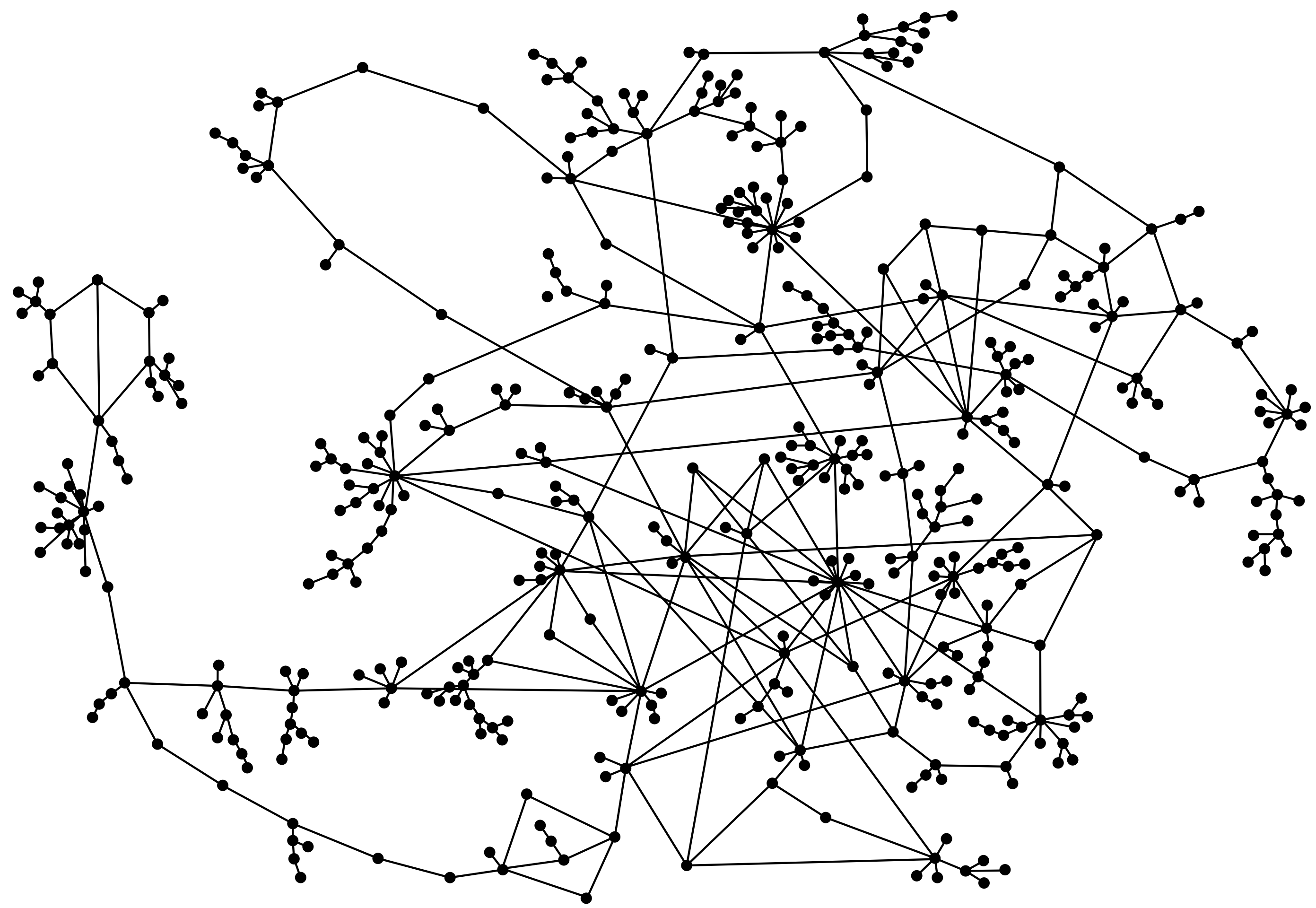
[Hay, VLDB 08]

Other attacks on naive anonymization

Active attack

Embed small random graph prior to anonymization.

[Backstrom, WWW 07]



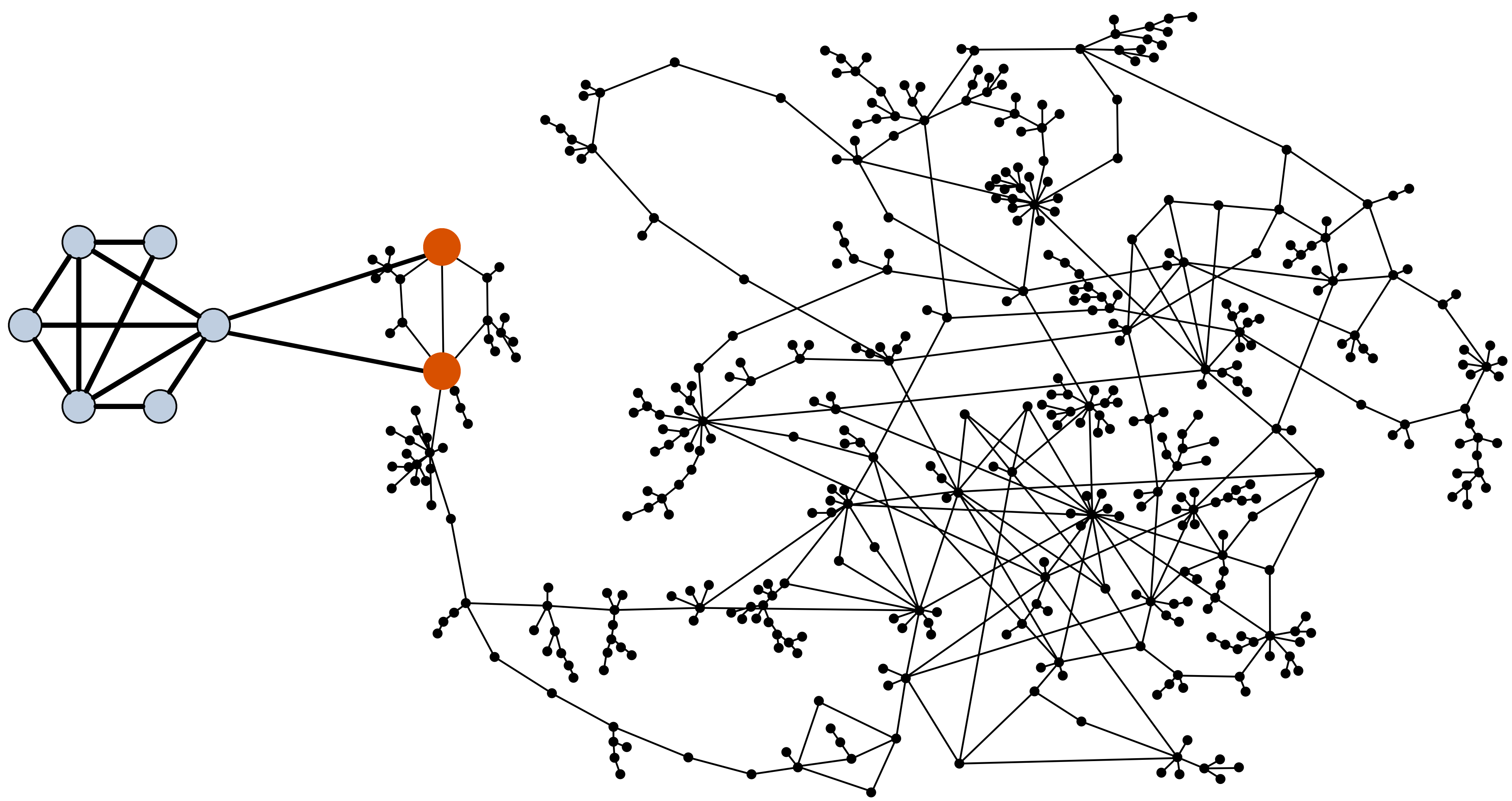
Auxiliary network attack

Use unanonymized public network with overlapping membership.

[Narayanan, OAKL 09]

Other attacks on naive anonymization

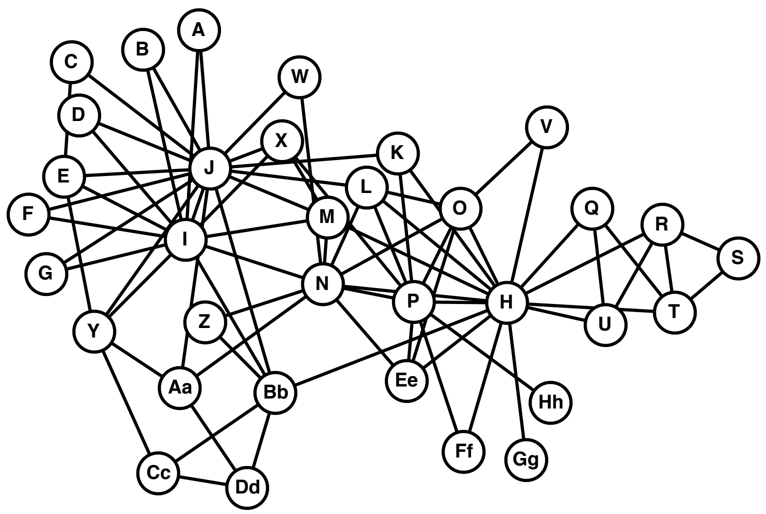
Active attack Embed small random graph prior to anonymization. **[Backstrom, WWW 07]**



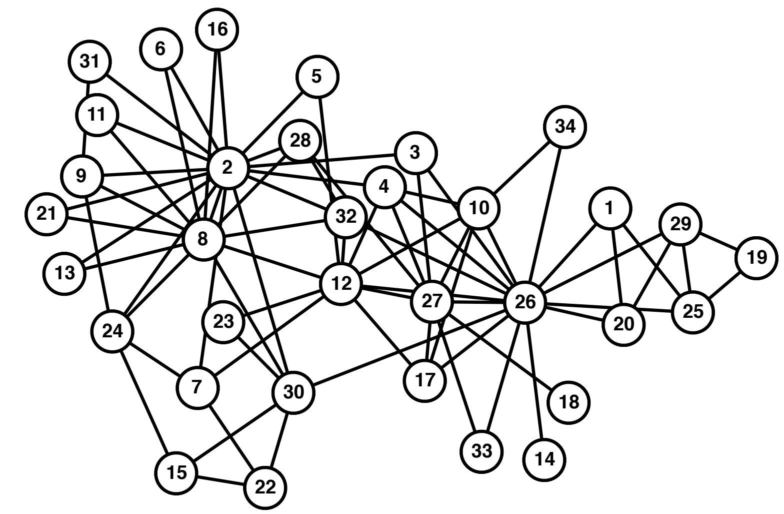
Auxiliary network attack Use unanonymized public network with overlapping membership. **[Narayanan, OAKL 09]**

Improved data publishing techniques

DATA OWNER | **ANALYST**



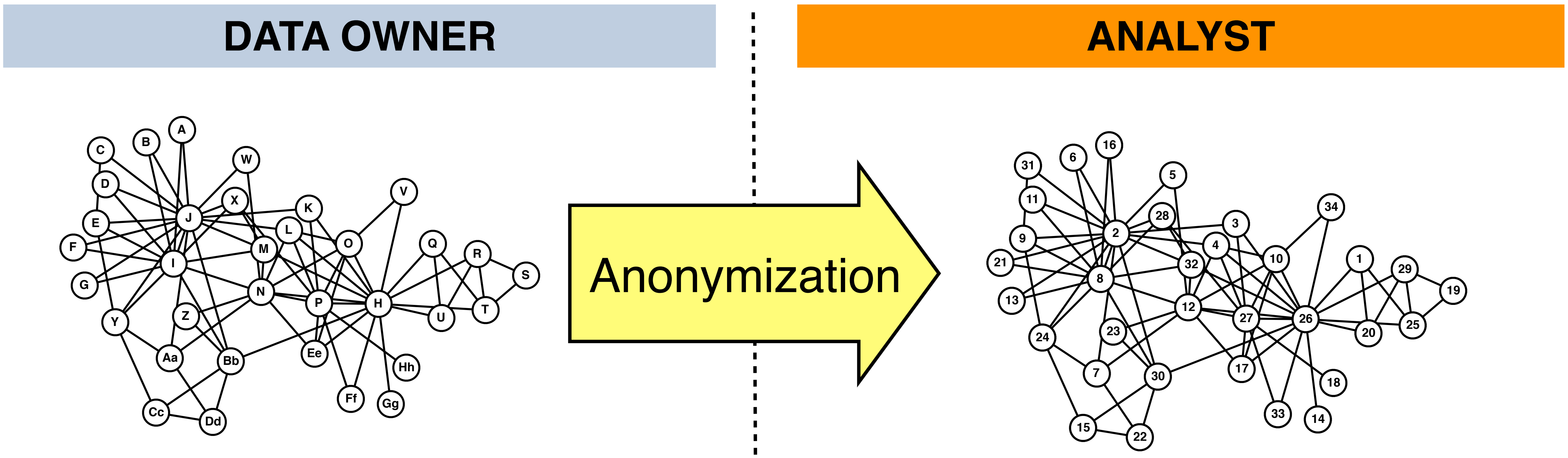
Anonymization



Original network

Naive anonymization

Improved data publishing techniques



Original network

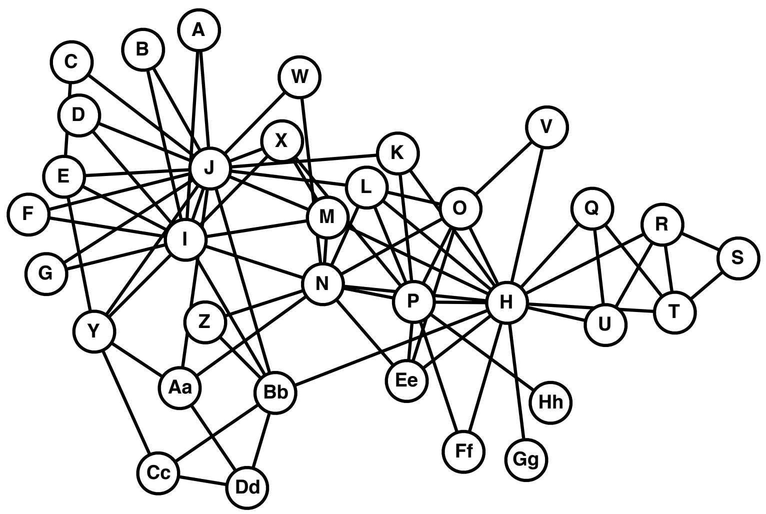
Naive anonymization

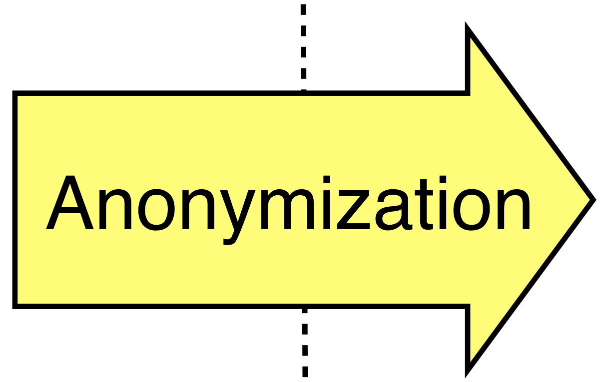
- Create topological similarity **[Liu, SIGMOD 08] [Zhou, ICDE 08] [Zou, VLDB 09]**

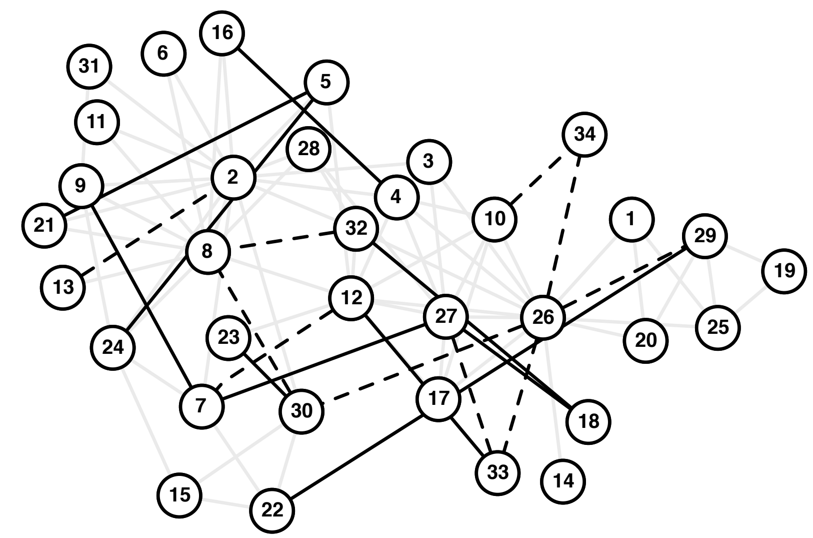
Improved data publishing techniques

DATA OWNER

ANALYST







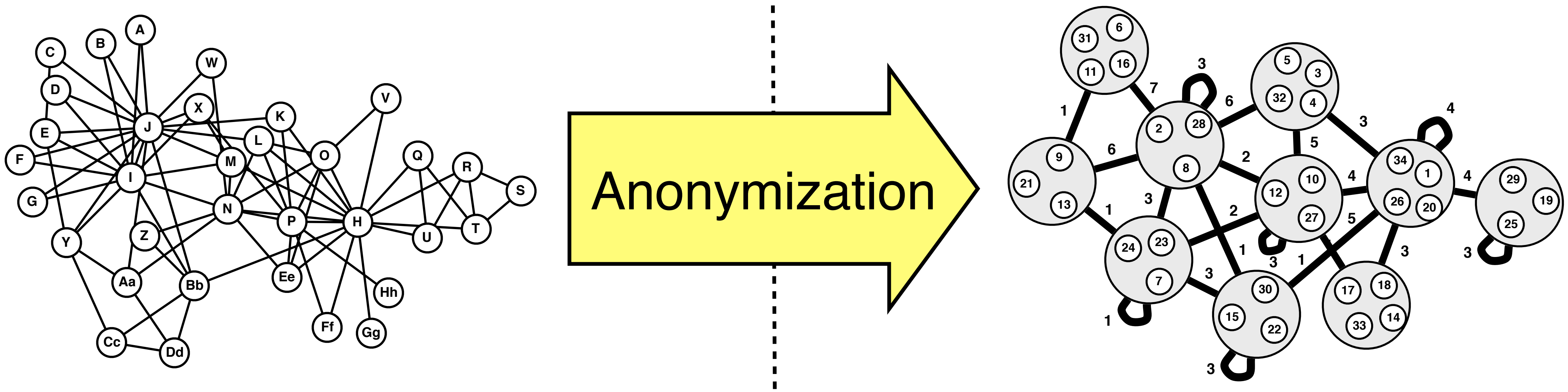
Original network

Randomized Edges

- Create topological similarity **[Liu, SIGMOD 08] [Zhou, ICDE 08] [Zou, VLDB 09]**
- Randomize edges **[Ying, SDM 2008]**

Improved data publishing techniques

DATA OWNER | **ANALYST**

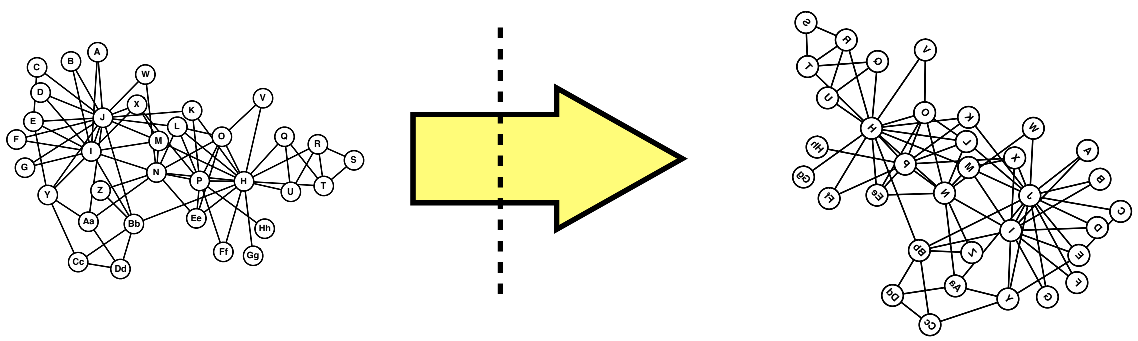


Original network | Node clustering

- Create topological similarity **[Liu, SIGMOD 08] [Zhou, ICDE 08] [Zou, VLDB 09]**
- Randomize edges **[Ying, SDM 2008]**
- Clustering/summarization **[Campan, PinKDD 08] [Hay, VLDB 08] [Cormode, VLDB 08] [Cormode, VLDB 09]**

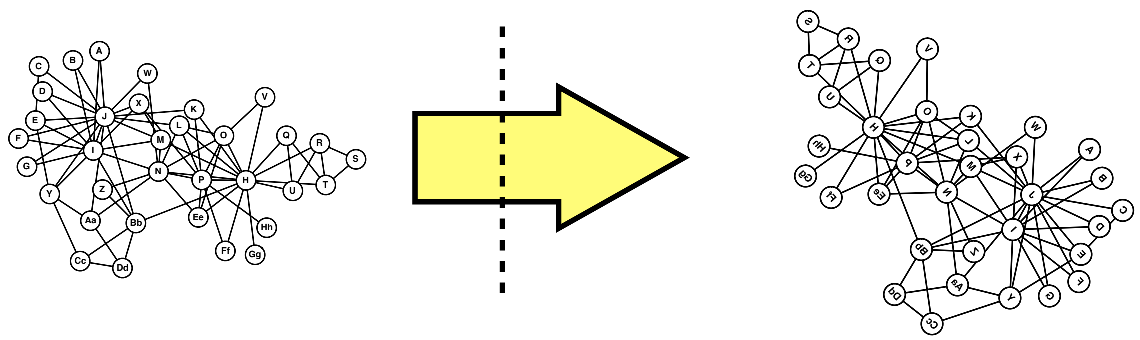
Data publishing v. output perturbation

- Data publishing



Data publishing v. output perturbation

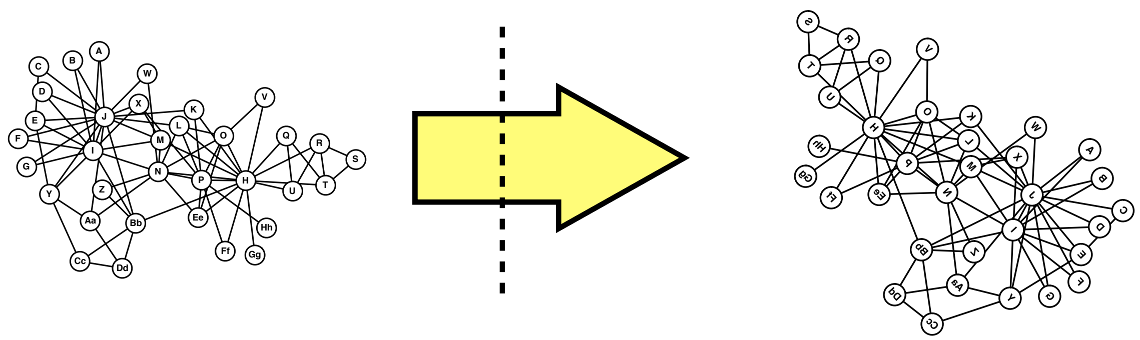
- Data publishing



Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

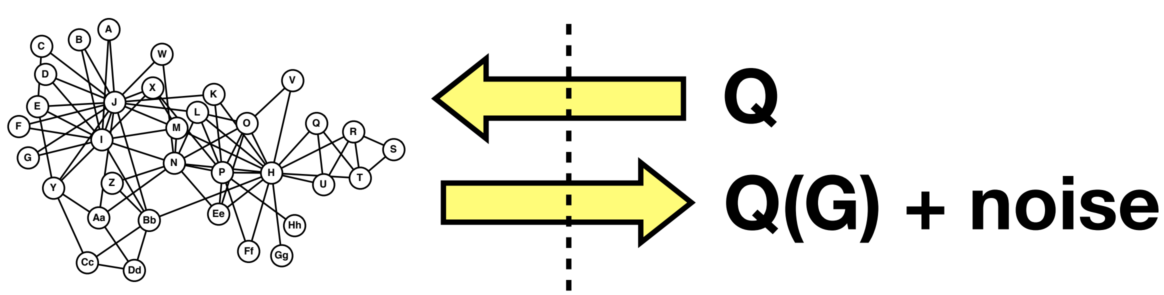
Data publishing v. output perturbation

- Data publishing



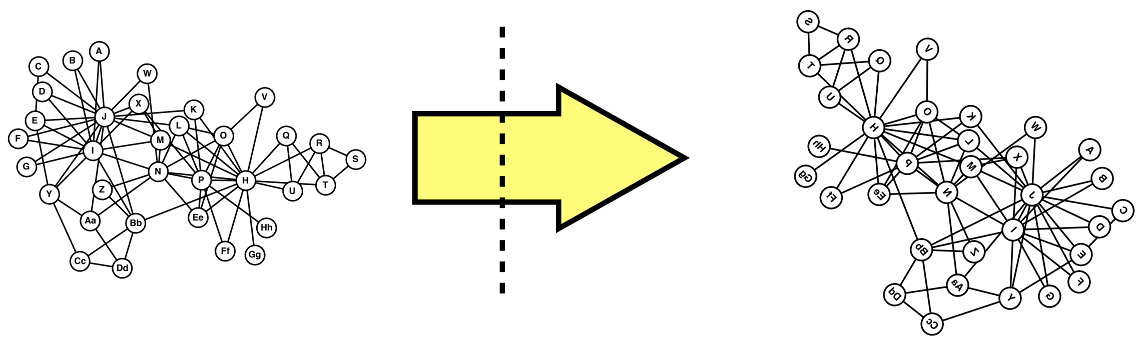
Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

- Output perturbation



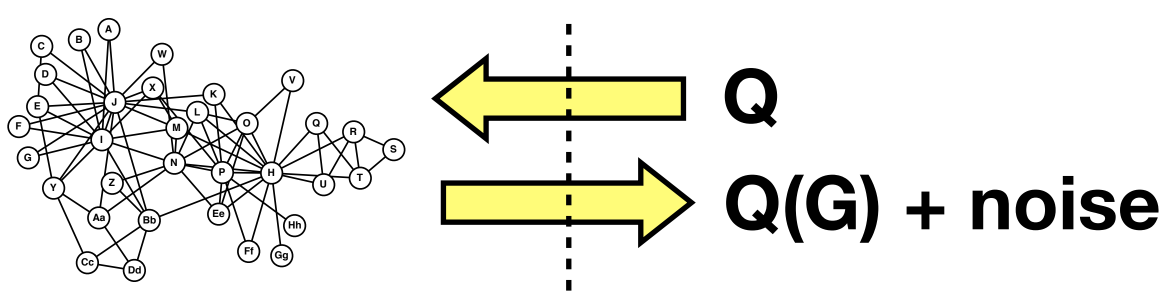
Data publishing v. output perturbation

- Data publishing



Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

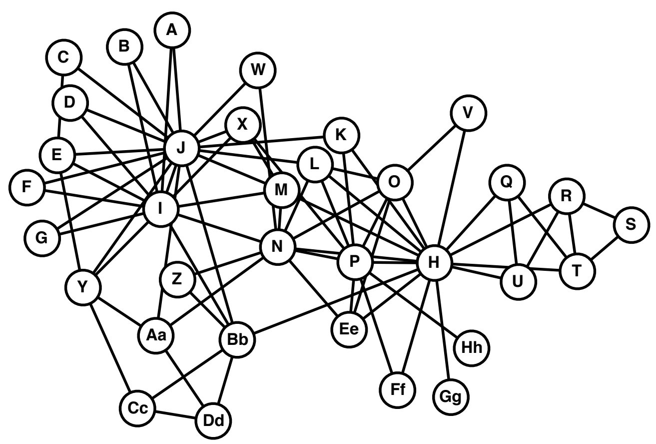
- Output perturbation



Ease of use	bad for practical analyses
Privacy	formal guarantees
Accuracy	provable bounds
Scalability	very good

Output perturbation

DATA OWNER | **ANALYST**



Original network

query

Q

Q(G) + random noise

noisy result

- Dwork, McSherry, Nissim, Smith **[Dwork, TCC 06]** have described an output perturbation mechanism satisfying ***differential privacy.***

- Comparatively few results for graph data.

Outline

1. Existing approaches to protecting network data

2. Background on differential privacy

3. Privately estimating the degree distribution

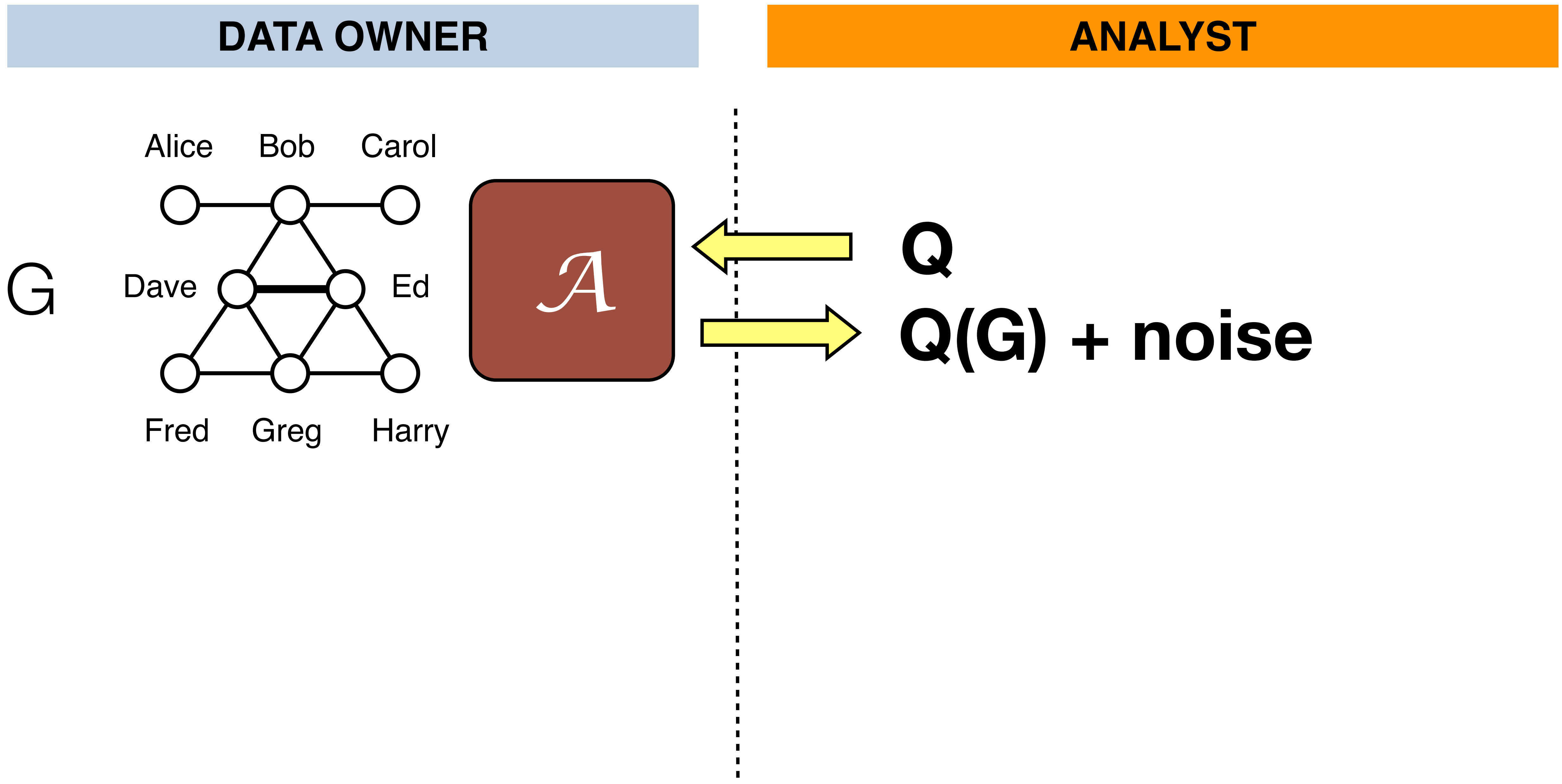
4. Privately counting motifs

5. Future goals and open questions

Outline

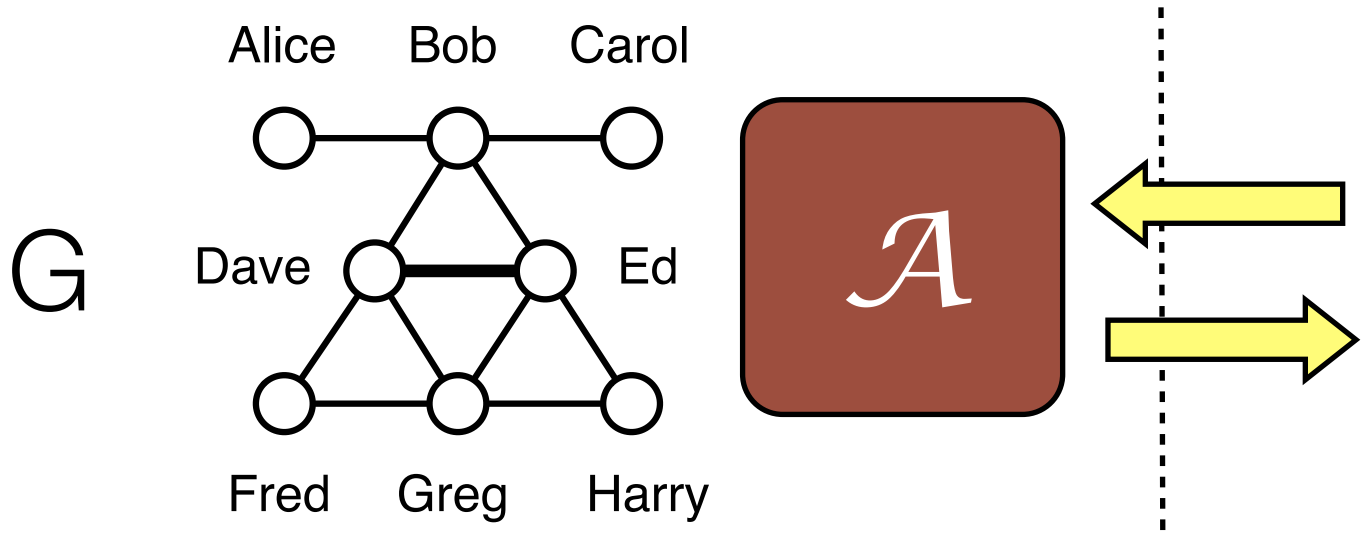
1. Existing approaches to protecting network data
2. Background on differential privacy
3. Privately estimating the degree distribution
4. Privately counting motifs
5. Future goals and open questions

The differential guarantee



The differential guarantee

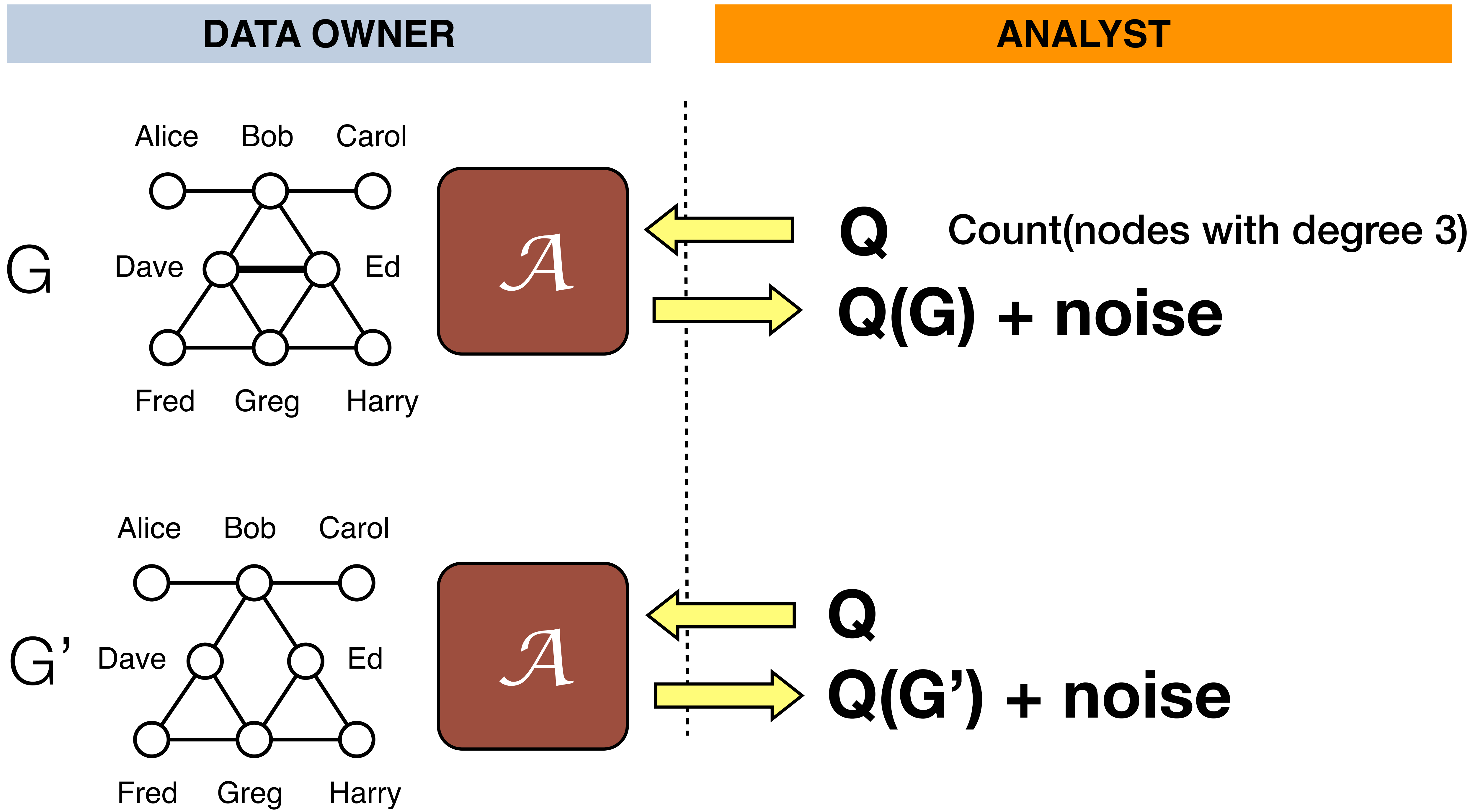
DATA OWNER	ANALYST



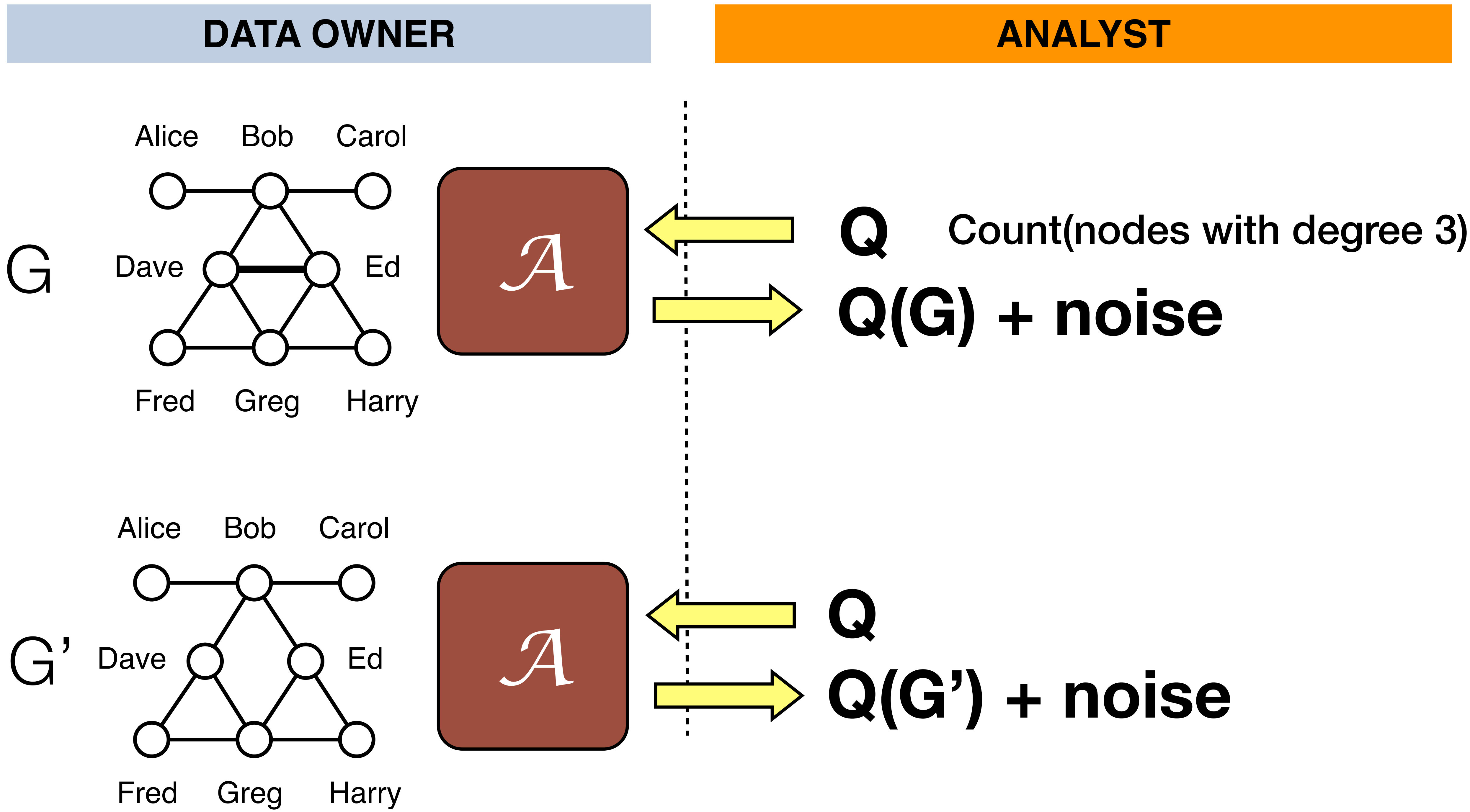
Q Count(nodes with degree 3)

Q(G) + noise

The differential guarantee



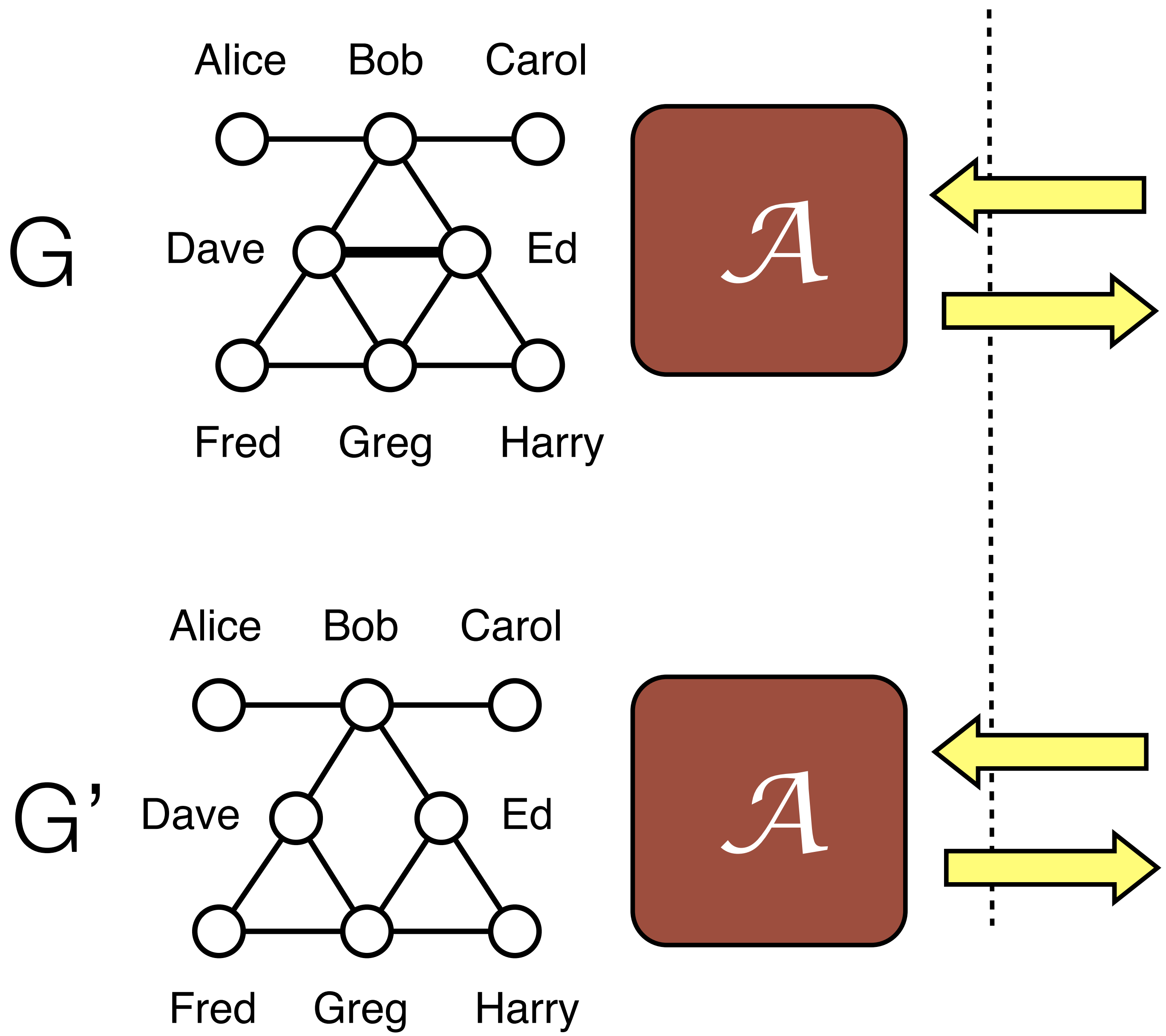
The differential guarantee



Two graphs are **neighbors** if they differ by at most one edge

The differential guarantee

DATA OWNER | **ANALYST**



Q **Count(nodes with degree 3)**

Q(G) + noise

0 + noise = p

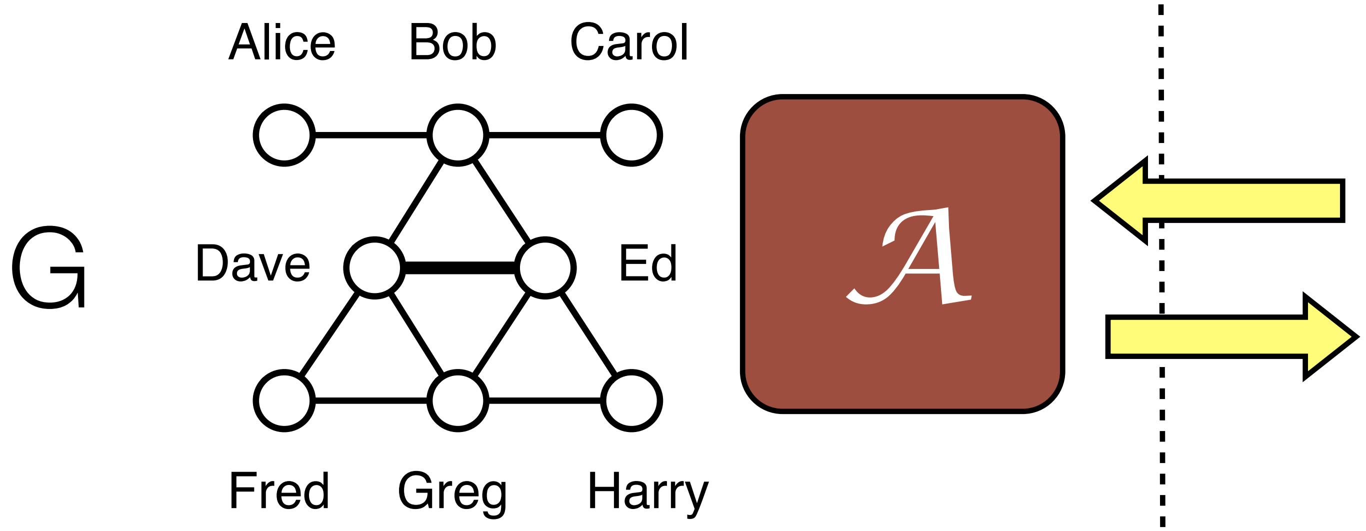
Q

Q(G') + noise

Two graphs are **neighbors** if they differ by at most one edge

The differential guarantee

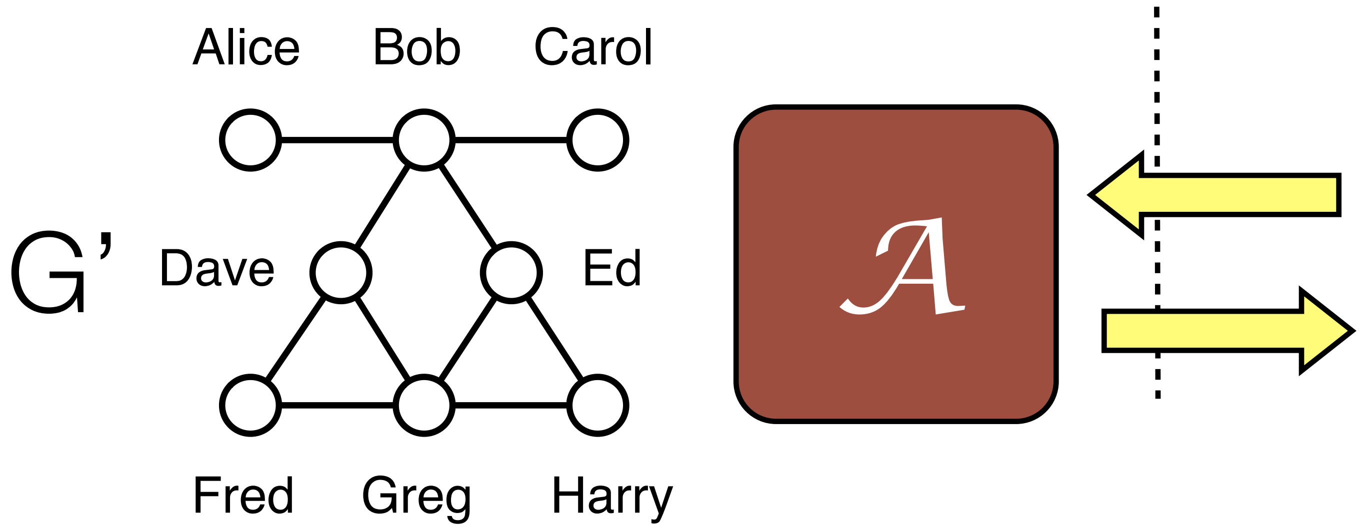
DATA OWNER	ANALYST



Q **Count(nodes with degree 3)**

Q(G) + noise

0 + noise = p



Q

Q(G') + noise

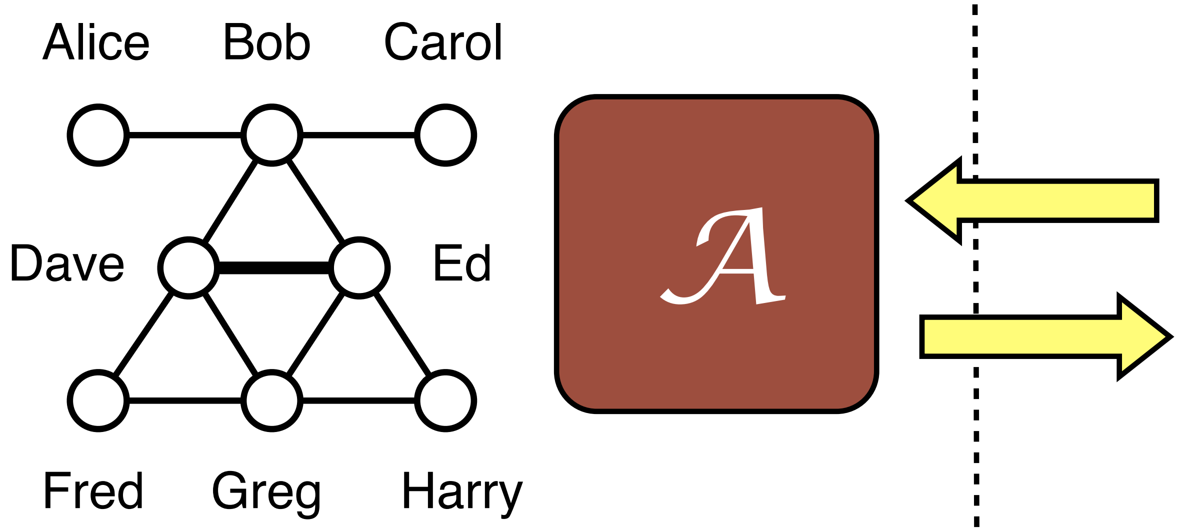
2 + noise = q

Two graphs are **neighbors** if they differ by at most one edge

The differential guarantee

DATA OWNER | **ANALYST**

G



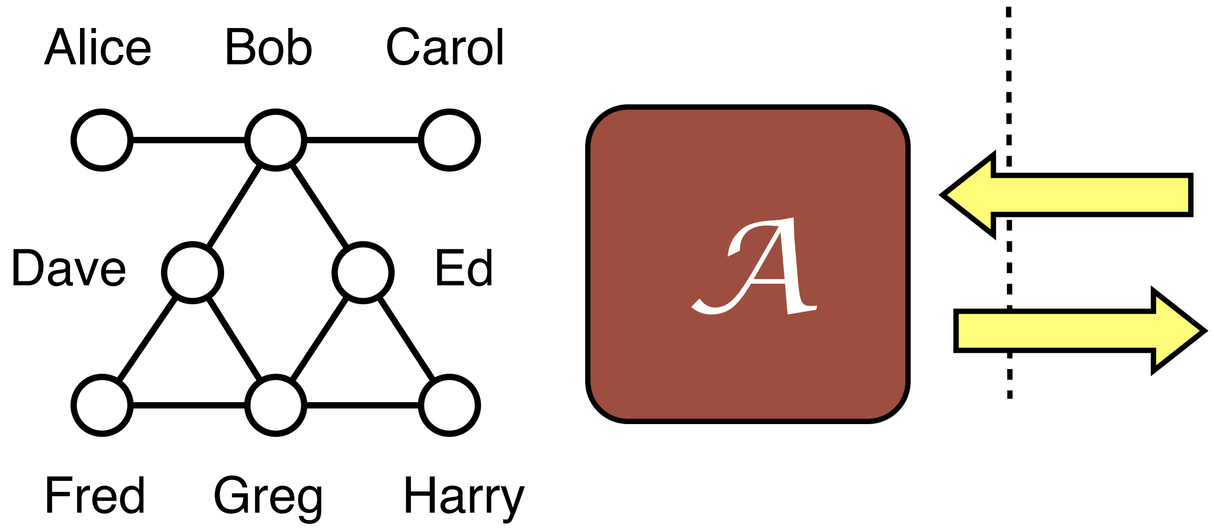
Q **Count(nodes with degree 3)**

Q(G) + noise

0 + noise = p

Indistinguishable outputs

G'



Q

Q(G') + noise

2 + noise = q

Two graphs are **neighbors** if they differ by at most one edge

Differential privacy

A randomized algorithm A provides **Ɛ-differential privacy** if:

for all neighboring graphs G and G', and

for any set of outputs *S*:

$$Pr[\mathcal{A}(G) \in S] \quad \leq \quad e^{\epsilon} Pr[\mathcal{A}(G') \in S]$$

[Dwork, TCC 06]

Differential privacy

A randomized algorithm A provides **Ɛ-differential privacy** if:

for all neighboring graphs G and G', and

for any set of outputs *S*:

$$Pr[\mathcal{A}(G) \in S] \quad \leq \quad e^{\epsilon} Pr[\mathcal{A}(G') \in S]$$

epsilon is a privacy parameter

Differential privacy

A randomized algorithm A provides **ε-differential privacy** if:
for all neighboring graphs G and G', and
for any set of outputs *S*:

$$Pr[\mathcal{A}(G) \in S] \quad \leq \quad e^{\epsilon} Pr[\mathcal{A}(G') \in S]$$

epsilon is a privacy parameter

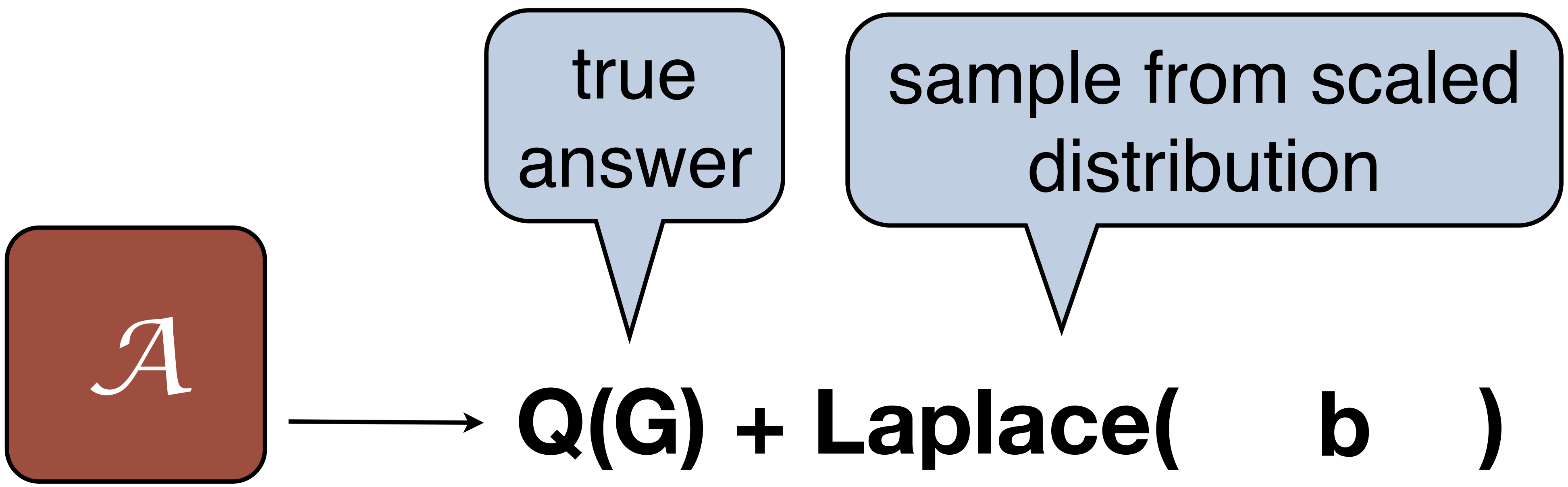
Epsilon is usually small: e.g. if $\epsilon = 0.1$ then $e^{\epsilon} \approx 1.10$

↓ epsilon = ↑ stronger privacy

[Dwork, TCC 06]

Calibrating noise

- The following algorithm for answering Q is ε-differentially private:



[Dwork, TCC 06]

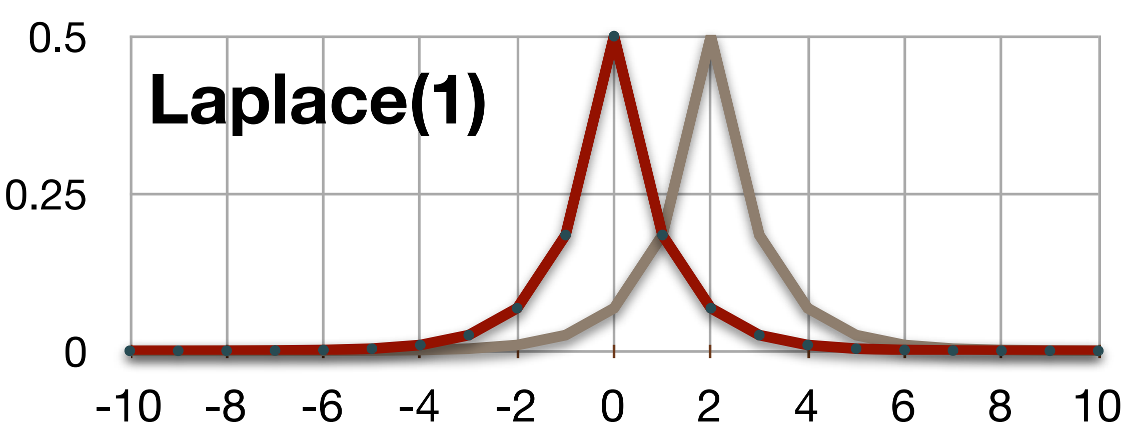
Calibrating noise

- The following algorithm for answering Q is ε-differentially private:

true answer

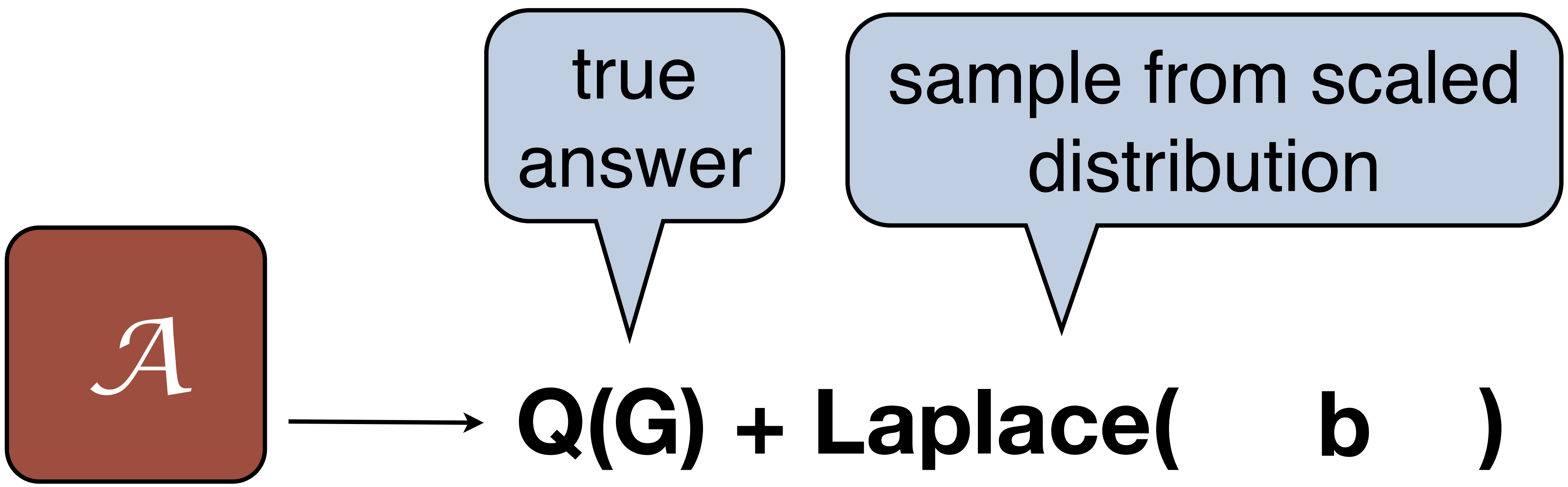
sample from scaled distribution

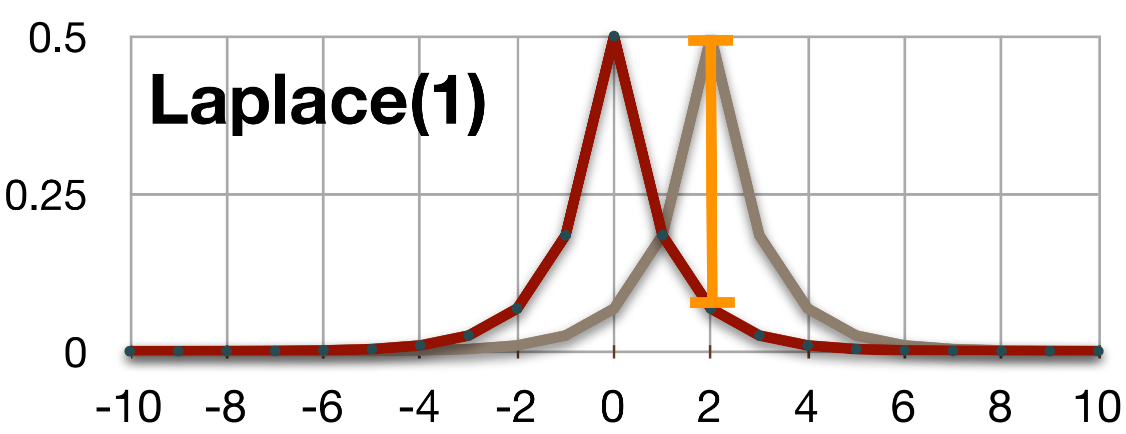
$\mathcal{A}$ → **Q(G) + Laplace(b)**



Calibrating noise

- The following algorithm for answering Q is ε-differentially private:

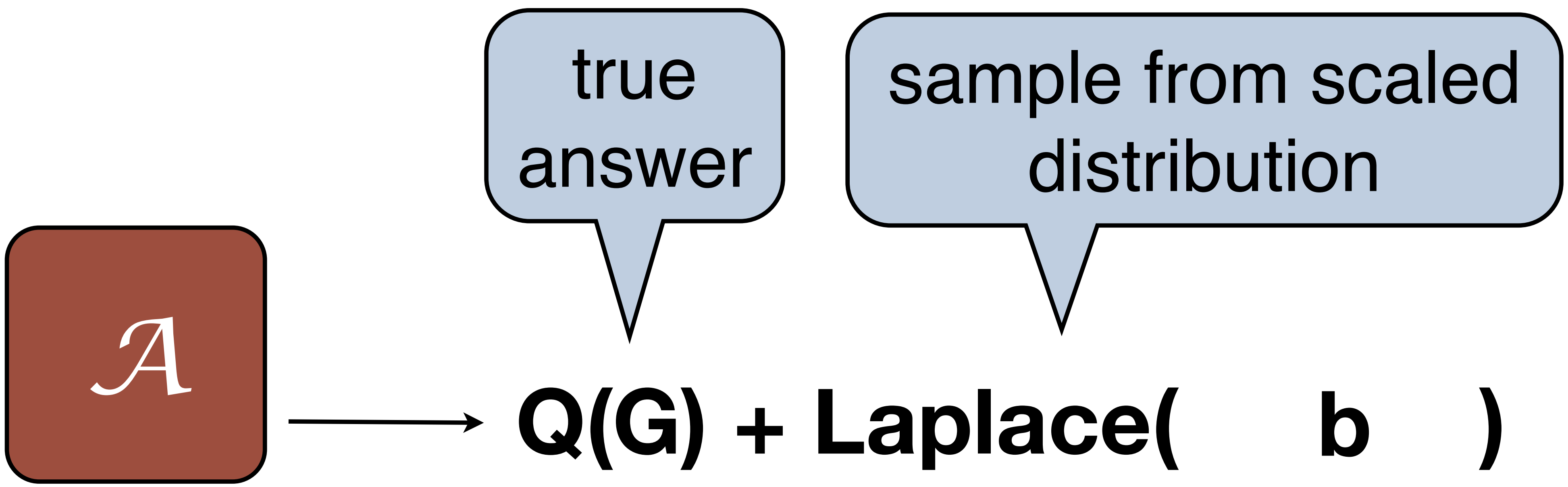


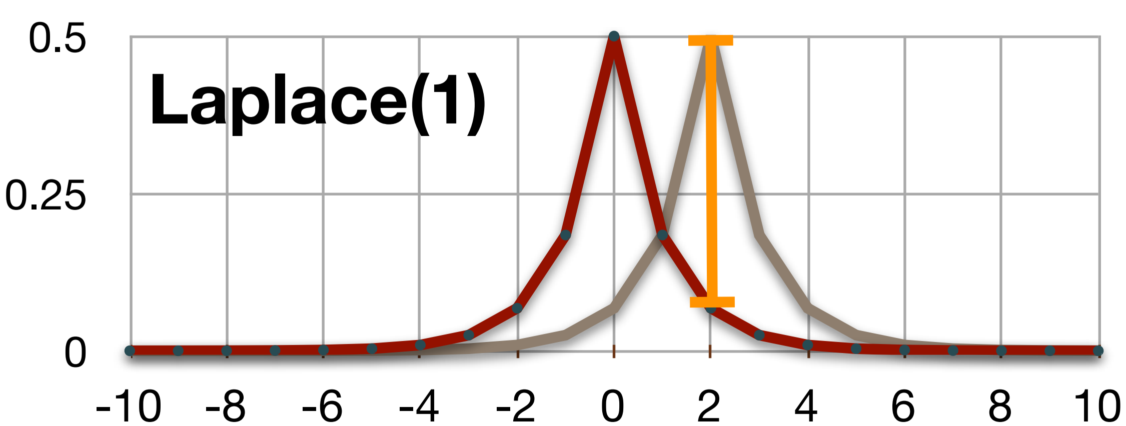


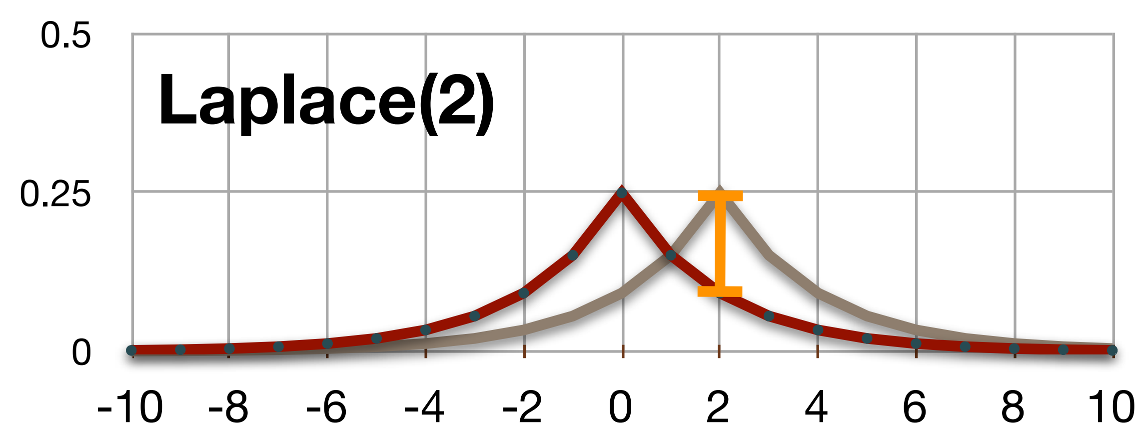
[Dwork, TCC 06]

Calibrating noise

- The following algorithm for answering Q is ε-differentially private:



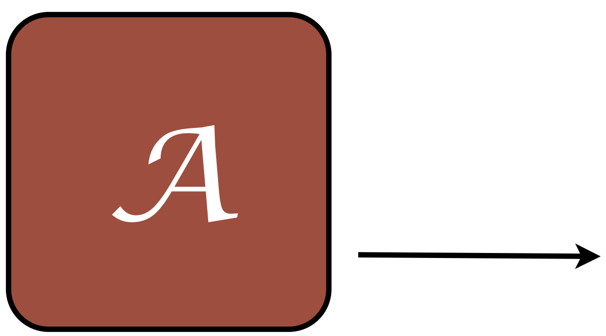




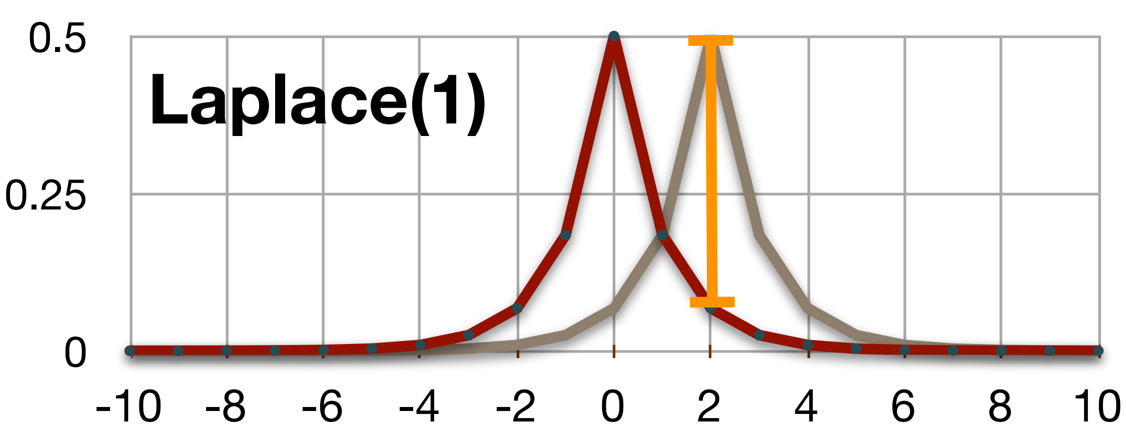
[Dwork, TCC 06]

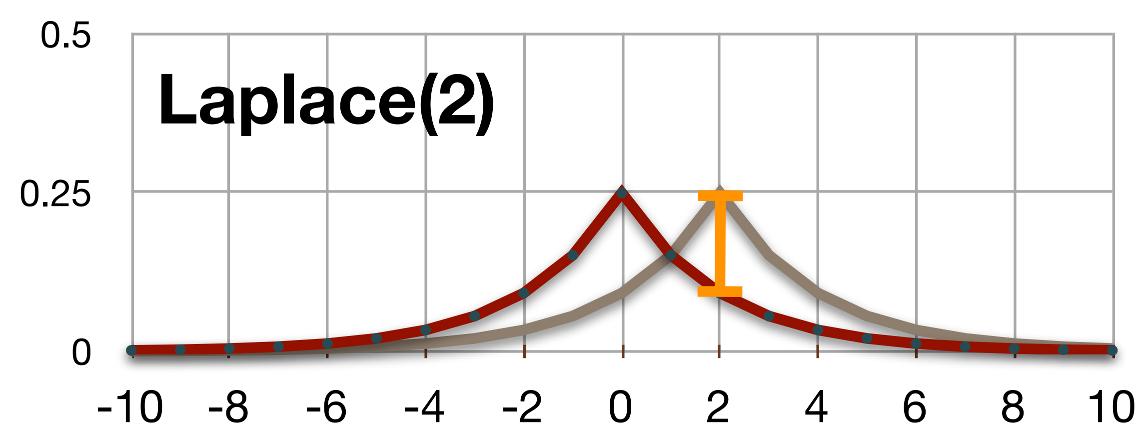
Calibrating noise

- The following algorithm for answering Q is ε-differentially private:



Q(G) + Laplace(   )





[Dwork, TCC 06]

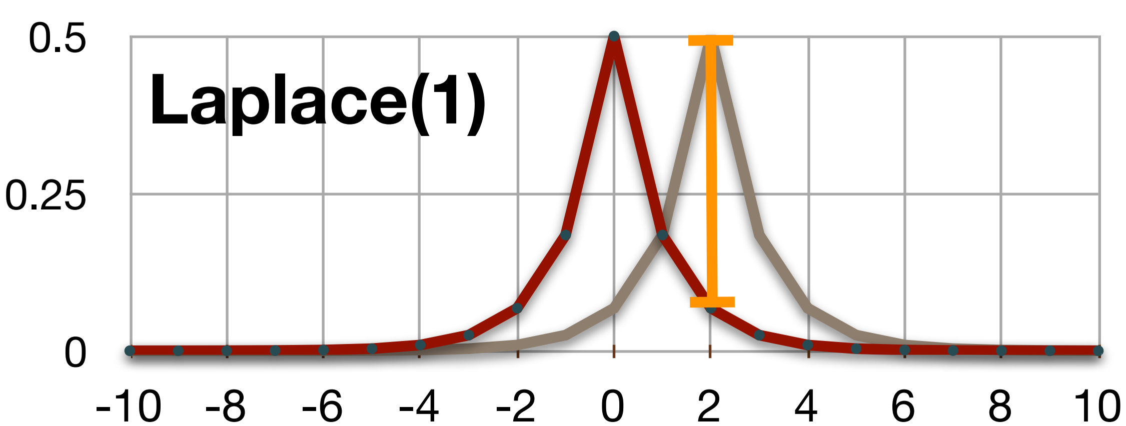
Calibrating noise

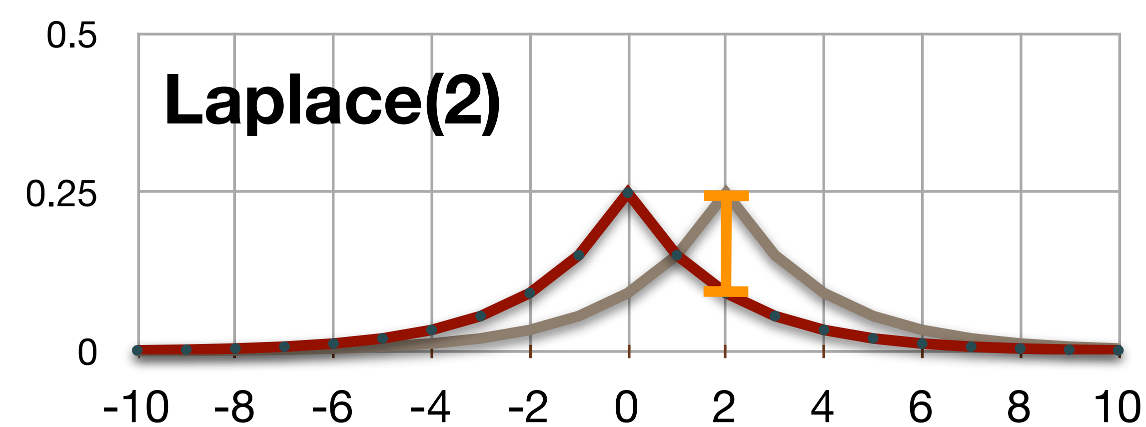
- The following algorithm for answering Q is ε-differentially private:

sensitivity of Q

privacy parameter

$$\mathcal{A} \longrightarrow Q(G) + \text{Laplace}(\,\Delta Q \,/\, \varepsilon\,)$$





Examples of query sensitivity

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

query **sensitivity** **truth** **noisy answer**

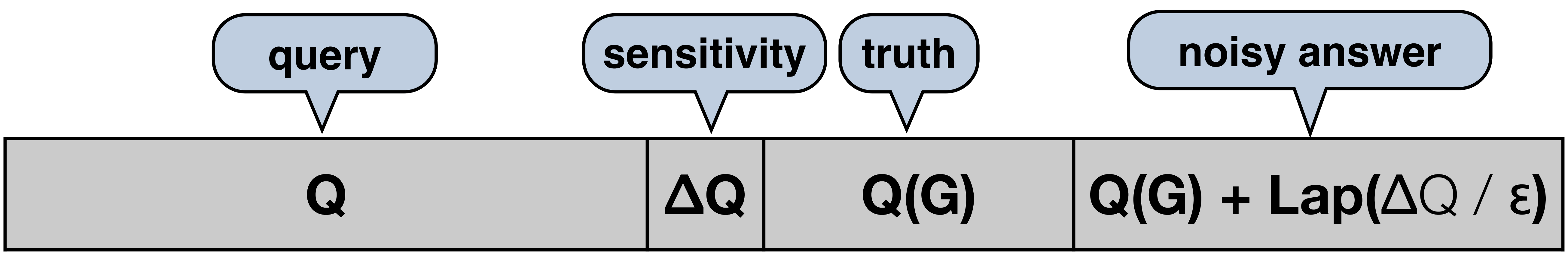
ε=0.5

Examples of query sensitivity

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs



ε=0.5

Examples of query sensitivity

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

query — sensitivity — truth — noisy answer

Q	**ΔQ**	**Q(G)**	**Q(G) + Lap(**ΔQ / ε**)**
$\mathbf{deg_A}$ (degree of node A)	1	$\mathbf{deg_{Dave}}$(G) = 4	4+Lap(2)

ε=0.5

Examples of query sensitivity

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

query — sensitivity — truth — noisy answer

Q	ΔQ	Q(G)	Q(G) + Lap(ΔQ / ε)
$\mathbf{deg_A}$ (degree of node A)	1	$\mathbf{deg_{Dave}}(G) = 4$	4+Lap(2)
$\mathbf{cnt_i}$ (# nodes with degree i)	2	$\mathbf{cnt_4}(G) = 4$	4+Lap(4)

ε=0.5

Multiple queries

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

L_1 dist for vectors

query

sensitivity

truth

noisy answer

ε=0.5

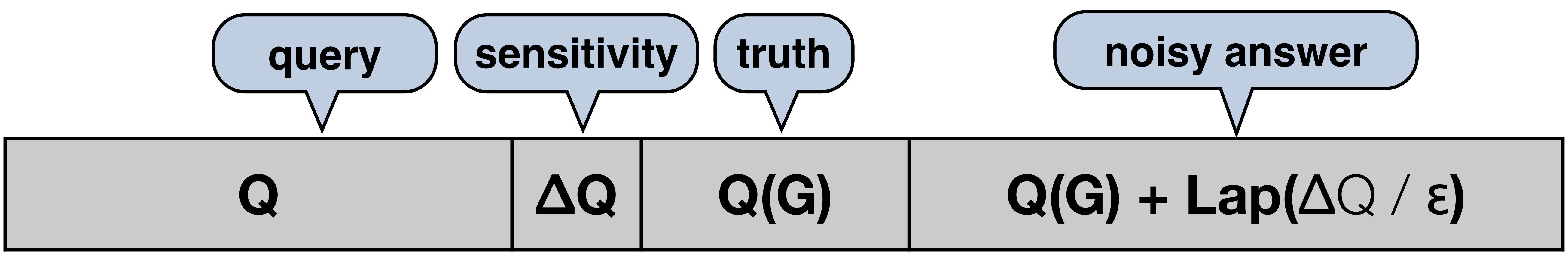
Multiple queries

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

L_1 dist for vectors



ε=0.5

Multiple queries

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

L_1 dist for vectors

query: Q	sensitivity: ΔQ	truth: Q(G)	noisy answer: Q(G) + Lap(ΔQ / ε)
[deg_A, deg_B, deg_C]	2	[1,4,1]	[1+Lap(4), 4+Lap(4), 1+Lap(4)]

ε=0.5

Multiple queries

The sensitivity of a query Q is

$$\Delta Q = \max_{G,G'} | Q(G) - Q(G') |$$

where G, G' are any two neighboring graphs

L_1 dist for vectors

Q (query)	ΔQ (sensitivity)	Q(G) (truth)	Q(G) + Lap(ΔQ / ε) (noisy answer)
$[deg_A, deg_B, deg_C]$	2	[1,4,1]	[1+Lap(4), 4+Lap(4), 1+Lap(4)]
$[cnt_0, cnt_1, cnt_2]$	4	[0,2,2]	[0+Lap(8), 2+Lap(8), 2+Lap(8)]

$\varepsilon = 0.5$

Differential privacy for networks

A participant’s sensitive information is **not** a single edge.

- **edge ε-differential privacy**: algorithm output is largely indistinguishable whether or not any **single edge** is present or absent.
- **k-edge ε-differential privacy**: algorithm output is largely indistinguishable whether or not any **set of k edges** is present or absent.
- **node ε-differential privacy**: algorithm output is largely indistinguishable whether or not any single **node (and all its edges)** is present or absent.

Laplace($\Delta Q / \varepsilon$ **)**

↓

Laplace($\Delta Q\ k / \varepsilon$ **)**

Suppose ΔQ=1. Then Laplace(100) satisfies:
1-edge 0.01-differential privacy
10-edge 0.1-differential privacy

Outline

1. Existing approaches to protecting network data
2. Background on differential privacy
3. Privately estimating the degree distribution
4. Privately counting motifs
5. Future goals and open questions

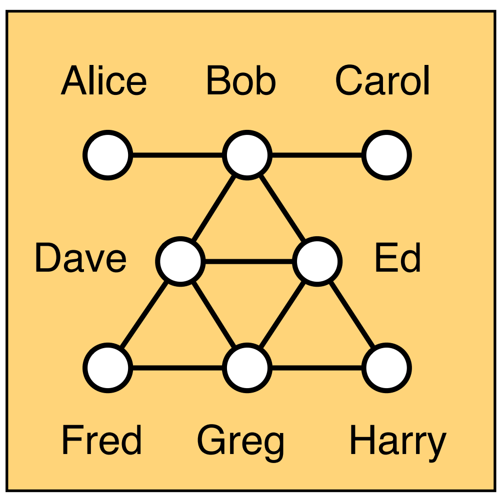
Outline

1. Existing approaches to protecting network data
2. Background on differential privacy
3. Privately estimating the degree distribution
4. Privately counting motifs
5. Future goals and open questions

The degree sequence of a network

- Degree sequence: the list of degrees of each node in a graph.

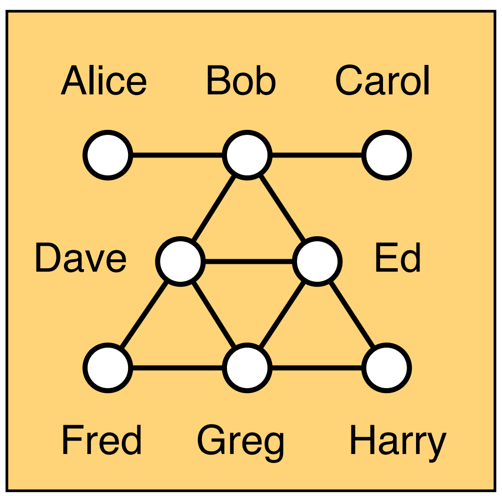
- A widely studied property of networks.



[1,1,2,2,4,4,4,4]

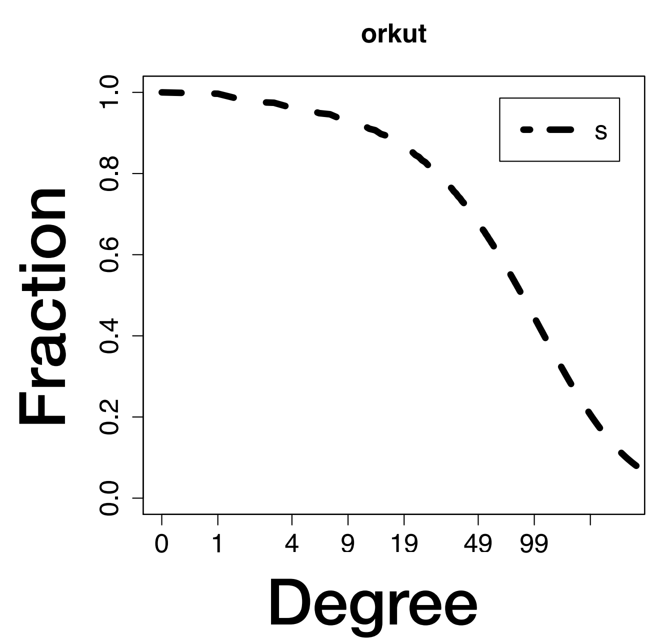
The degree sequence of a network

- Degree sequence: the list of degrees of each node in a graph.
- A widely studied property of networks.



[1,1,2,2,4,4,4,4]

Orkut crawl



Inverse cummulative distribution

The degree sequence is sensitive

- Why not release the true degree sequence of a network?
  - In extreme cases, the degree sequence can determine the structure of the graph --- no better than naive anonymization.
  - Background knowledge could lead to disclosures.
  - The degree sequence may not be the only statistic we release -- we must protect against combined disclosures.

Two basic queries for degrees

Degree of each node	
deg_A	degree of node A
D	$[deg_A, deg_B, \ldots\quad]$

Frequency of each degree	
cnt_i	count of nodes with
F	$[cnt_0, cnt_1, \ldots\quad cnt_{n-1}]$

Two basic queries for degrees

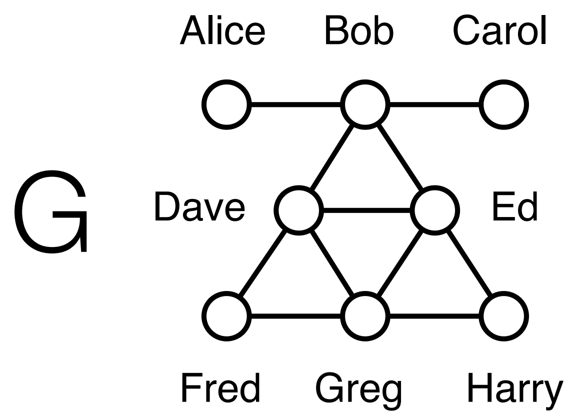
Degree of each node	
deg_A	degree of node A
D	$[deg_A, deg_B, \ldots\ ]$

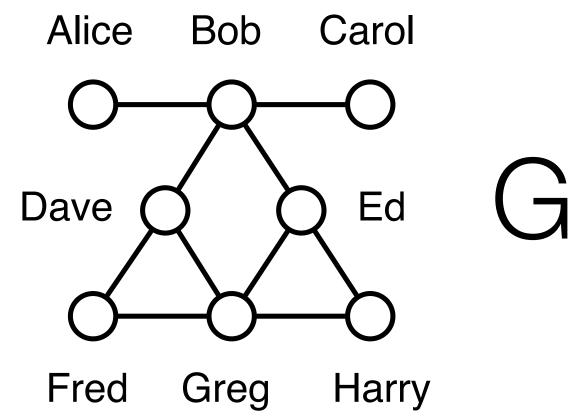
Frequency of each degree	
cnt_i	count of nodes with
F	$[cnt_0, cnt_1, \ldots\ cnt_{n-1}]$

D(G) = [1,4,1,4,4,2,4,2]	
D(G') = [1,4,1,<u>3</u>,<u>3</u>,2,4,2]	

$\Delta D=2$

Two basic queries for degrees





Degree of each node	
deg_A	degree of node A
D	$[deg_A, deg_B, \ldots \quad]$

Frequency of each degree	
cnt_i	count of nodes with
F	$[cnt_0, cnt_1, \ldots \quad cnt_{n-1}]$

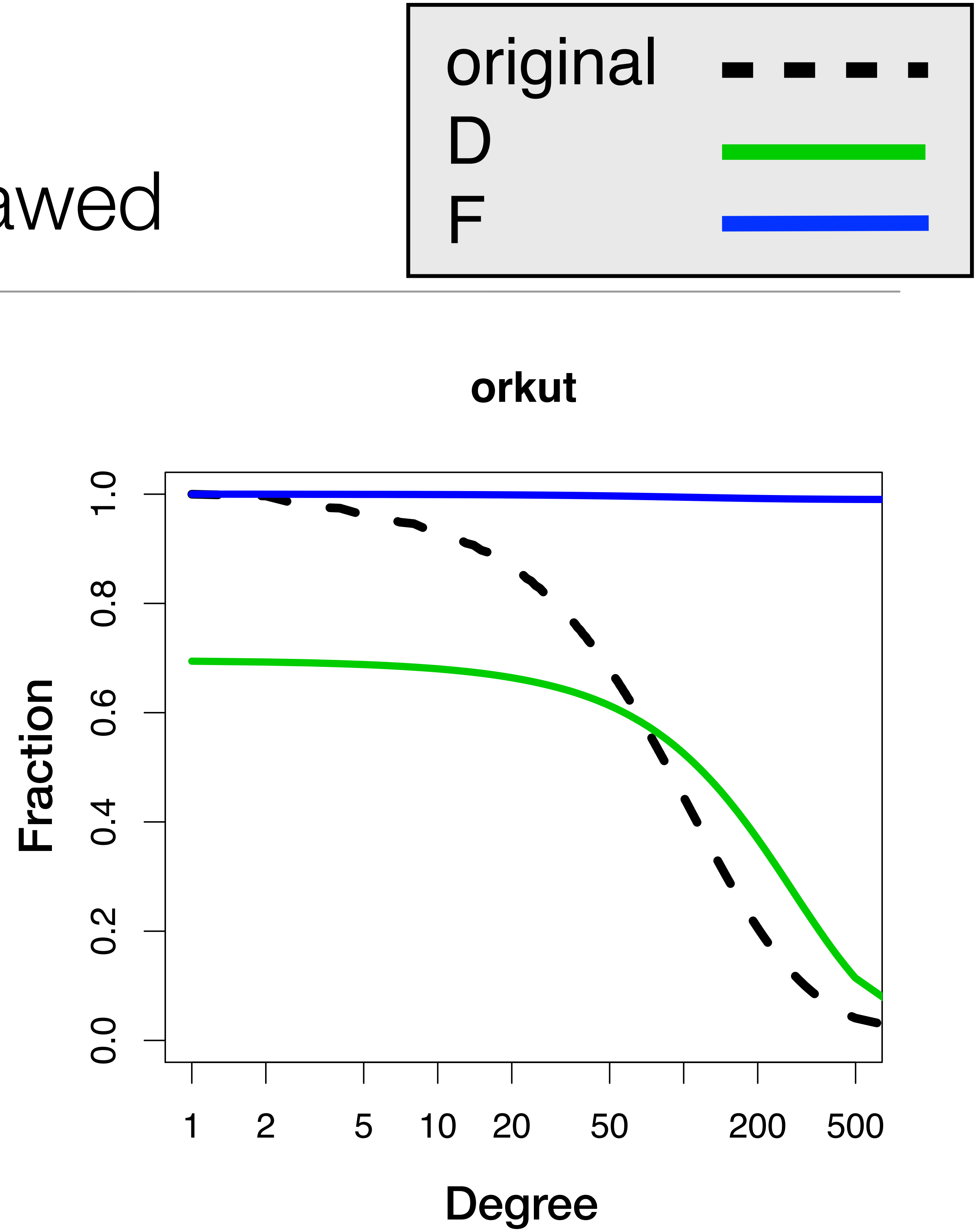
$D(G) = [1,4,1,4,4,2,4,2]$	$F(G) = [0,2,2,0,4,0,0,0]$
$D(G') = [1,4,1,\underline{3},\underline{3},2,4,2]$	$F(G') = [0,2,2,\underline{2},\underline{2},0,0,0]$

$\Delta D=2$

$\Delta F=4$

These queries are both flawed

- D requires independent samples from Laplace(2/ε) in each component.
- F requires independent samples from Laplace(4/ε) in each component.
- Thus Mean Squared Error is $O(n/\varepsilon^2)$



(Laplace(b) has variance $2b^2$)

An alternative query for degrees

Degree of each node	
deg_A	degree of node A
D	[deg_A, deg_B, ...]

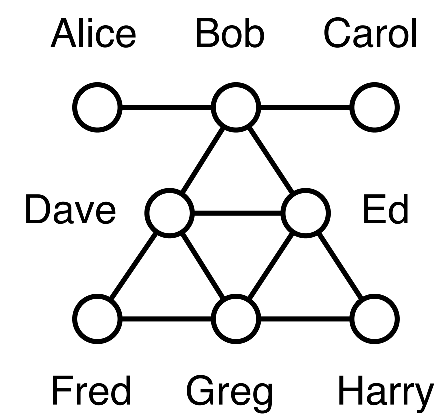
Degree of each node, ranked	
rnk_i	return the rank i^{th} degree
S	[rnk_1, rnk_2, ... rnk_n]

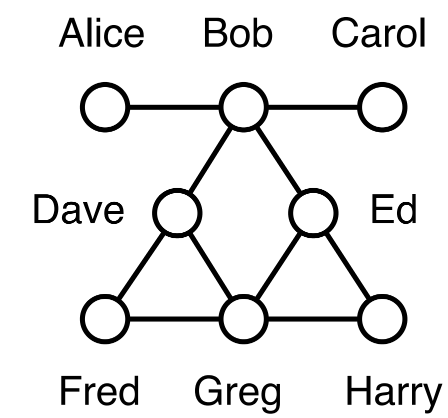
D(G) = [1,4,1,4,4,2,4,2]
D(G') = [1,4,1,<u>3</u>,<u>3</u>,2,4,2]

$\Delta D=2$

An alternative query for degrees

G





G’

Degree of each node	
deg_A	degree of node A
D	$[deg_A, deg_B, \ldots\ ]$

Degree of each node, ranked	
rnk_i	return the rank i^{th} degree
S	$[rnk_1, rnk_2, \ldots\ rnk_n]$

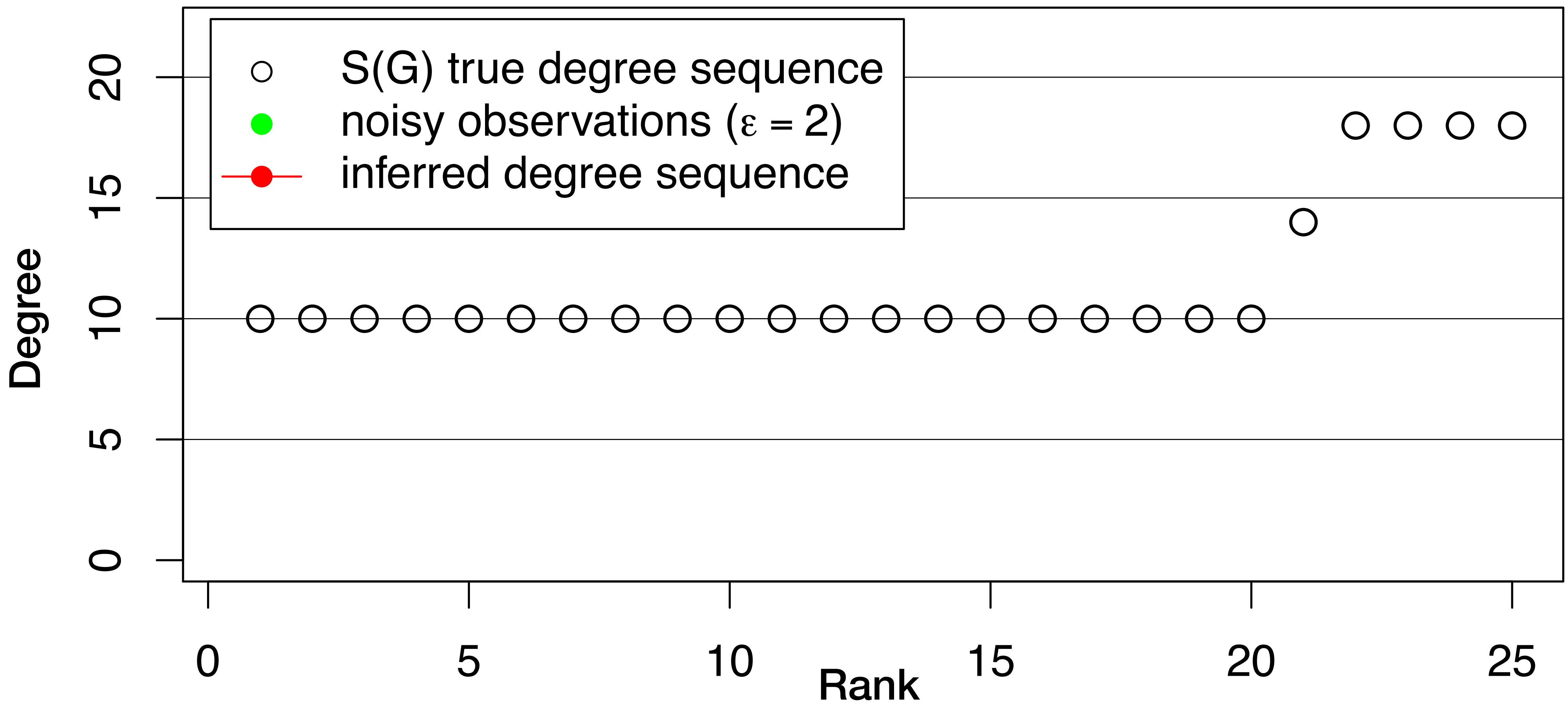
D(G) = [1,4,1,4,4,2,4,2]
D(G’) = [1,4,1,3,3,2,4,2]

S(G) = [1,1,2,2,4,4,4,4]
S(G’) = [1,1,2,2,3,3,4,4]

$\Delta D=2$

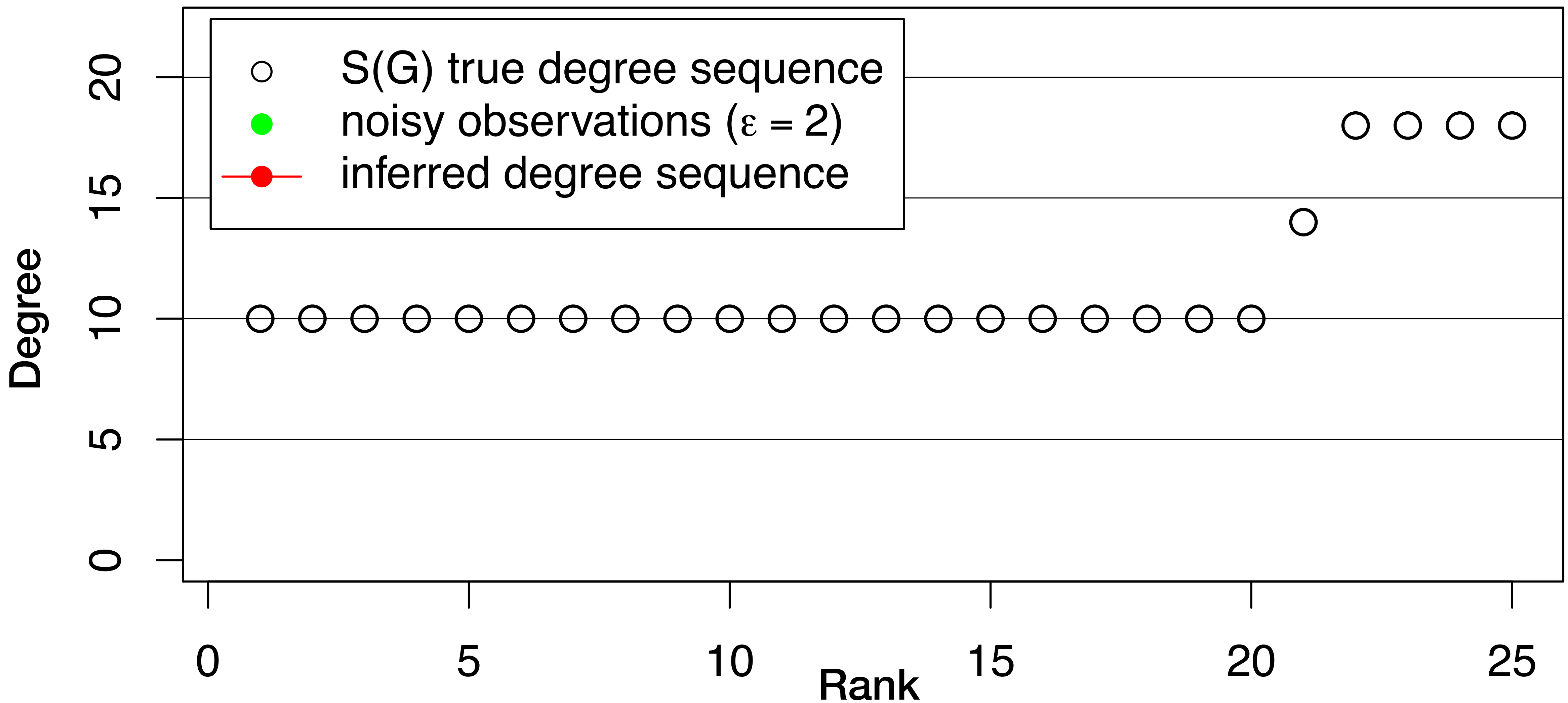
$\Delta S=2$

Using the sort constraint

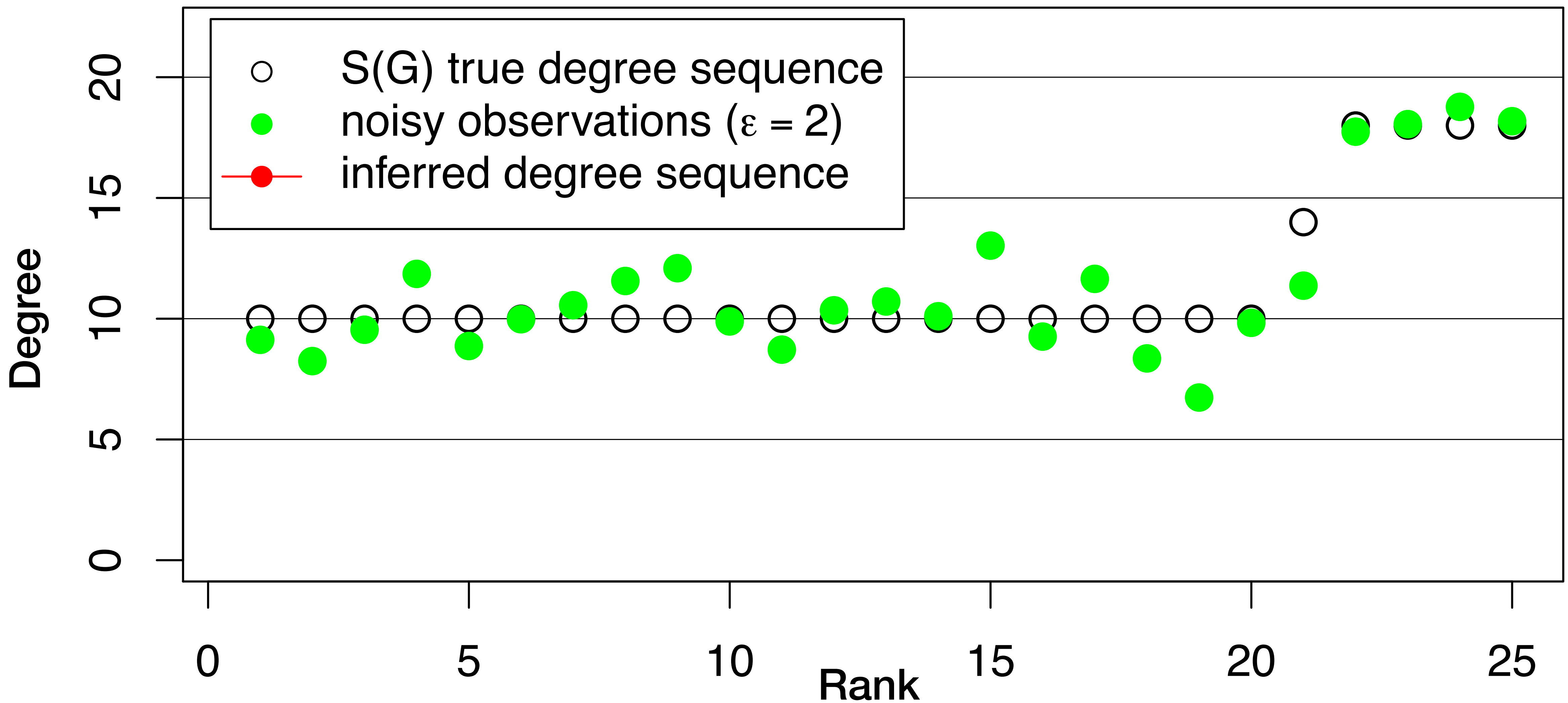


S(G) = [10, 10, ...10, 10, 14, 18,18,18,18]

Using the sort constraint

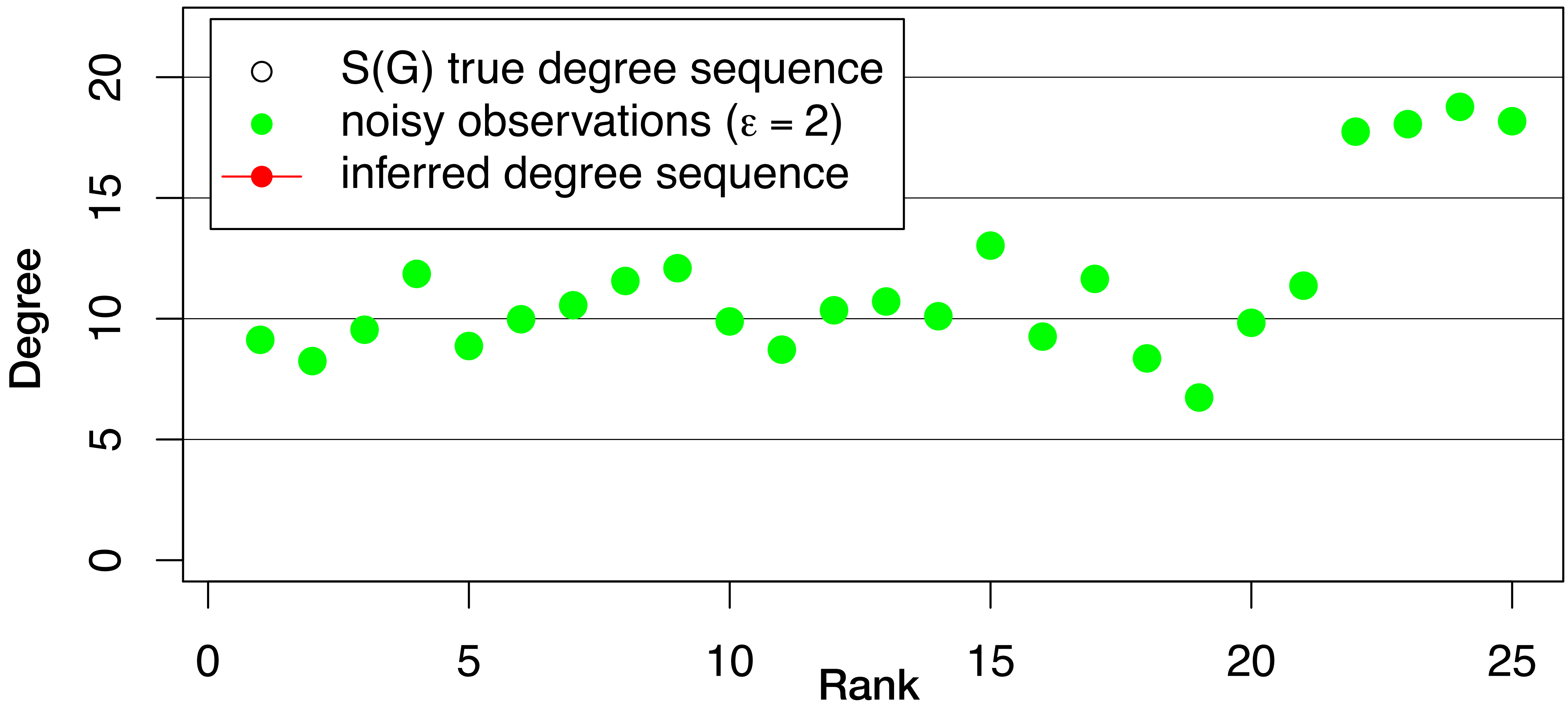


Using the sort constraint



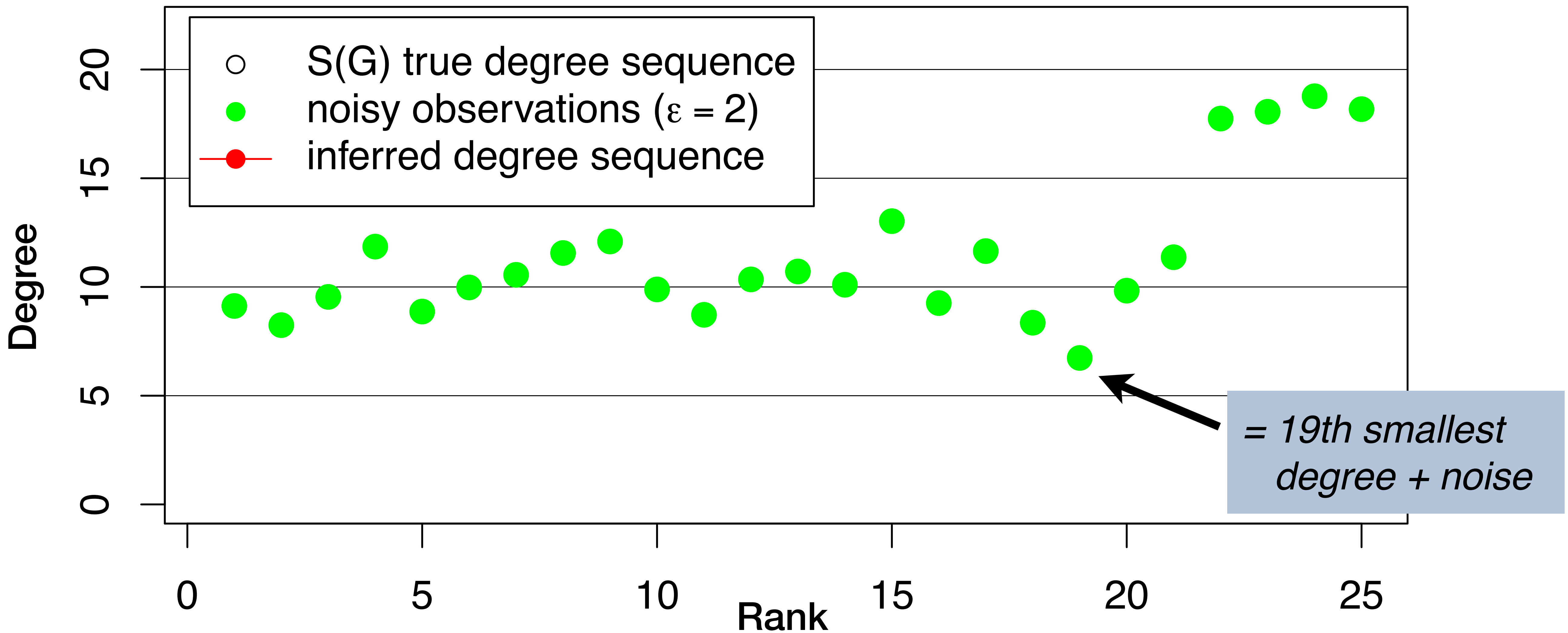
- The output of the sorted degree query is not (in general) sorted.
- We derive a new sequence by computing the **closest** non-decreasing sequence: i.e. minimizing L2 distance.

Using the sort constraint



- The output of the sorted degree query is not (in general) sorted.
- We derive a new sequence by computing the **closest** non-decreasing sequence: i.e. minimizing L2 distance.

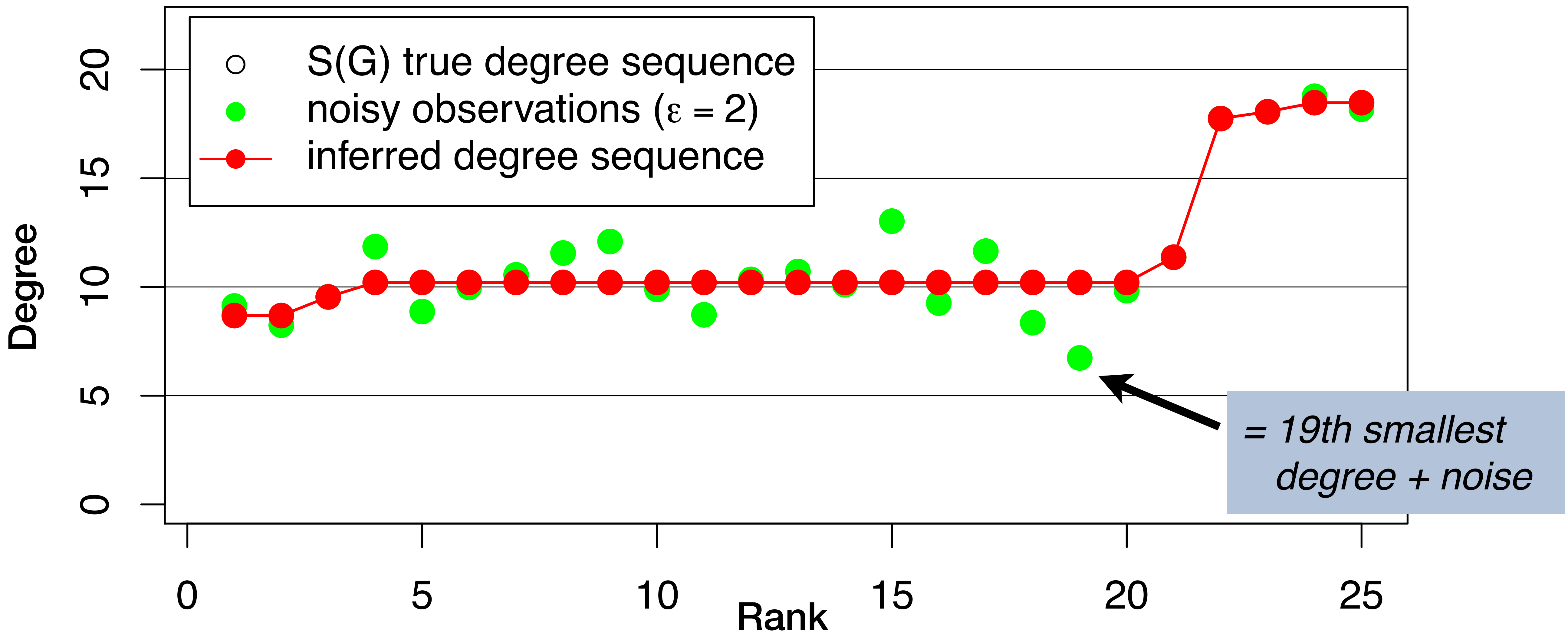
Using the sort constraint



- The output of the sorted degree query is not (in general) sorted.

- We derive a new sequence by computing the **closest** non-decreasing sequence: i.e. minimizing L2 distance.

Using the sort constraint



- The output of the sorted degree query is not (in general) sorted.
- We derive a new sequence by computing the **closest** non-decreasing sequence: i.e. minimizing L2 distance.

Experimental results

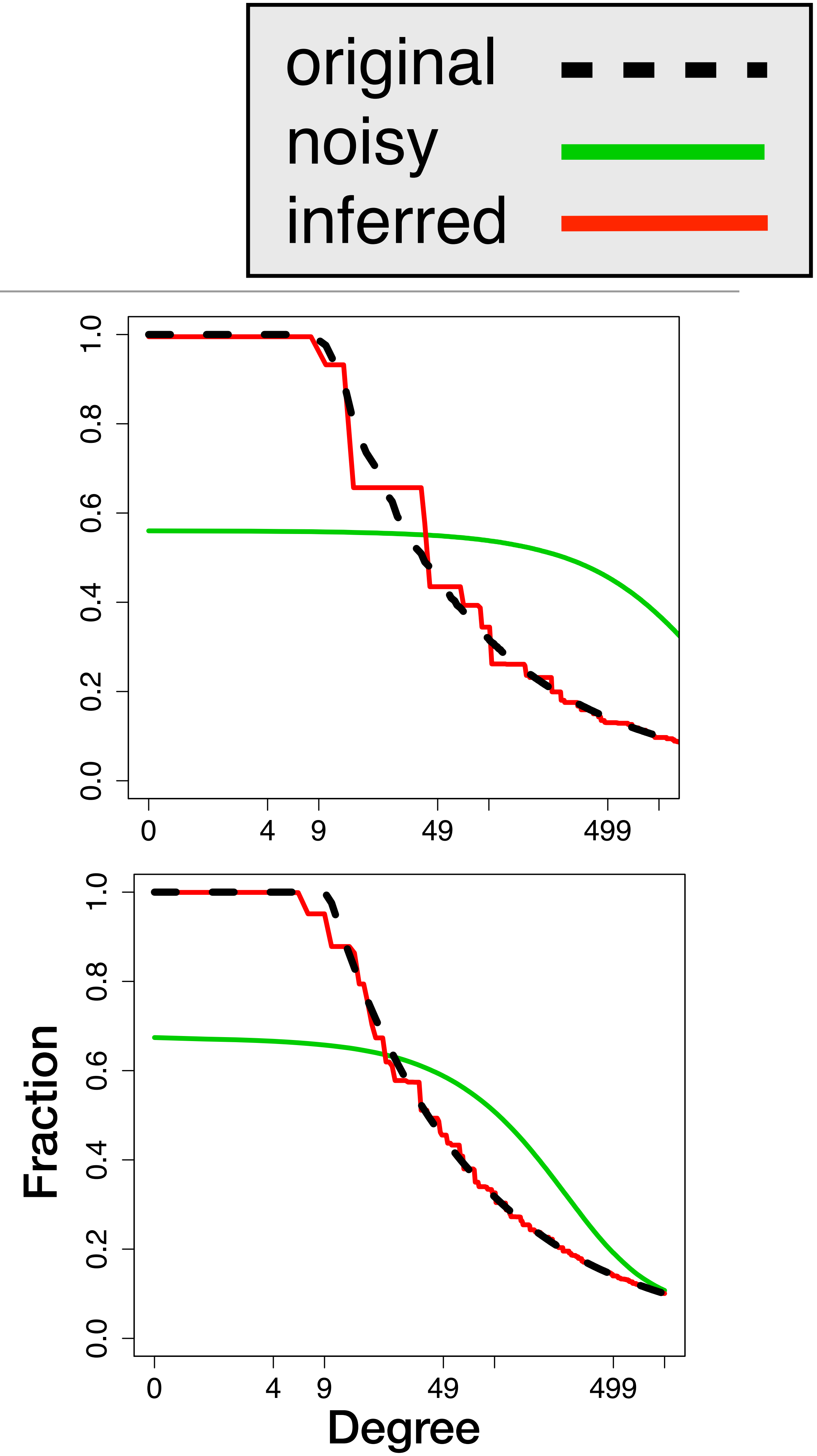
power law, α=1.5, n=5M

(100-edge,
0.1-differential
privacy)

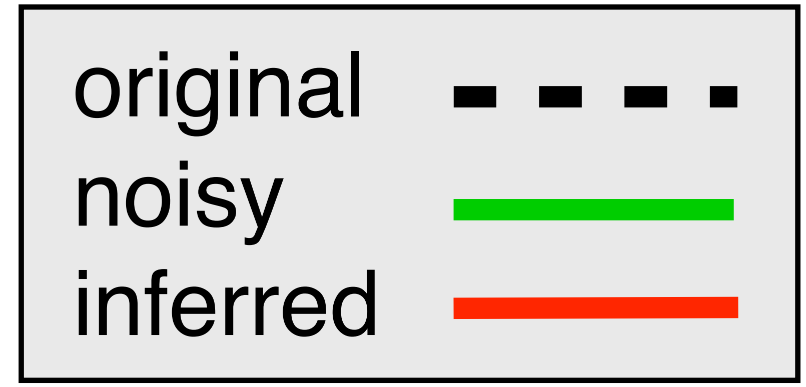
ε=.001

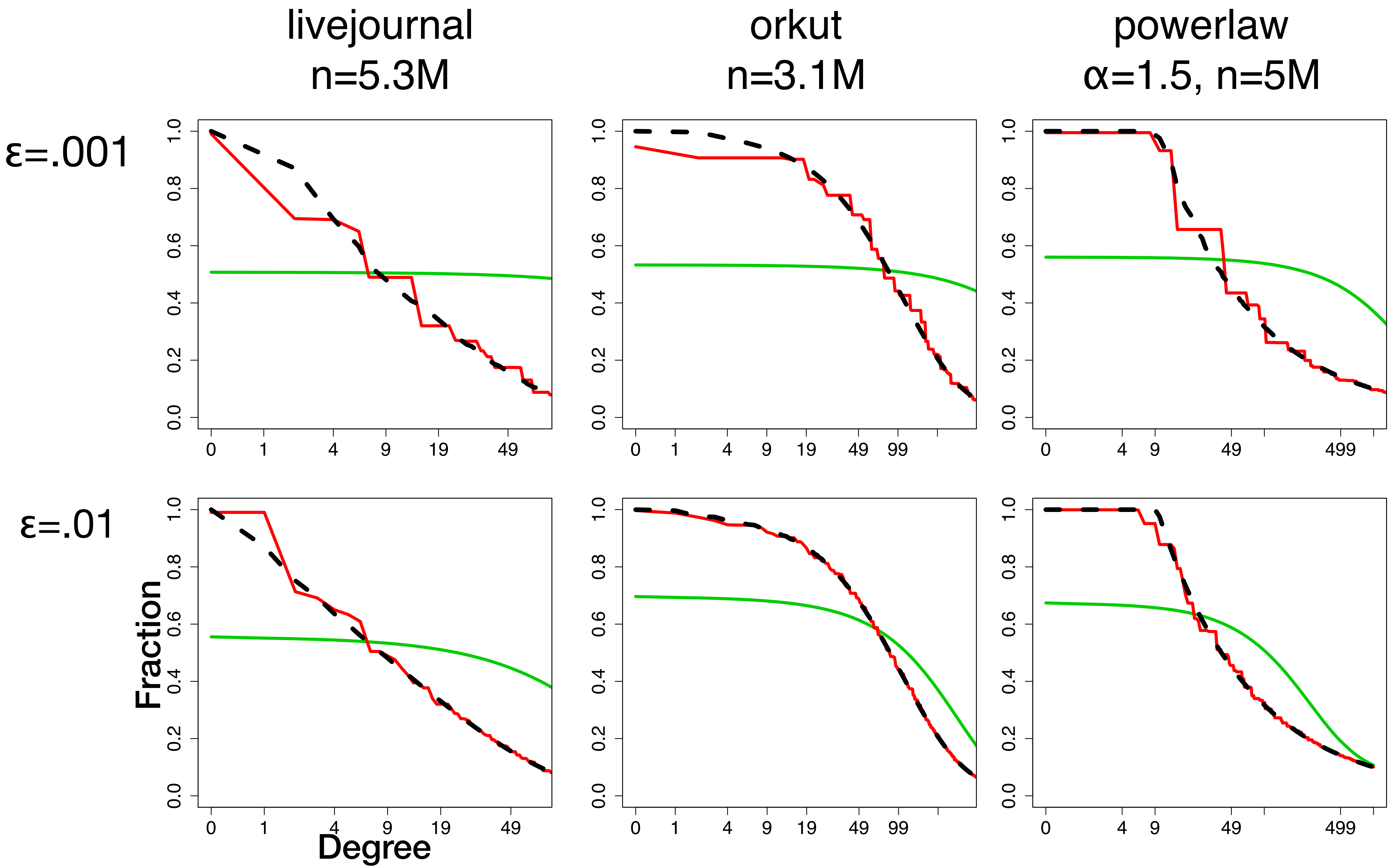
(10-edge,
0.1-differential
privacy)

ε=.01



Experimental results, continued





Inference does **not** weaken privacy

DATA OWNER | **ANALYST**

Inference does **not** weaken privacy

DATA OWNER	ANALYST
	1. Formulate **S,** having constraints $\Upsilon_{\mathbf{S}}$

Inference does **not** weaken privacy

DATA OWNER	ANALYST

← **S**

1. Formulate **S**,
having constraints $\Upsilon_{\mathbf{S}}$

2. Submit **S**

Inference does **not** weaken privacy

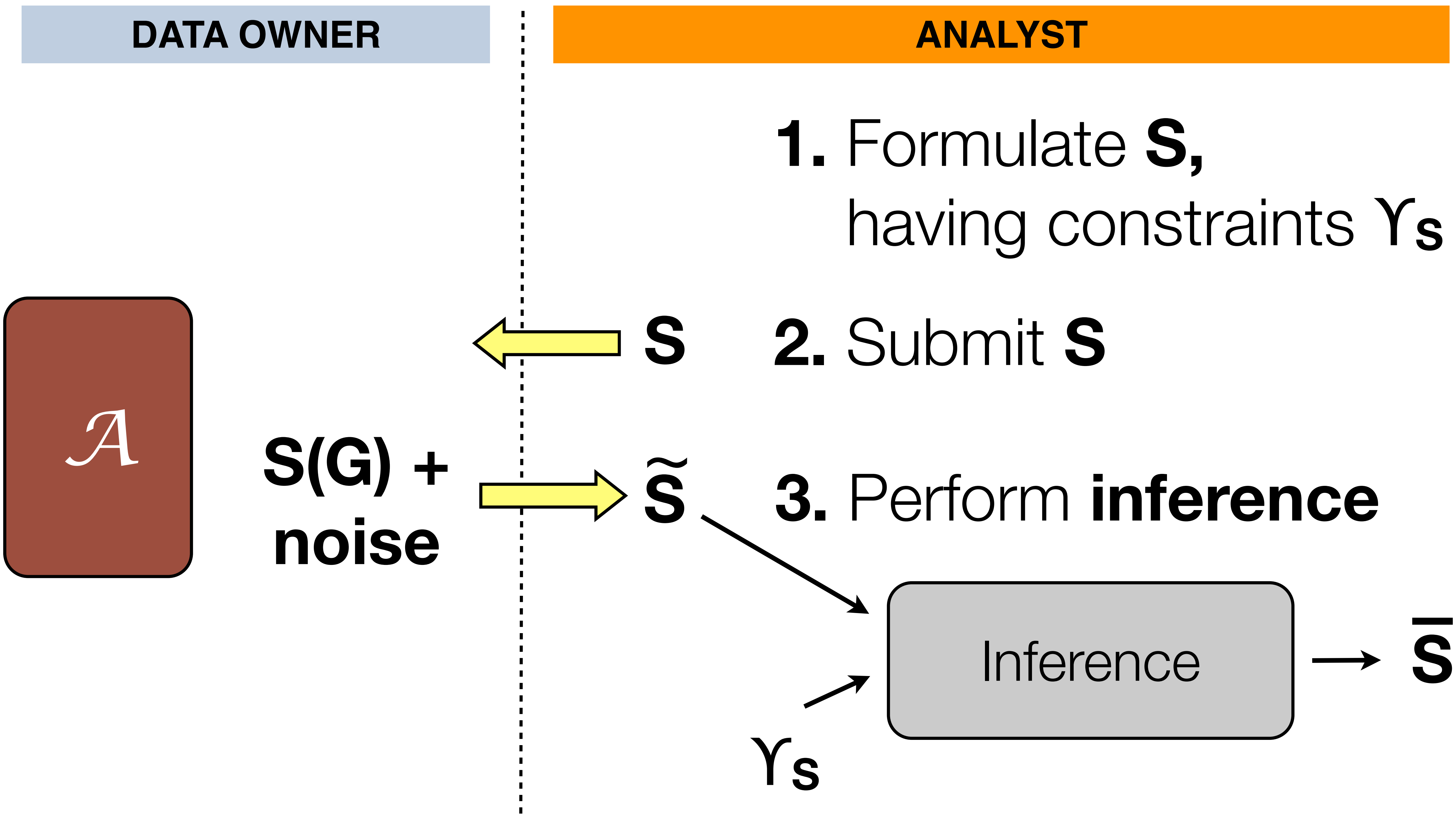
DATA OWNER | **ANALYST**

1. Formulate **S,**
having constraints Υ_S

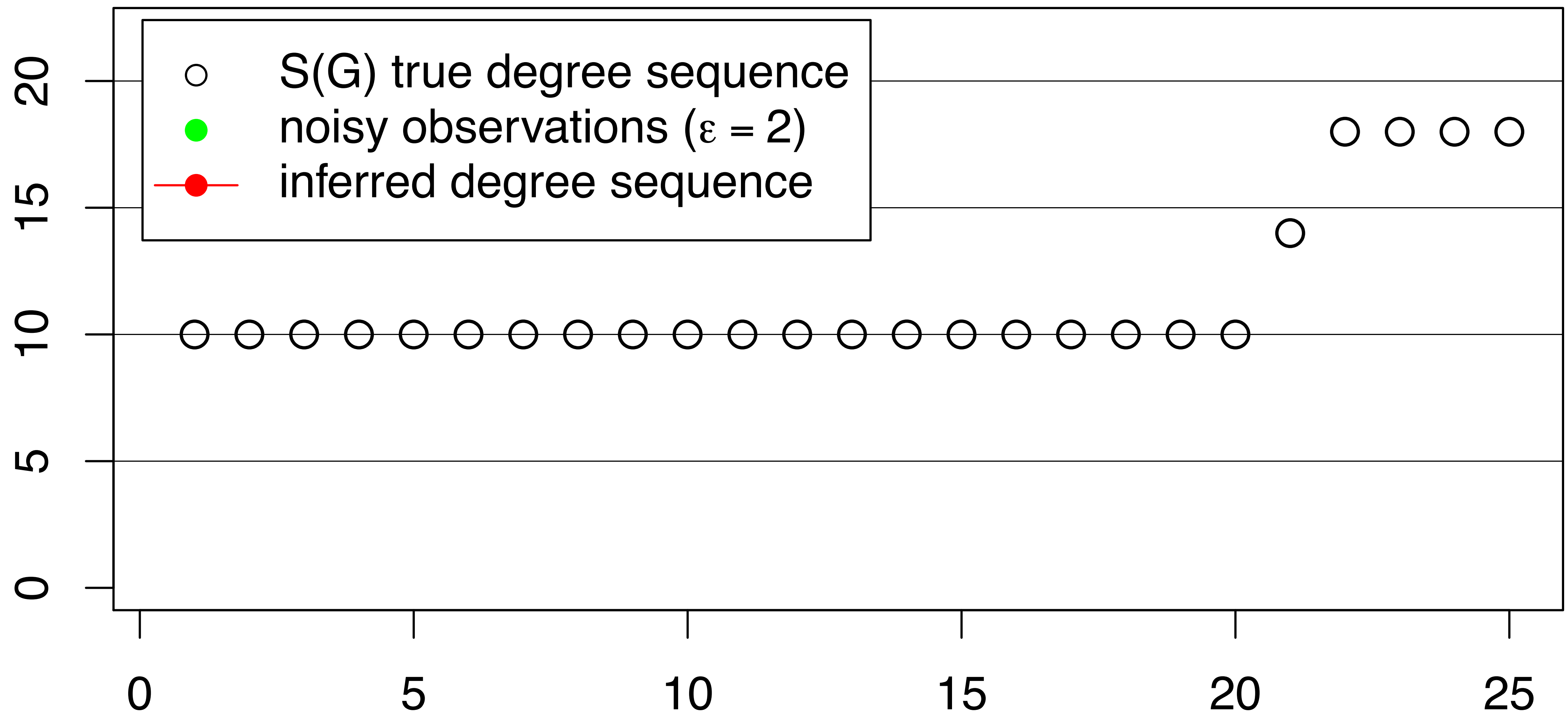
S ← **2.** Submit **S**

S(G) + noise → $\tilde{S}$

Inference does **not** weaken privacy

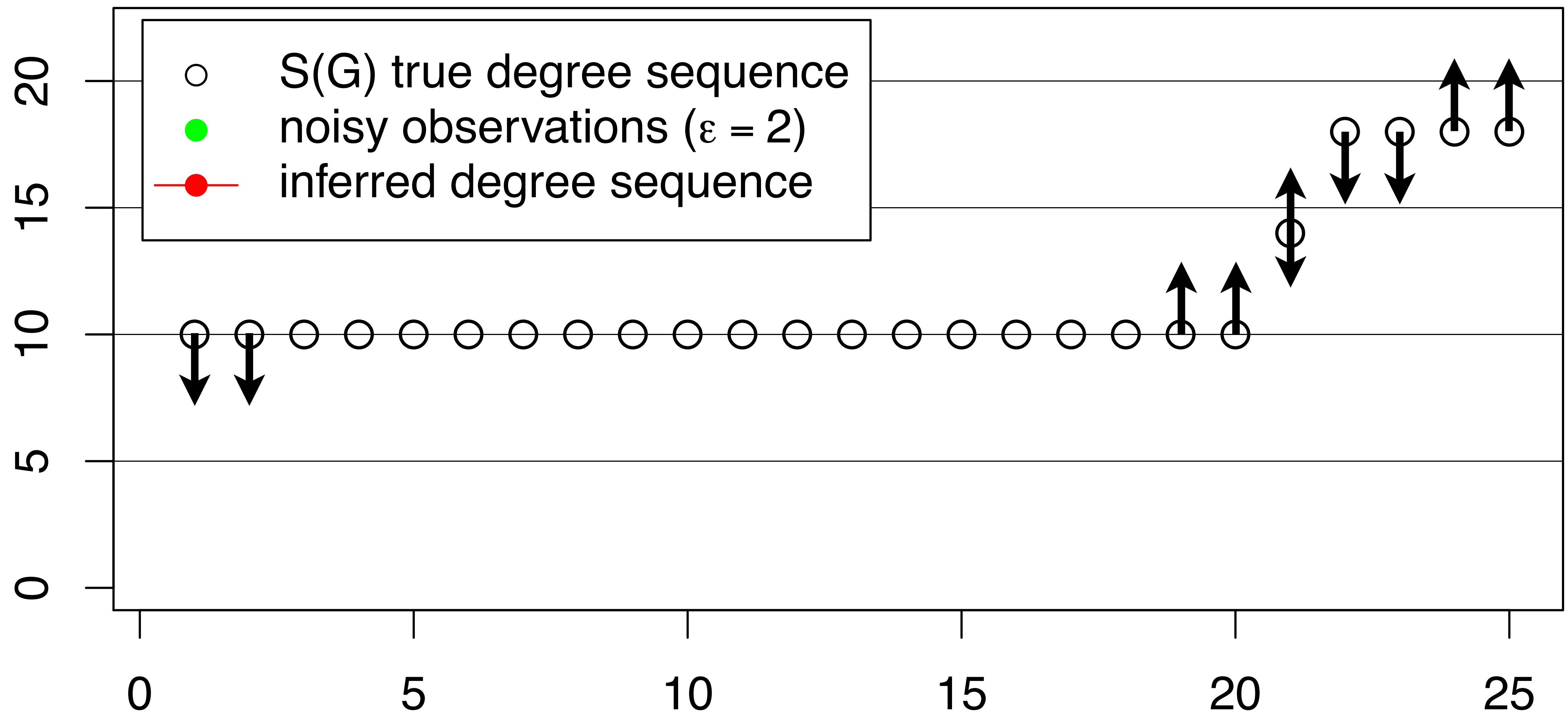


After inference, noise only where needed



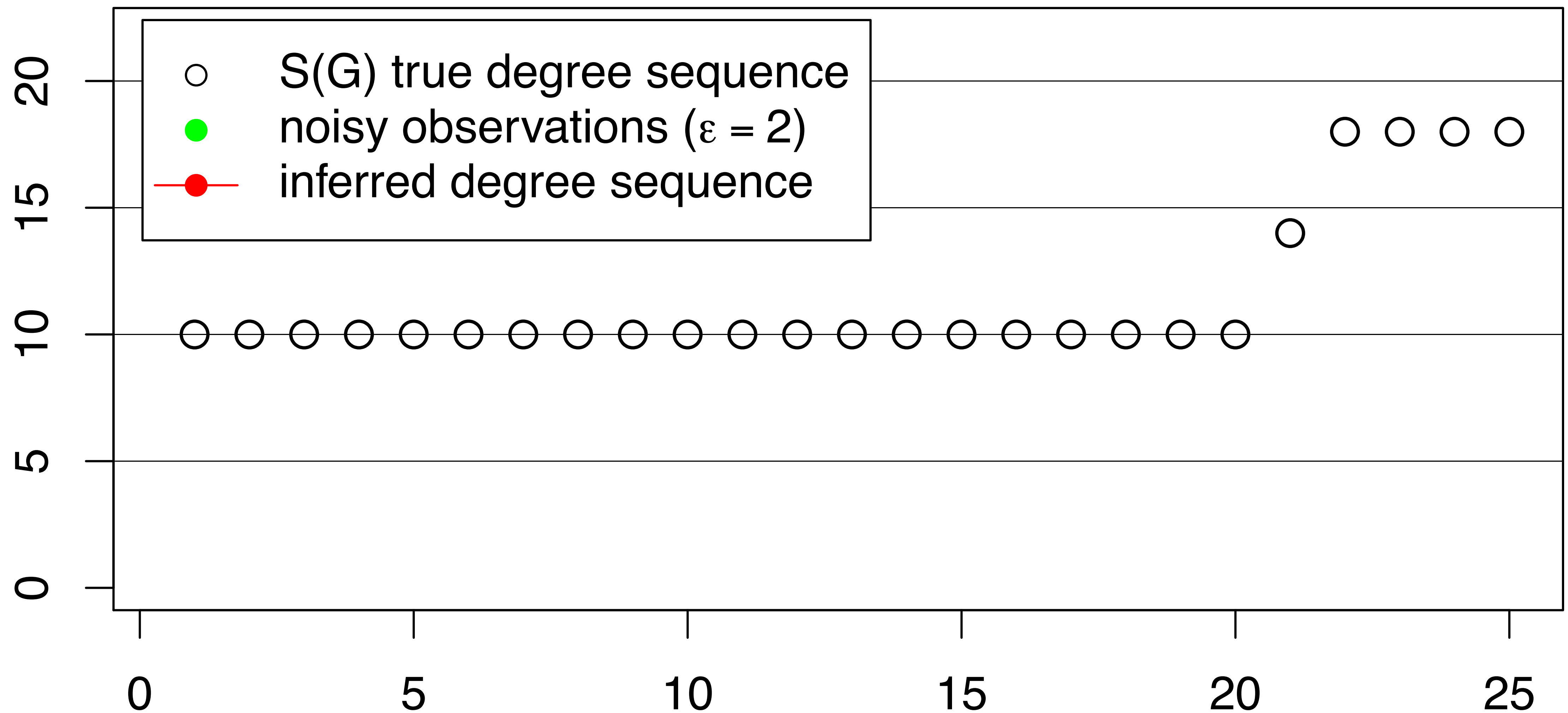
- Standard Laplace noise is sufficient *but not necessary* for differential privacy.
- By using inference, effectively apply a different noise distribution -- more noise where it is needed, less otherwise.
  - Improvement in accuracy will depend on sequence

After inference, noise only where needed



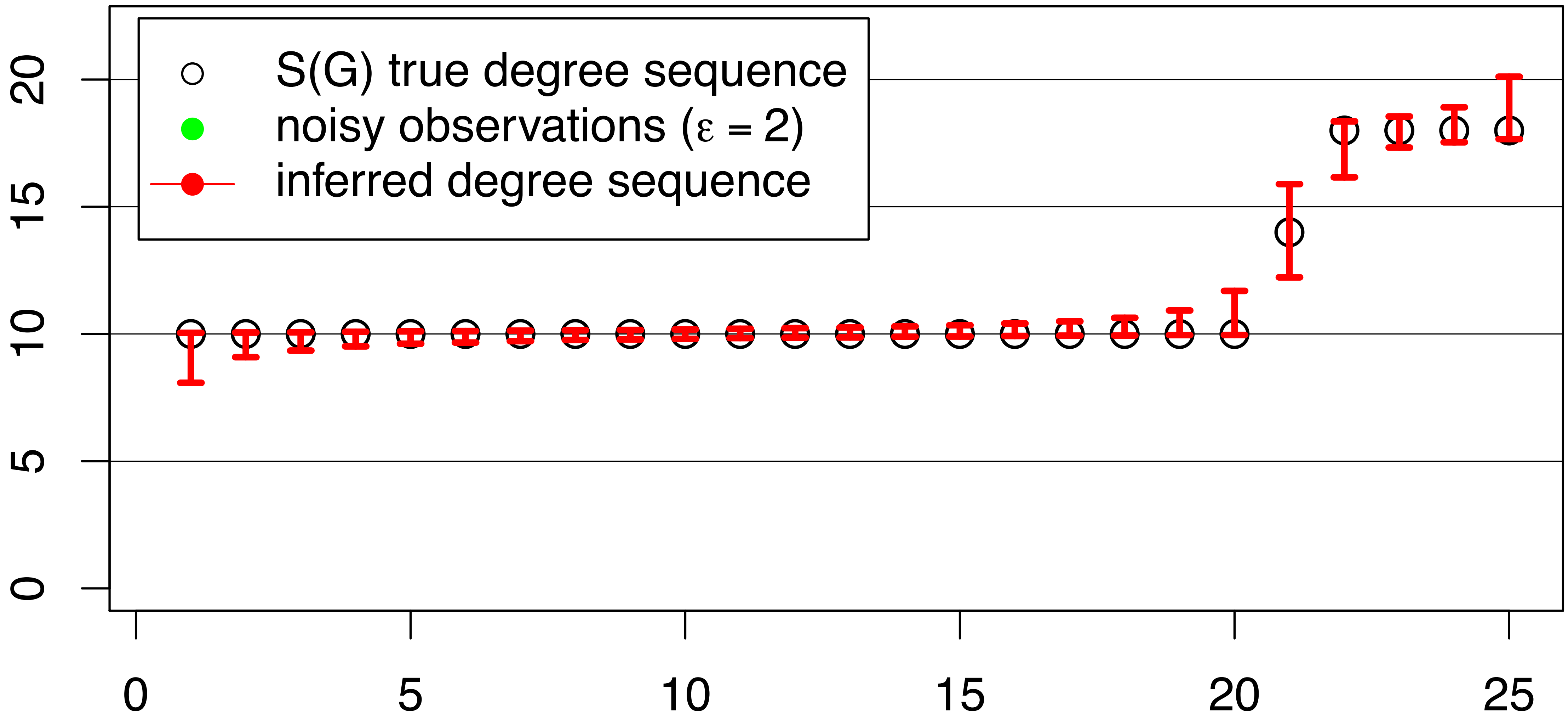
- Standard Laplace noise is sufficient *but not necessary* for differential privacy.
- By using inference, effectively apply a different noise distribution -- more noise where it is needed, less otherwise.
    - Improvement in accuracy will depend on sequence

After inference, noise only where needed



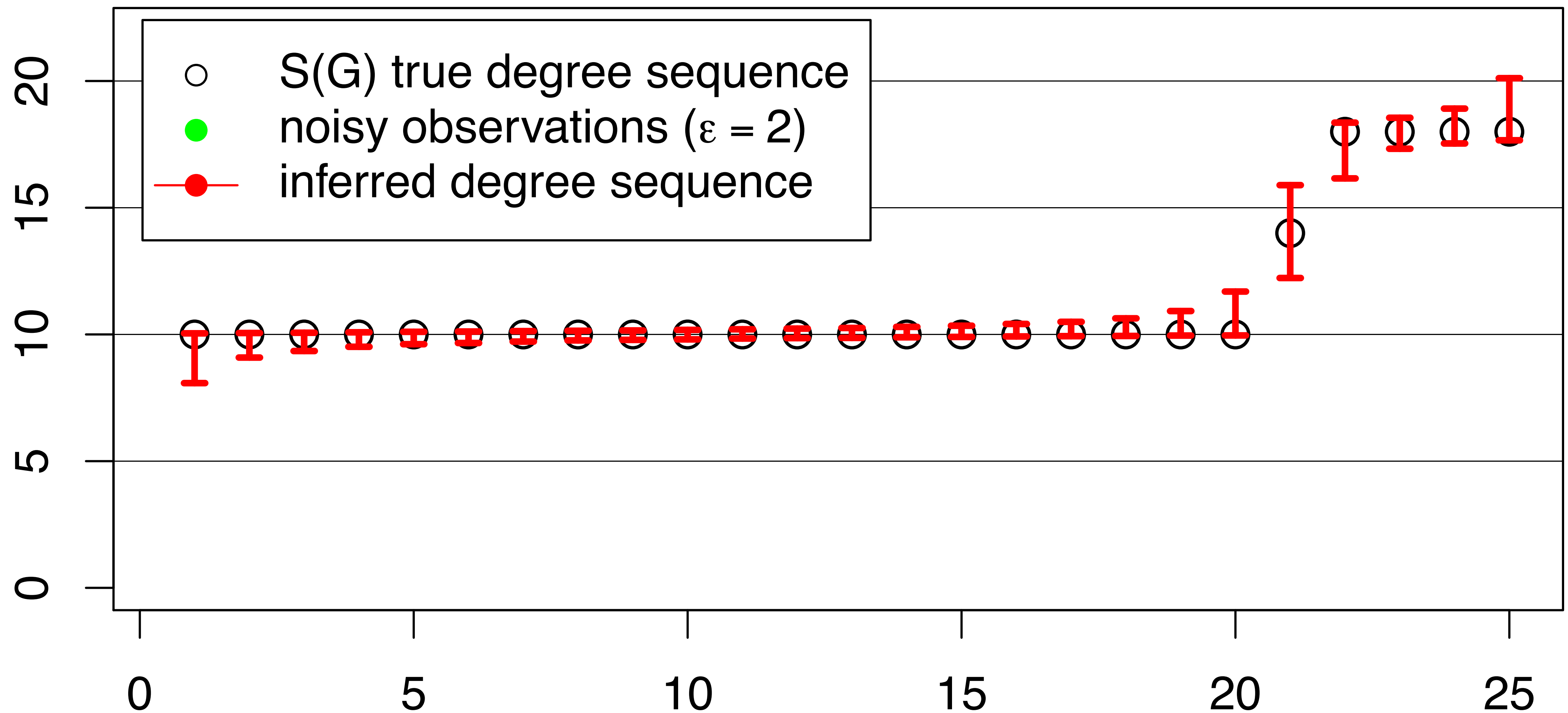
- Standard Laplace noise is sufficient *but not necessary* for differential privacy.
- By using inference, effectively apply a different noise distribution -- more noise where it is needed, less otherwise.
  - Improvement in accuracy will depend on sequence

After inference, noise only where needed



- Standard Laplace noise is sufficient *but not necessary* for differential privacy.
- By using inference, effectively apply a different noise distribution -- more noise where it is needed, less otherwise.
    - Improvement in accuracy will depend on sequence

After inference, noise only where needed



- Standard Laplace noise is sufficient *but not necessary* for differential privacy.
- By using inference, effectively apply a different noise distribution -- more noise where it is needed, less otherwise.
  - Improvement in accuracy will depend on sequence

Accuracy is improved without sacrificing privacy!

- The accuracy achieved **depends on the input sequence**.

Mean Squared Error of Degree Sequence		
Before inference,	$\tilde{\mathbf{S}}$	$\Theta(n/\epsilon^2)$
After inference,	$\bar{\mathbf{S}}$	$O(d \log^3 n/\epsilon^2)$

number of distinct degrees

- Performing inference is efficient: the sorted sequence which minimizes the L2 distance has an elegant closed form solution:
  - shown $O(n^2)$ in **[Hay, PVLDB 10]**
  - improved to $O(n)$ in **[Hay, ICDM 09]**

Outline

1. Existing approaches to protecting network data
2. Background on differential privacy
3. Privately estimating the degree distribution
4. Privately counting motifs
5. Future goals and open questions

Outline

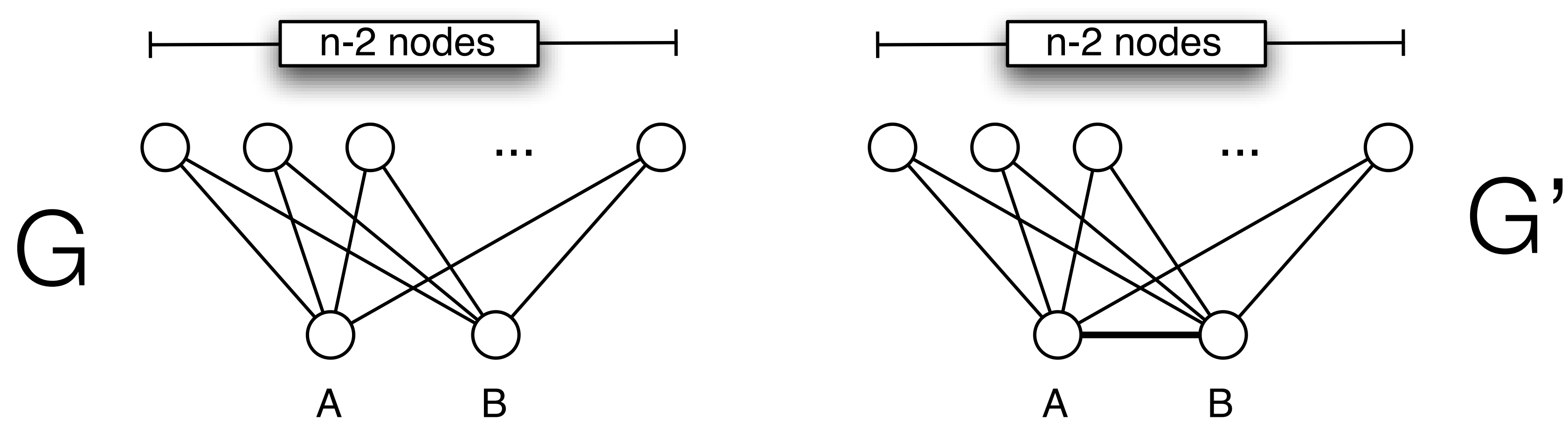
1. Existing approaches to protecting network data
2. Background on differential privacy
3. Privately estimating the degree distribution
4. Privately counting motifs
5. Future goals and open questions

Accurate motif analysis is hard

- Motif analysis measures the frequency of occurrence of small subgraphs in a network.
- Common example: **transitivity** in the network:
    - when A is friends with B and C, are B and C also friends?
    - $\mathbf{Q_{TRIANGLE}}$: return the number of triangles in the graph

Accurate motif analysis is hard

- Motif analysis measures the frequency of occurrence of small subgraphs in a network.
- Common example: **transitivity** in the network:
  - when A is friends with B and C, are B and C also friends?
  - **QTRIANGLE**: return the number of triangles in the graph

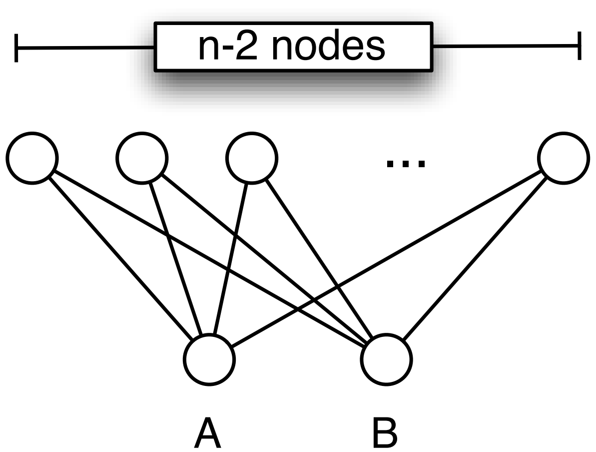


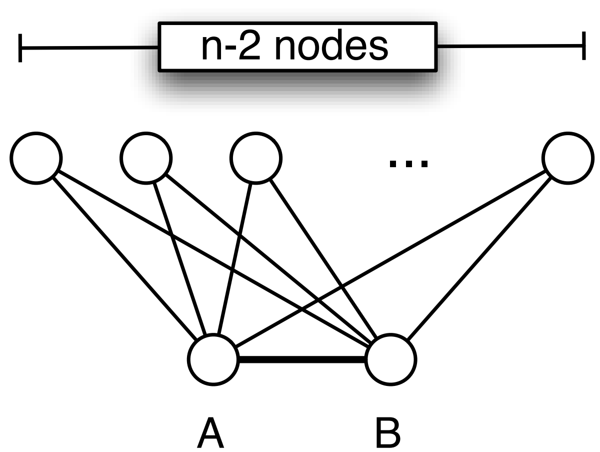
QTRIANGLE (G) = 0   **QTRIANGLE (G') = n-2**

Accurate motif analysis is hard

- Motif analysis measures the frequency of occurrence of small subgraphs in a network.
- Common example: **transitivity** in the network:
  - when A is friends with B and C, are B and C also friends?
  - **QTRIANGLE**: return the number of triangles in the graph

G





G’

High Sensitivity:

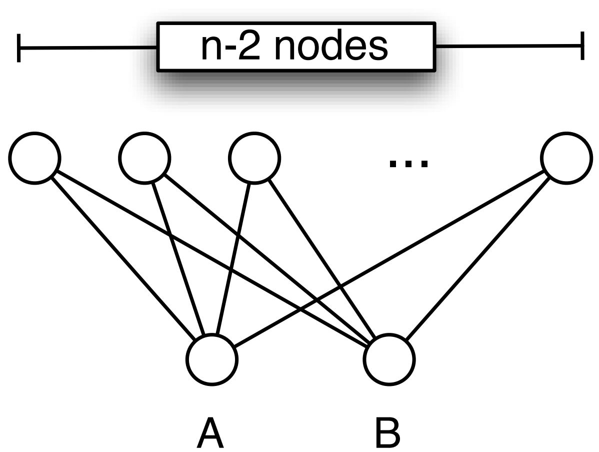
ΔQTRIANGLE=O(n)

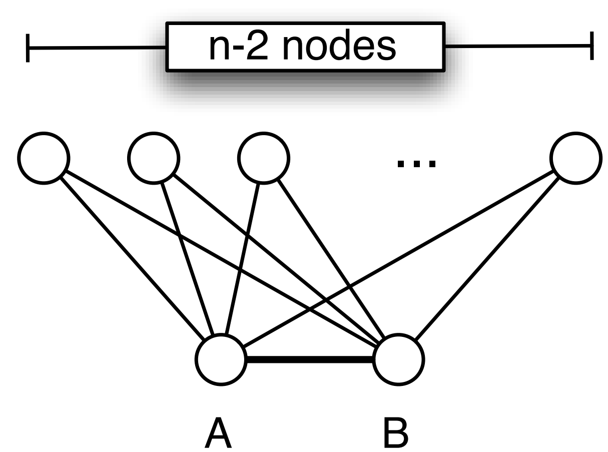
QTRIANGLE (G) = 0   **QTRIANGLE (G’) = n-2**

Accurate motif analysis is hard

- Motif analysis measures the frequency of occurrence of small subgraphs in a network.
- Common example: **transitivity** in the network:
  - when A is friends with B and C, are B and C also friends?
  - **QTRIANGLE**: return the number of triangles in the graph

G





G’

High Sensitivity:

ΔQTRIANGLE=O(n)

QTRIANGLE (G) = 0　　**QTRIANGLE (G’) = n-2**

Accurate motif analysis requires weakening privacy

- There exist output perturbation methods that achieve significantly better accuracy--expected error $\Theta(\log^2 n)$ instead of $\Theta(n)$:
  - **[Rastogi, PODS 09]** Limiting assumptions on the prior knowledge of the adversary, and satisfying adversarial privacy.
    - works for general class of “motif” queries.
  - **[Nissim, STOC 07]** Under certain assumptions about the input graphs, and a modest relaxation of differential privacy:
    - works only for triangle queries (but could be extended).

Outline

1. Existing approaches to protecting network data

2. Background on differential privacy

3. Privately estimating the degree distribution

4. Privately counting motifs

5. Future goals and open questions

Outline

1. Existing approaches to protecting network data

2. Background on differential privacy

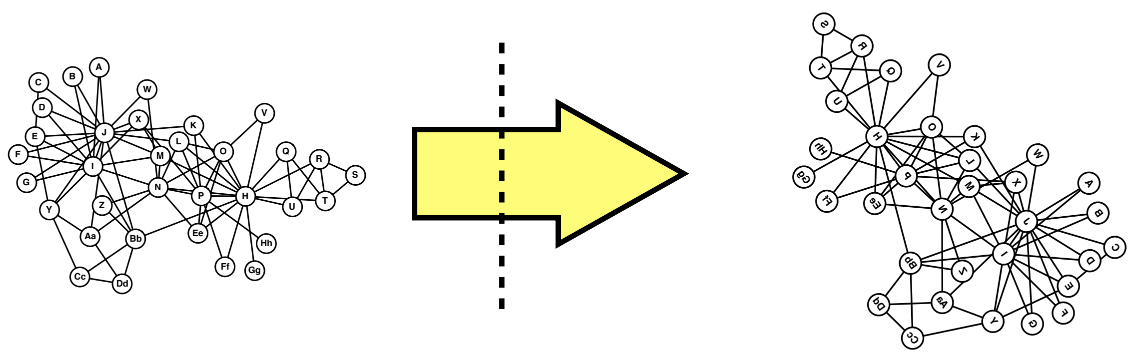
3. Privately estimating the degree distribution

4. Privately counting motifs

5. Future goals and open questions

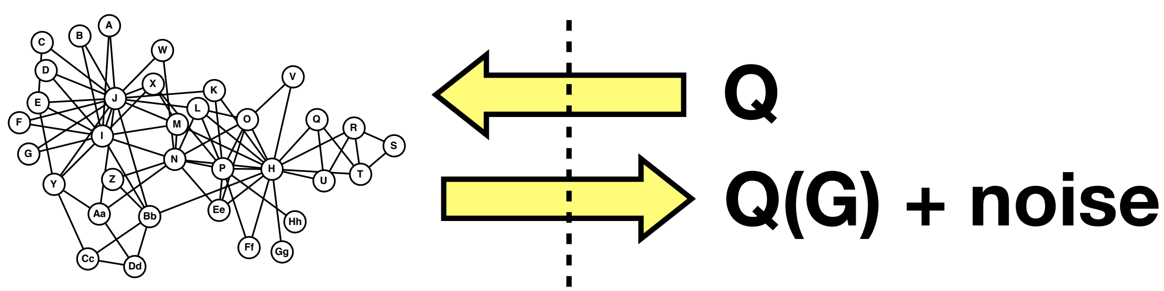
Data publishing v. output perturbation

- Data publishing



Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

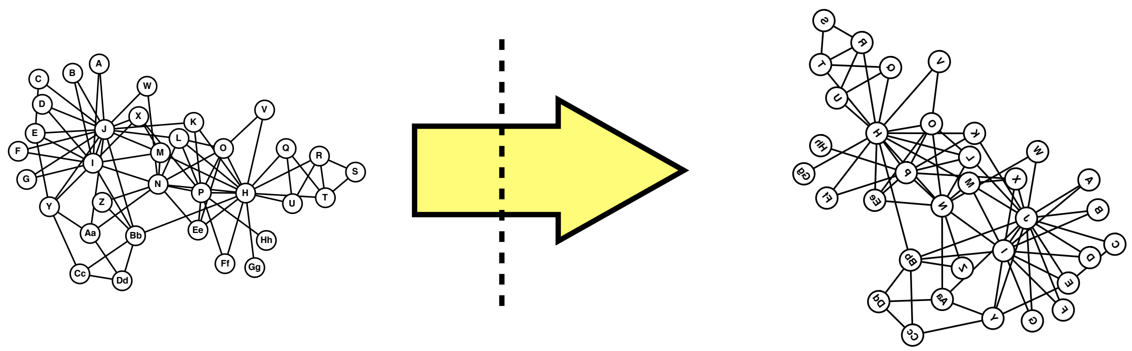
- Output perturbation



Ease of use	bad for practical
Privacy	formal guarantees
Accuracy	provable bounds
Scalability	very good

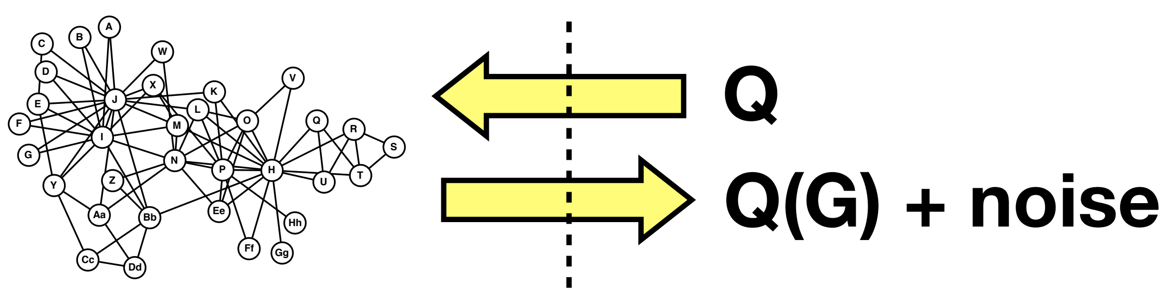
Data publishing v. output perturbation

- Data publishing



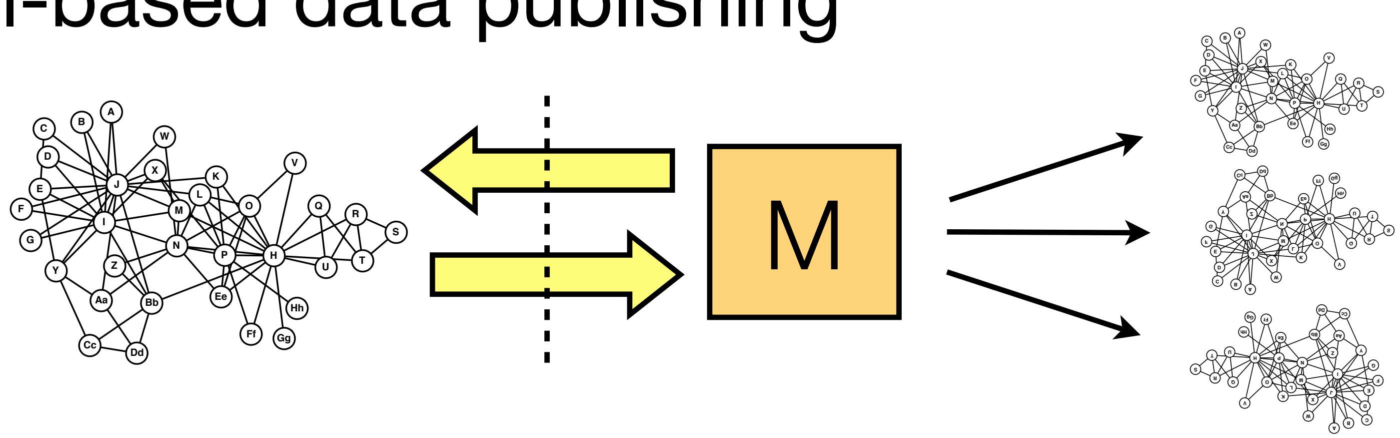
Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

- Output perturbation



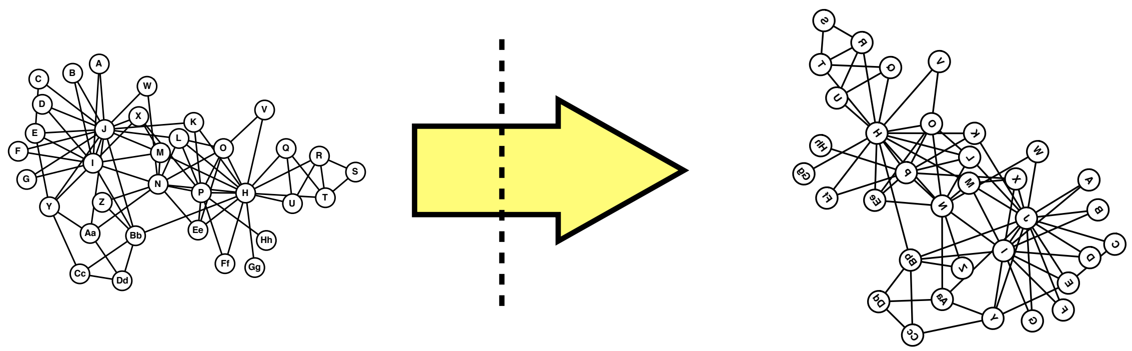
Ease of use	bad for practical
Privacy	formal guarantees
Accuracy	provable bounds
Scalability	very good

- Model-based data publishing



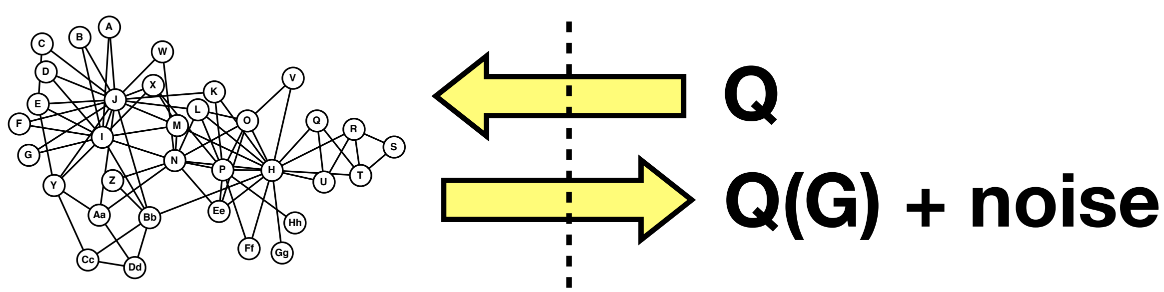
Data publishing v. output perturbation

- Data publishing



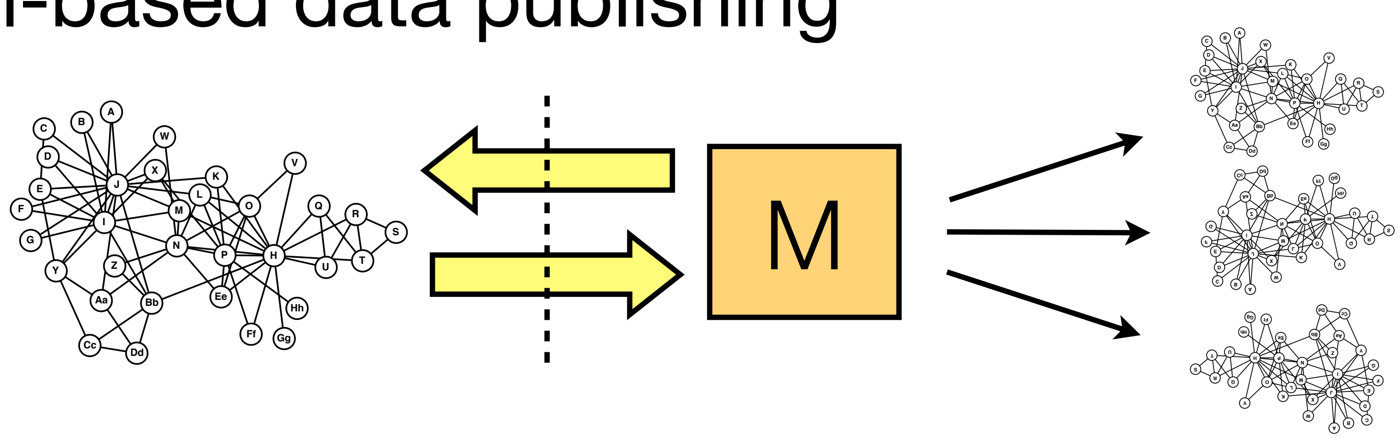
Ease of use	good
Privacy	weak guarantees
Accuracy	no formal guarantees
Scalability	sometimes bad

- Output perturbation



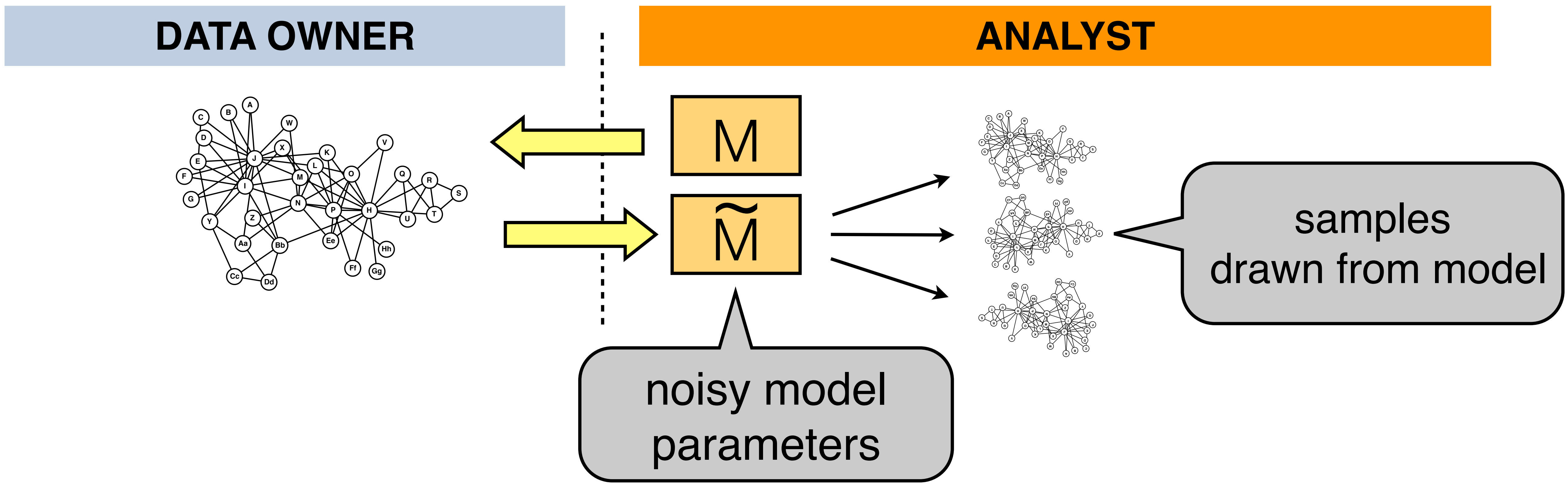
Ease of use	bad for practical
Privacy	formal guarantees
Accuracy	provable bounds
Scalability	very good

- Model-based data publishing



The best of both worlds ??

Toward differentially-private synthetic data



- To realize the benefits of synthetic data, data owner can release noisy parameters of network model.

- Baseline: the degree distribution as network model
  - Deriving the power law parameter — very accurate
  - Measuring clustering coefficient — not constrained by deg. distr.

A useful paradigm for improving accuracy

DATA OWNER | **ANALYST**

1. Formulate **S,**
having constraints Υ_S

$\mathcal{A}$

S ← **2.** Submit **S**

S(G) + noise → $\tilde{S}$ **3.** Perform **inference**

$\tilde{S}$, Υ_S → Inference → $\bar{S}$

See [Hay, PVLDB 10]

Questions?

Additional details on our work may be found here:

- **[Hay, PVLDB 10]** M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially-private queries through consistency. To appear, Proceedings of the VLDB Endowment (PVLDB), 2010.
- **[Hay, ICDM 09]** M. Hay, C. Li, G. Miklau, and D. Jensen. Accurate estimation of the degree distribution of private networks. In International Conference on Data Mining (ICDM) 2009.
- **[Rastogi, PODS 09]** V. Rastogi, M. Hay, G. Miklau, and D. Suciu. Relationship privacy: Output perturbation for queries with joins. In Principles of Database Systems (PODS), 2009.
- **[Hay, VLDB 08]** M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural identification in anonymized social networks. In Proceedings of the VLDB Endowment (PVLDB), 2008.

References

- **[Backstrom, WWW 07]** L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou R3579X? Anonymized social networks hidden patterns and structural steganography. In WWW, 2007.
- **[Liu, SIGMOD 08]** K. Liu and E. Terzi. Towards identity anonymization on graphs. In SIGMOD, 2008.
- **[Zhou, ICDE 08]** B. Zhou and J. Pei. Preserving privacy in social networks against neighborhood attacks. In ICDE, 2008.
- **[Zou, VLDB 09]** L. Zou, L. Chen, and T. Ozsu. K-automorphism: A general framework for privacy preserving network publication. In Proceedings of VLDB Conference, 2009.
- **[Ying, SDM 2008]** X. Ying and X. Wu. Randomizing social networks: a spectrum preserving approach. In SIAM International Conference on Data Mining, 2008.
- **[Cormode, VLDB 08]** G. Cormode, D. Srivastava, T. Yu, and Q. Zhang. Anonymizing bipartite graph data using safe groupings. In VLDB Conference, 2008.

References (con't)

- **[Cormode, VLDB 09]** G. Cormode, D. Srivastava, S. Bhagat, and B. Krishnamurthy. Class-based graph anonymization for social network data. In VLDB Conference, 2009.
- **[Narayanan, OAKL 09]** A. Narayanan and V. Shmatikov. De-anonymizing social networks. In Security and Privacy, 2009.
- **[Campan, PinKDD 08]** A. Campan and T. M. Truta. A clustering approach for data and structural anonymity in social networks. In PinKDD, 2008.
- **[Rastogi, VLDB 07]** V. Rastogi, S. Hong, and D. Suciu. The boundary between privacy and utility in data publishing. In VLDB, pages 531–542, 2007.
- **[Dwork, TCC 06]** C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In Third Theory of Cryptography Conference, 2006.
- **[Nissim, STOC 07]** K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In STOC, pages 75–84, 2007.