Challenges in Privacy-Preserving Analysis of Structured Data

Kamalika Chaudhuri

Computer Science and Engineering

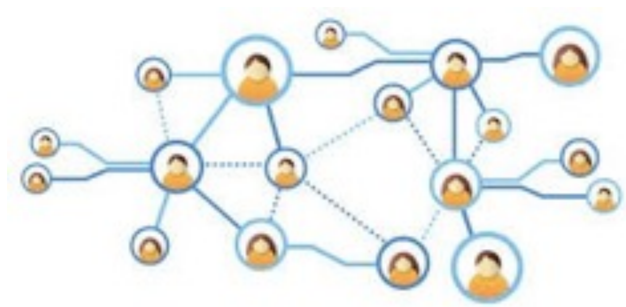
University of California, San Diego

Sensitive Structured Data

Medical Records

Search Logs

Social Networks

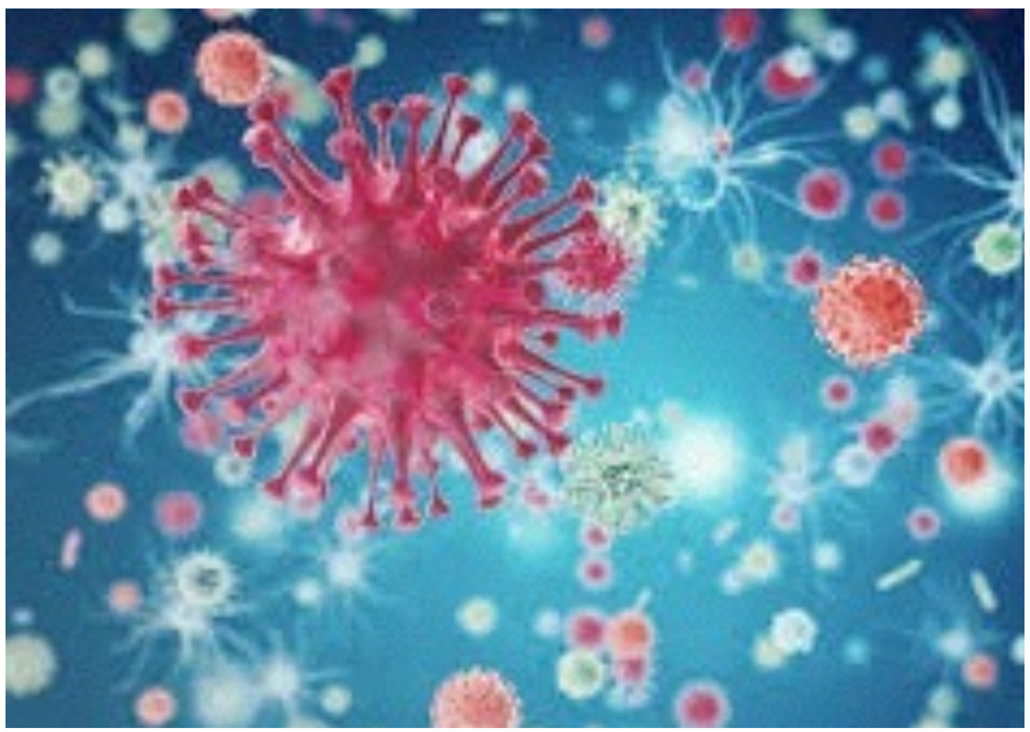


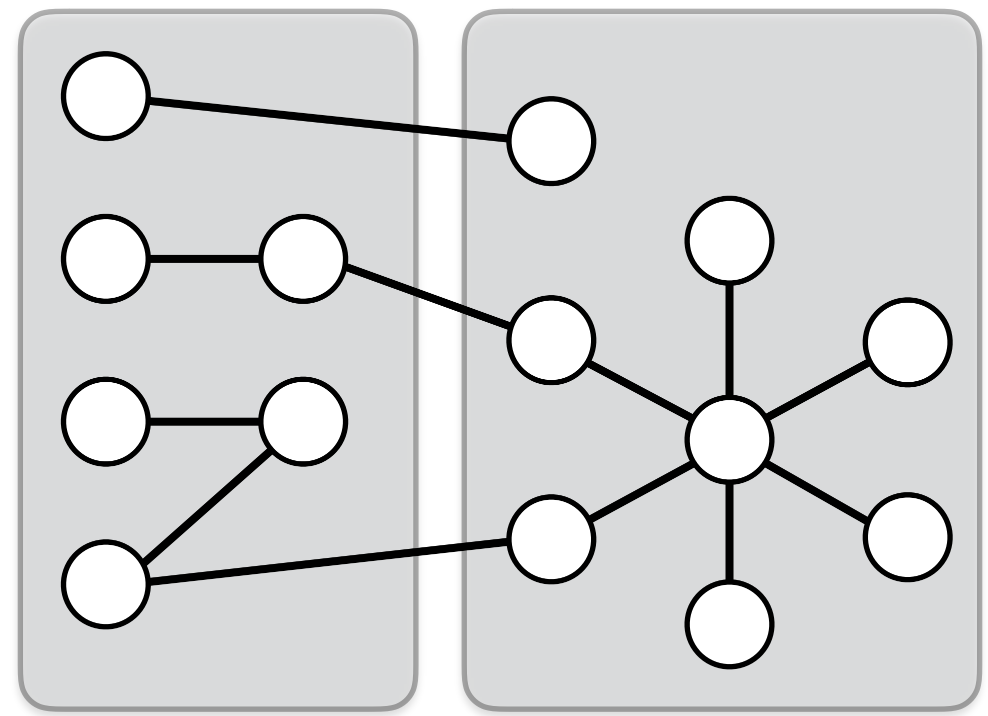
This Talk: Two Case Studies

1. Privacy-preserving HIV Epidemiology

2. Privacy in Time-series data

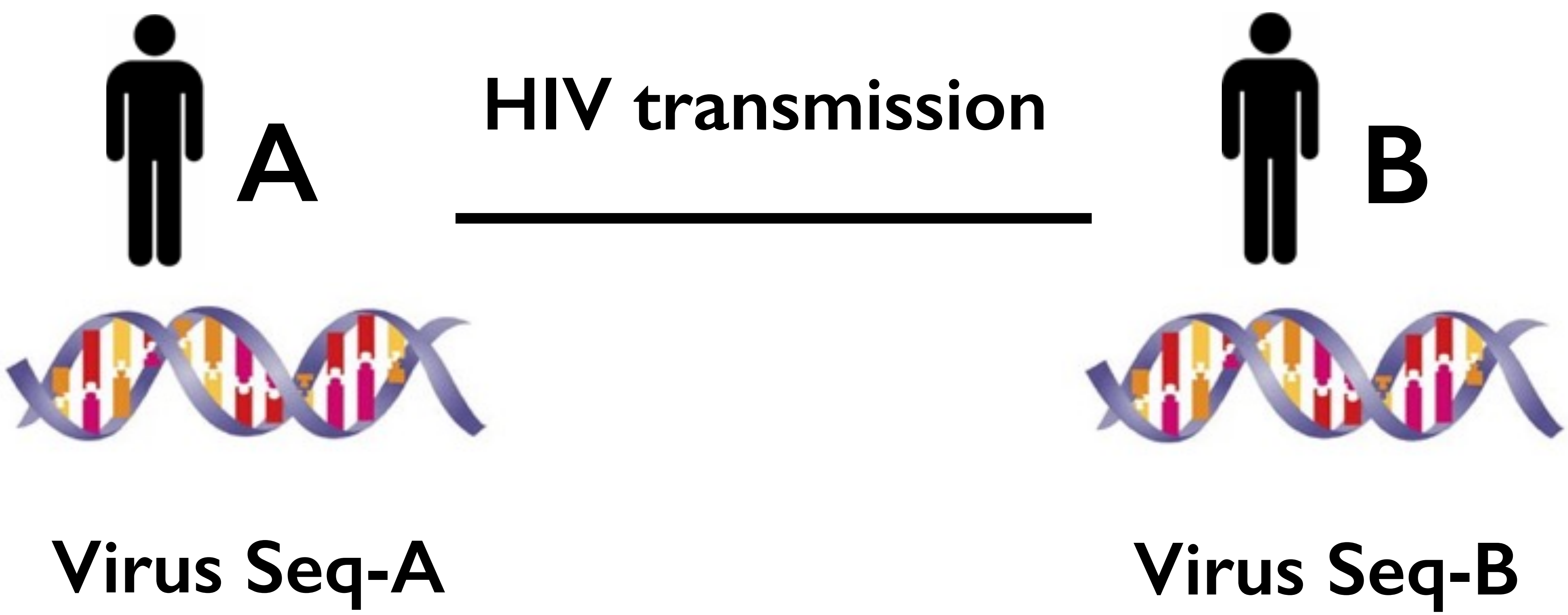
HIV Epidemiology





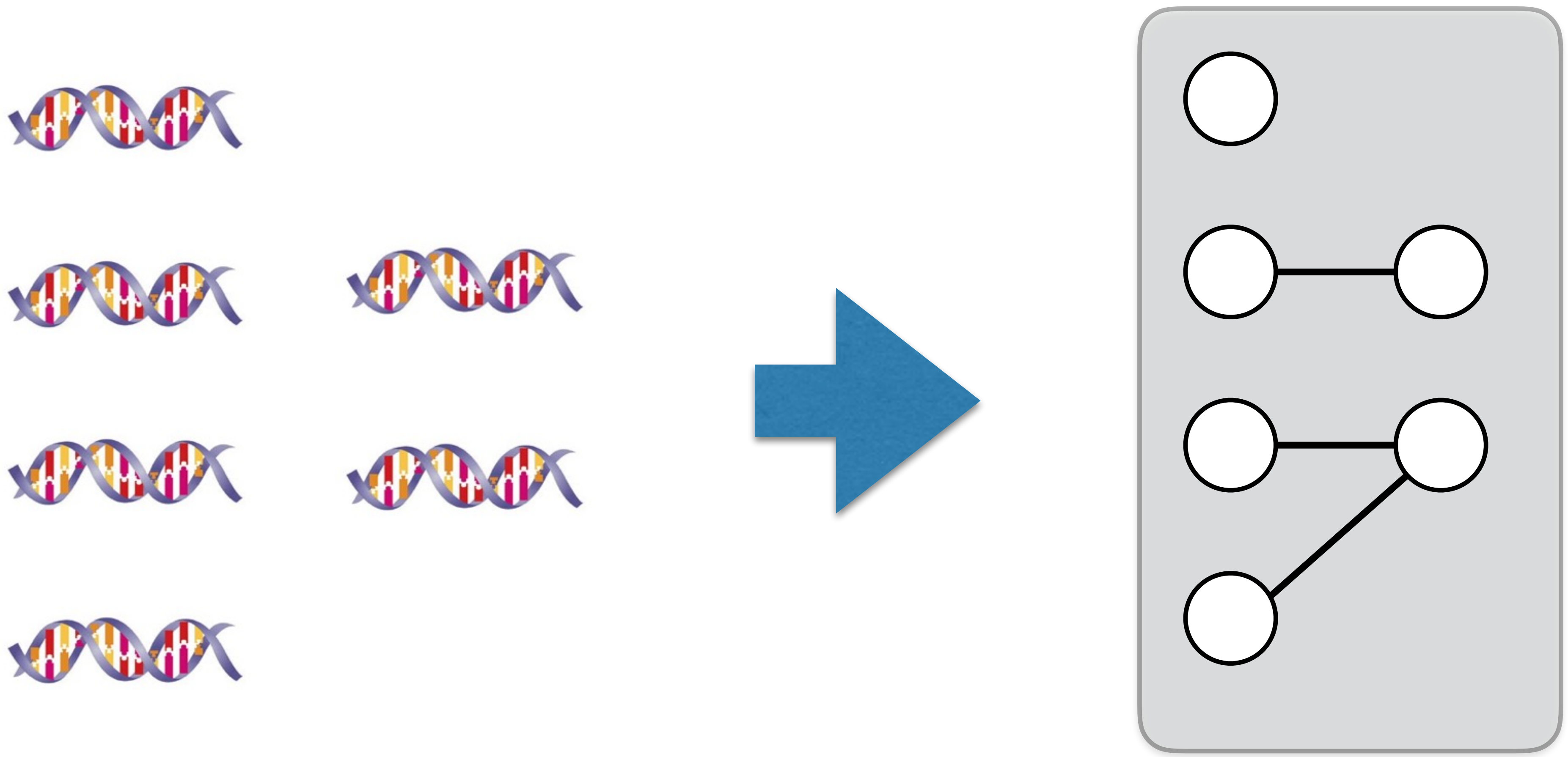
Goal: Understand how HIV spreads among people

HIV Transmission Data



distance (Seq-A, Seq-B) $< t$

From Sequences to Transmission Graphs

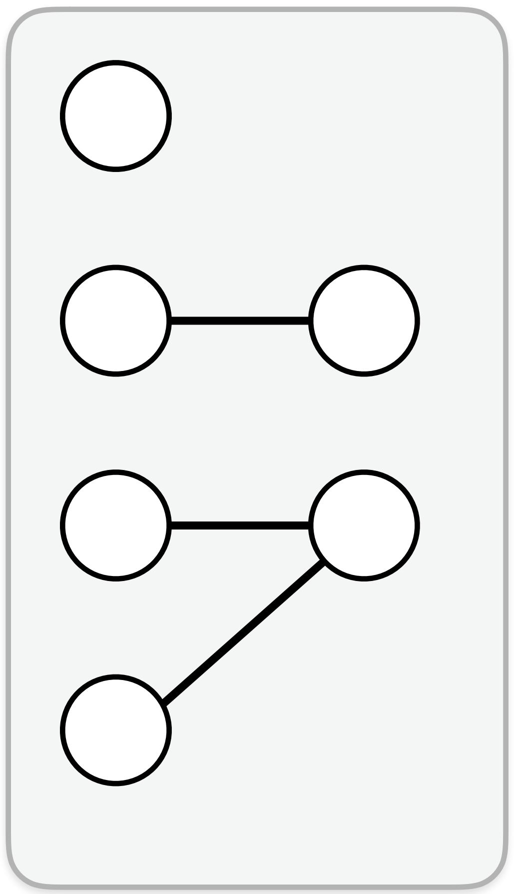


Viral Sequences

Node = Patient

Edge = Plausible transmission

…Growing over Time

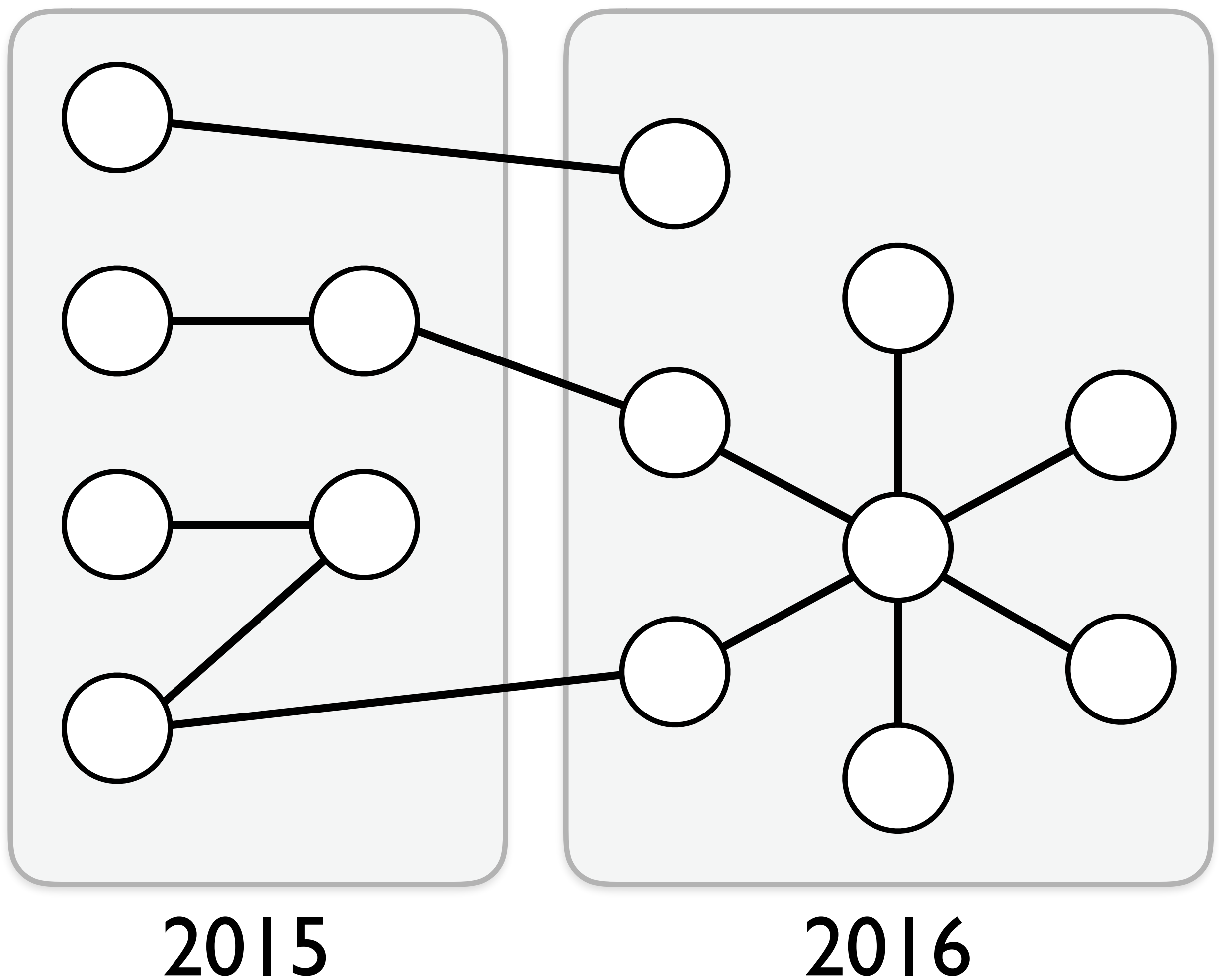


2015

Node = Patient

Edge = Transmission

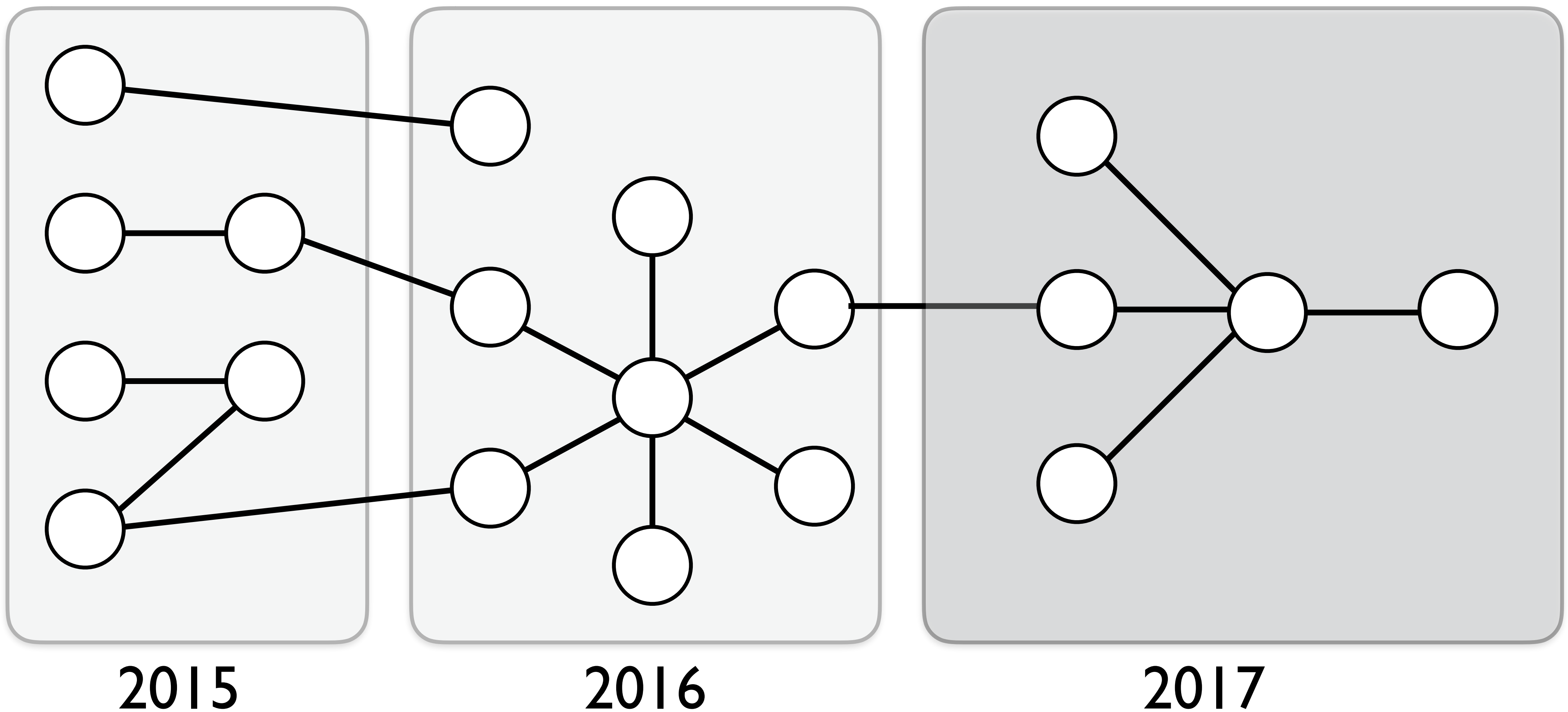
…Growing over Time



Node = Patient

Edge = Transmission

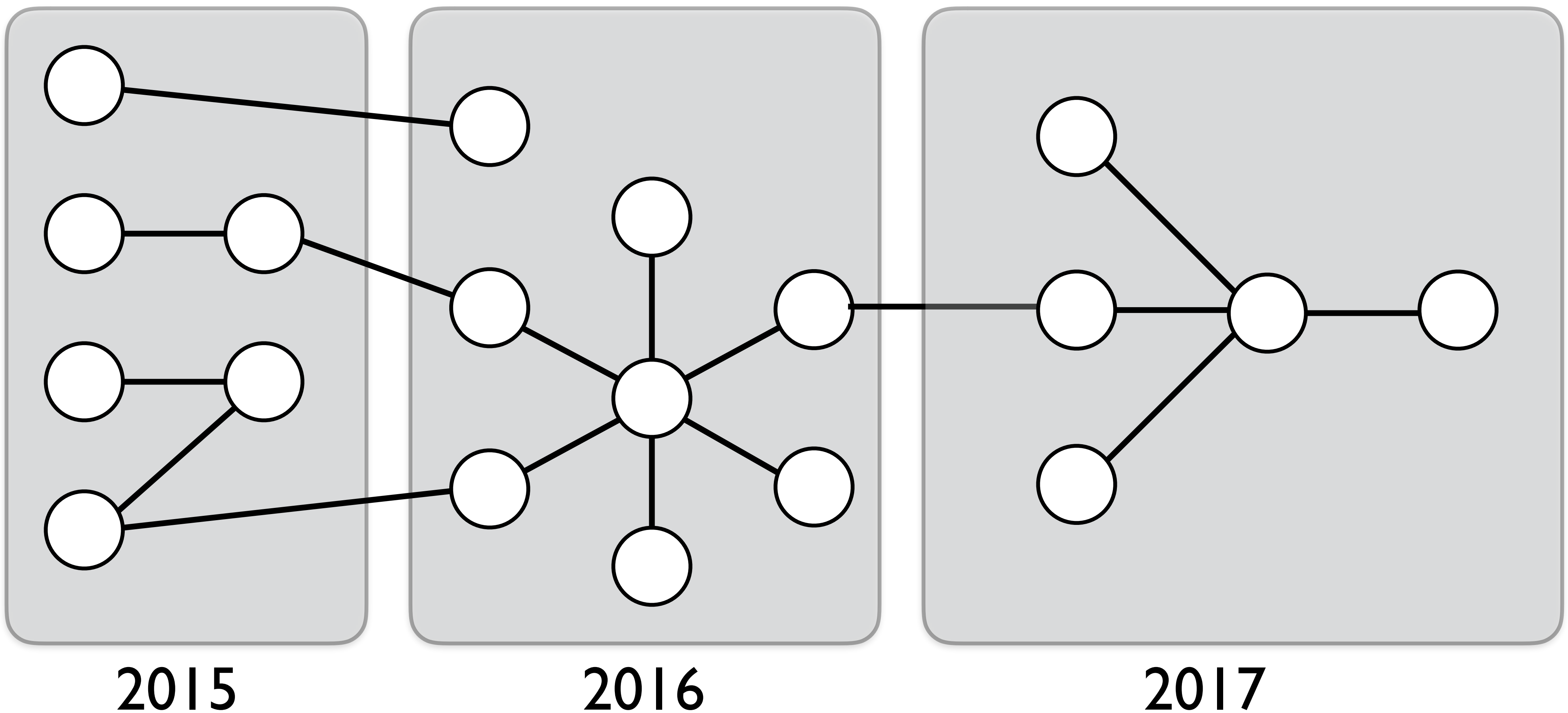
…Growing over Time



Node = Patient

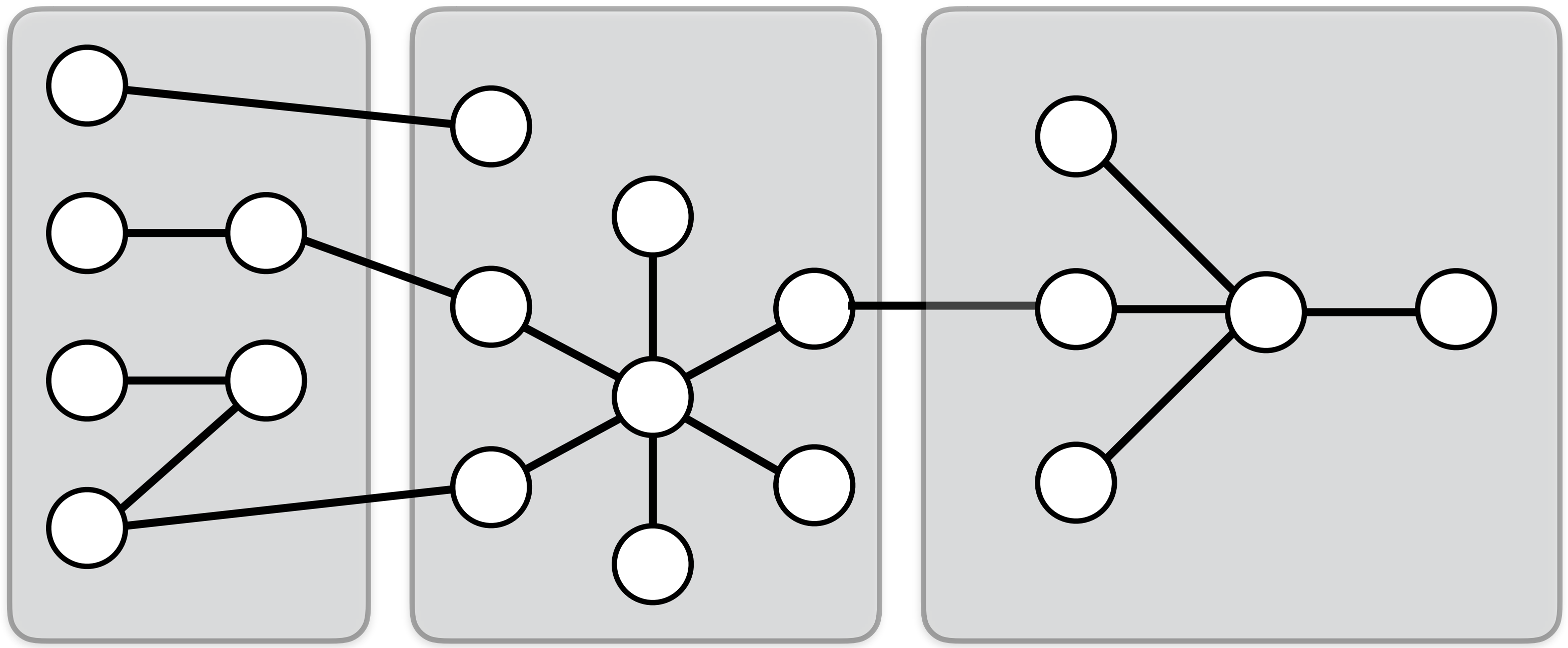
Edge = Transmission

…Growing over Time



Goal: Release properties of G with privacy across time

Problem: Continual Graph Statistics Release



Given: (Growing) graph G

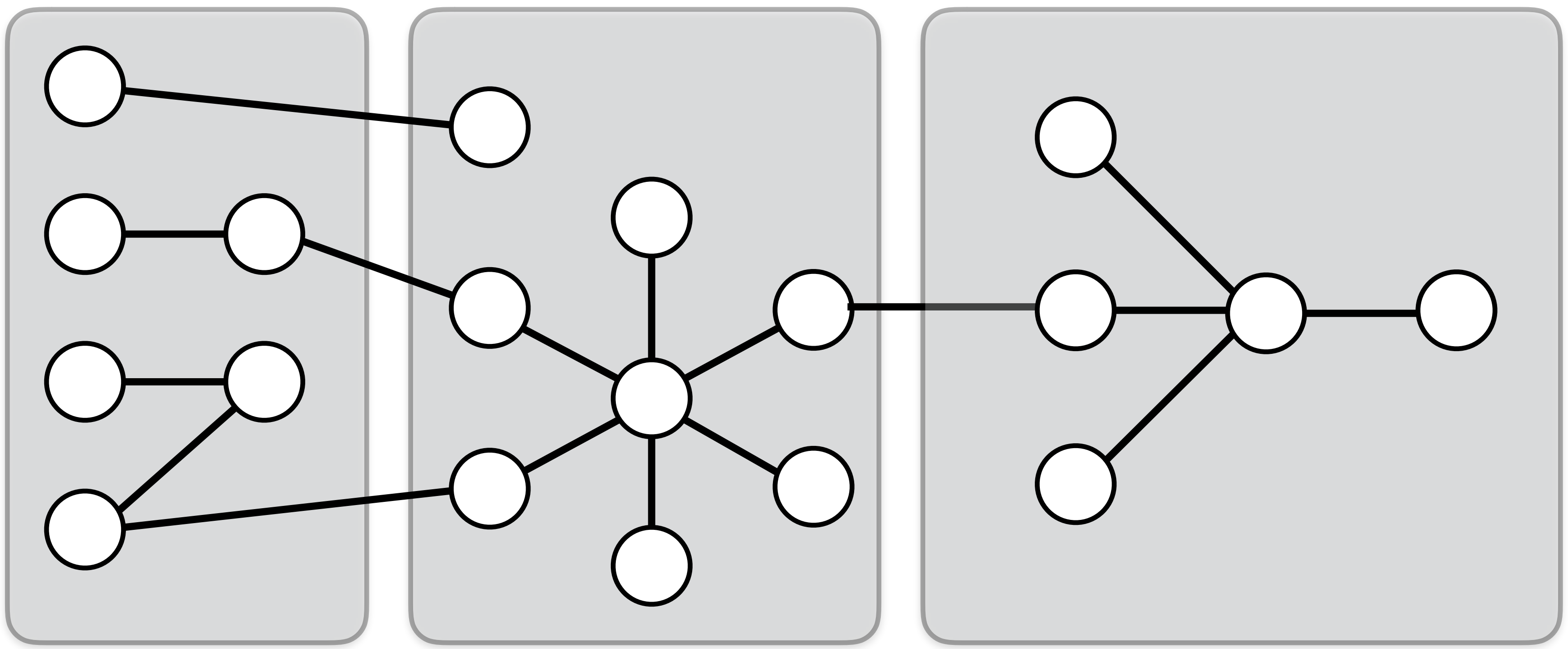
At time t, nodes and adjacent edges $(\partial V_t, \partial E_t)$ arrive

Goal: At time t, release f(G_t), where f = graph statistic, and

$$G_t = (\cup_{s \leq t} \partial V_s, \cup_{s \leq t} \partial E_s)$$

while preserving patient **privacy** and **high accuracy**

What kind of Privacy?



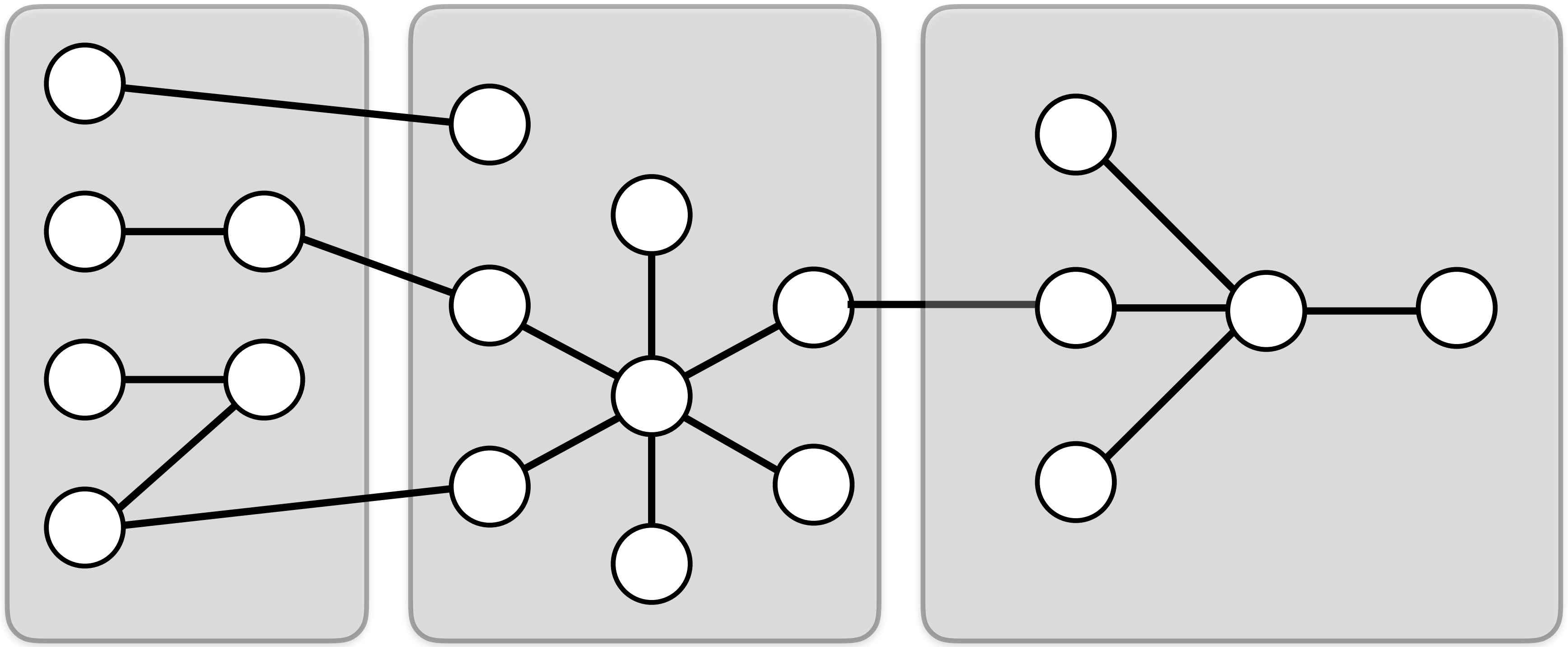
Node = Patient

Edge = Transmission

Hide: Patient A is in the graph

Release: Large scale properties

What kind of Privacy?



Node = Patient

Edge = Transmission

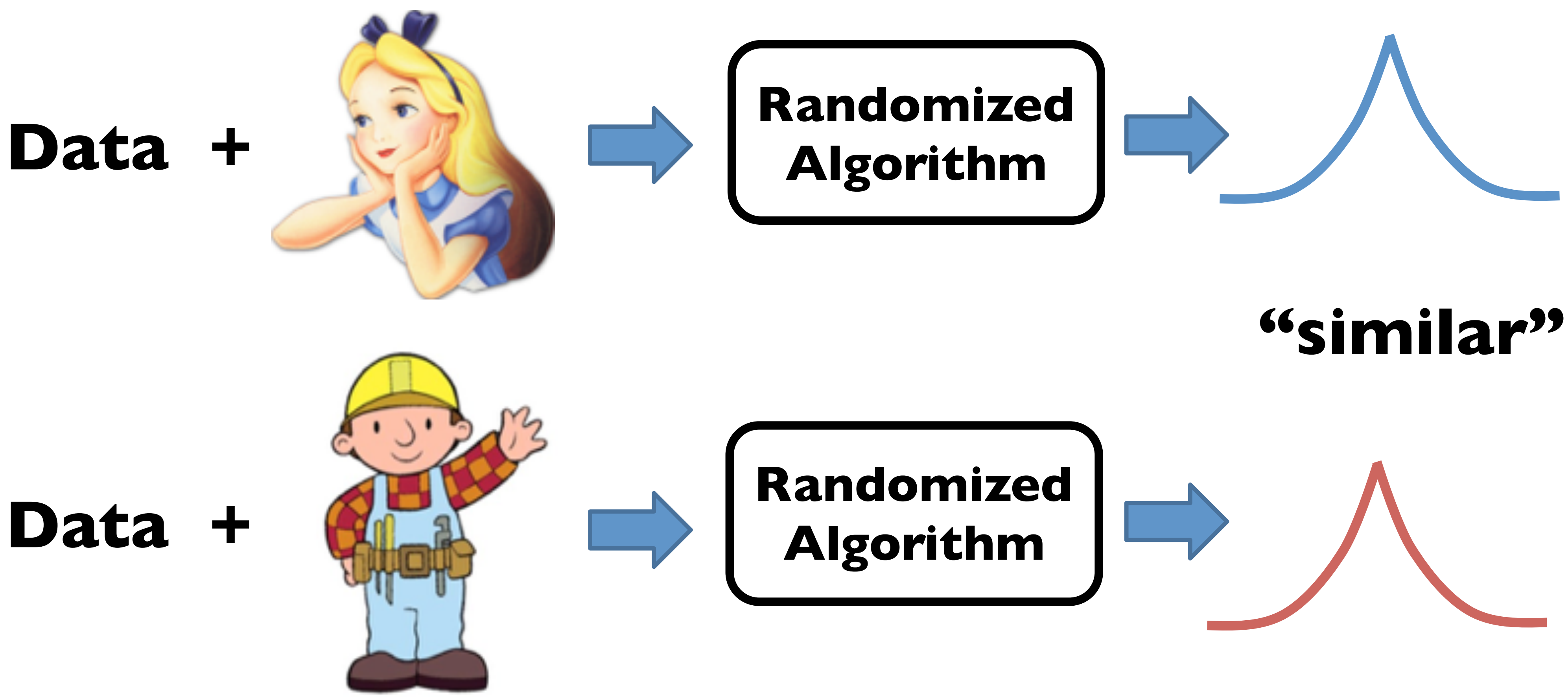
Hide: A particular patient has HIV

Privacy notion: Node Differential Privacy

Talk Outline

- The Problem: Private HIV Epidemiology
- Privacy Definition: Differential Privacy

Differential Privacy [DMNS06]



Participation of a single person does not change output

Differential Privacy: Attacker's View

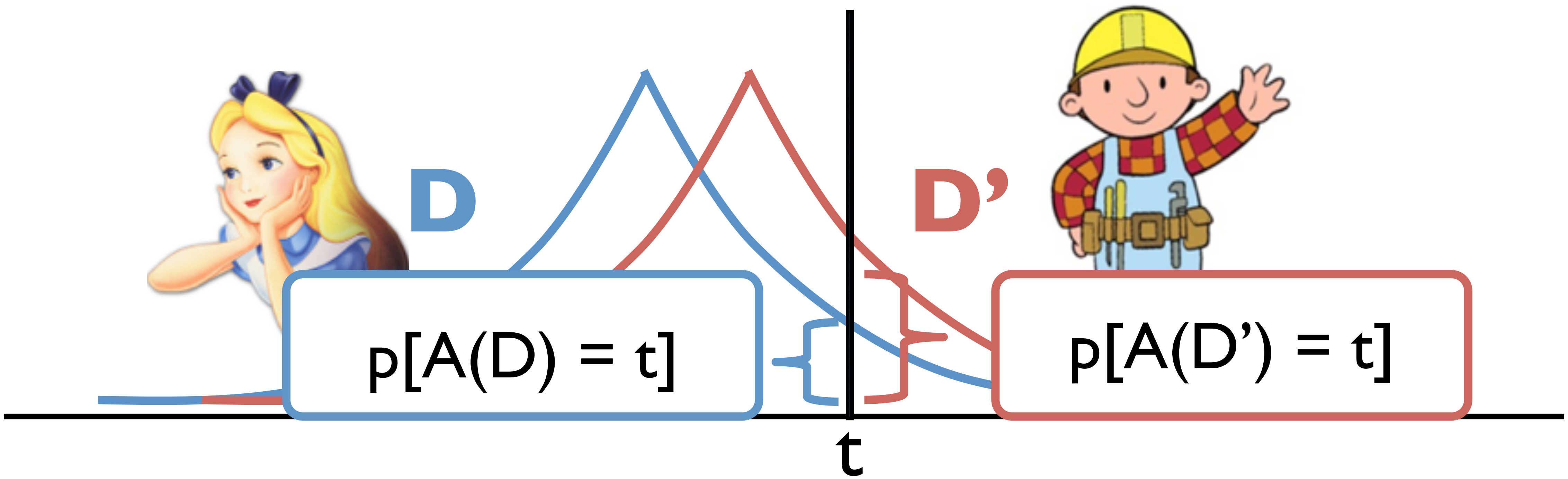
Prior Knowledge + **Algorithm** Output on Data &  = **Conclusion** on 

Prior Knowledge + **Algorithm** Output on Data &  = **Conclusion** on 

Note: a. Algorithm could draw **personal conclusions** about Alice

b. Alice has the **agency** to participate or not

Differential Privacy [DMNS06]



For all D, D’ that differ in one person’s value,

If A = ϵ-differentially private randomized algorithm, then:

$$\sup_t \left| \log \frac{p(A(D) = t)}{p(A(D') = t)} \right| \leq \epsilon$$

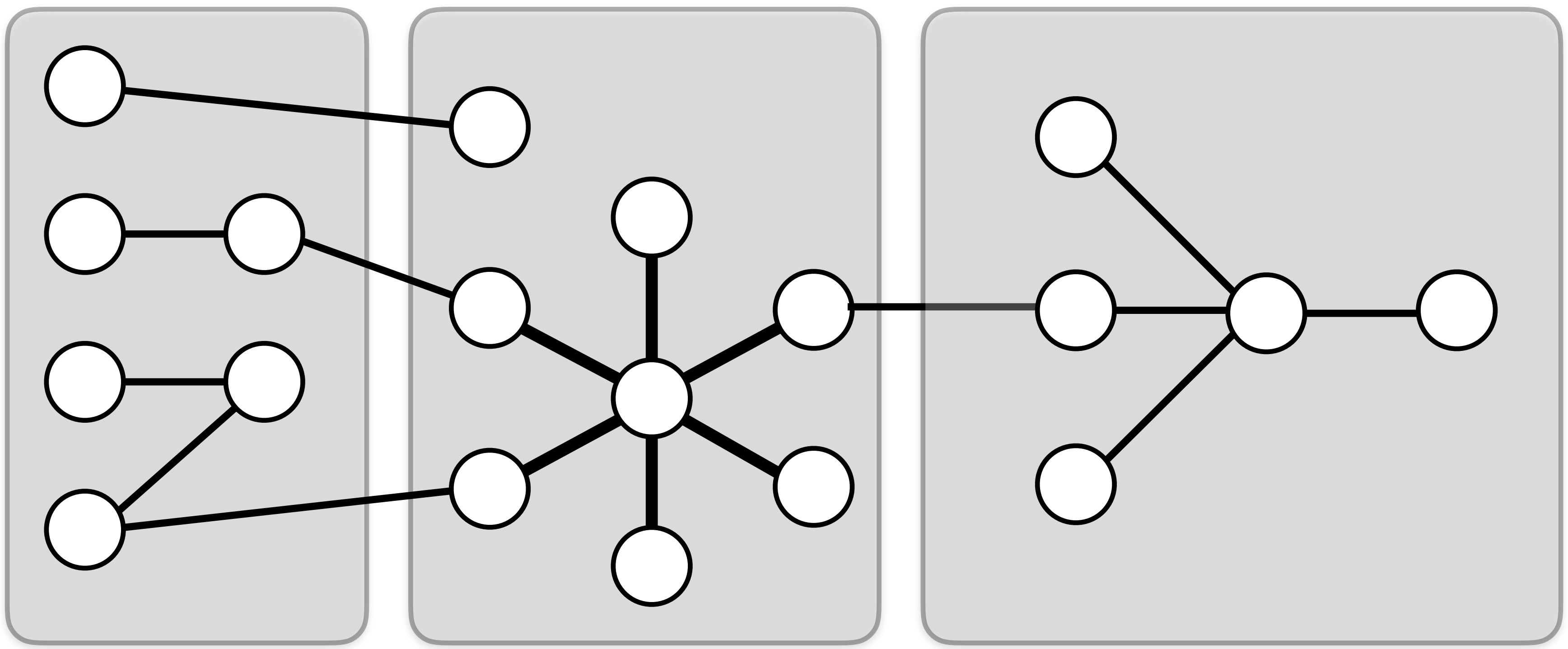
Differential Privacy

1. Provably strong notion of privacy

2. Good approximations for many functions

   e.g, means, histograms, etc.

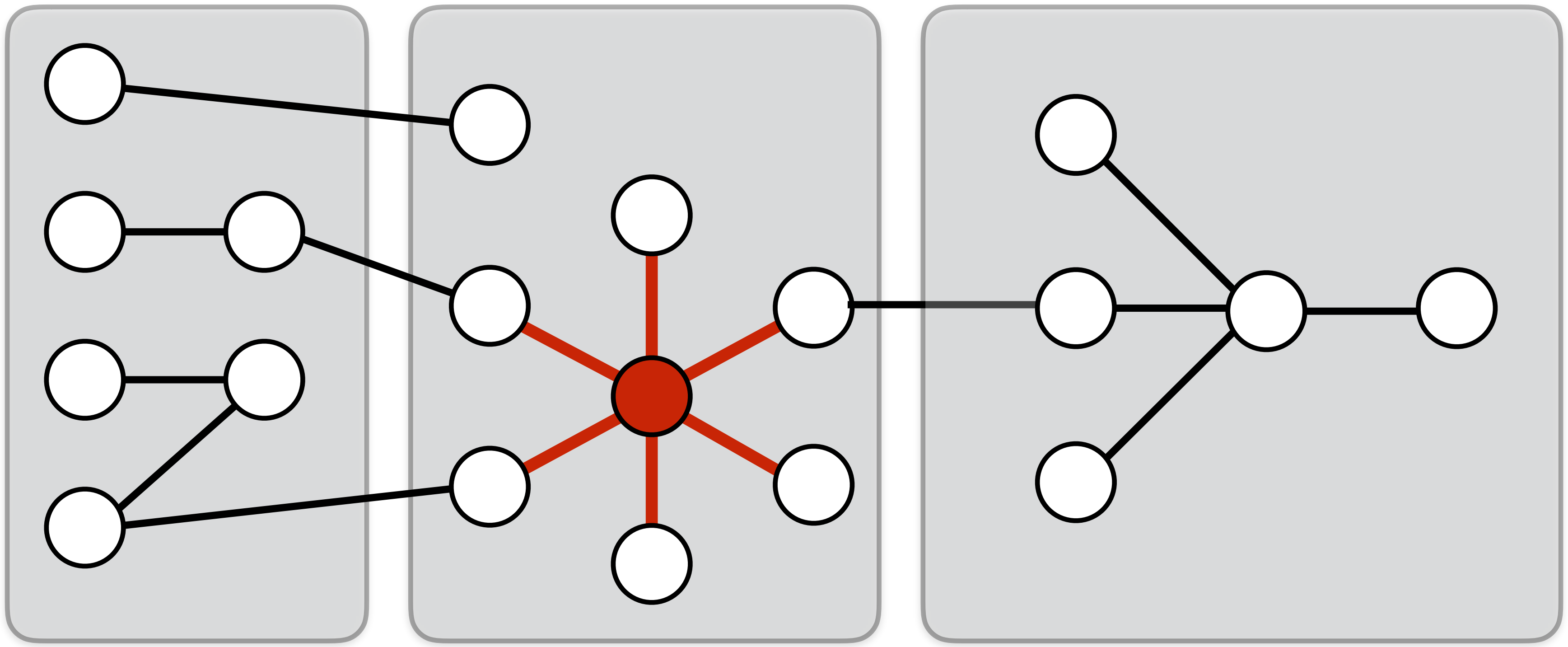
Node Differential Privacy



Node = Patient

Edge = Transmission

Node Differential Privacy



Node = Patient

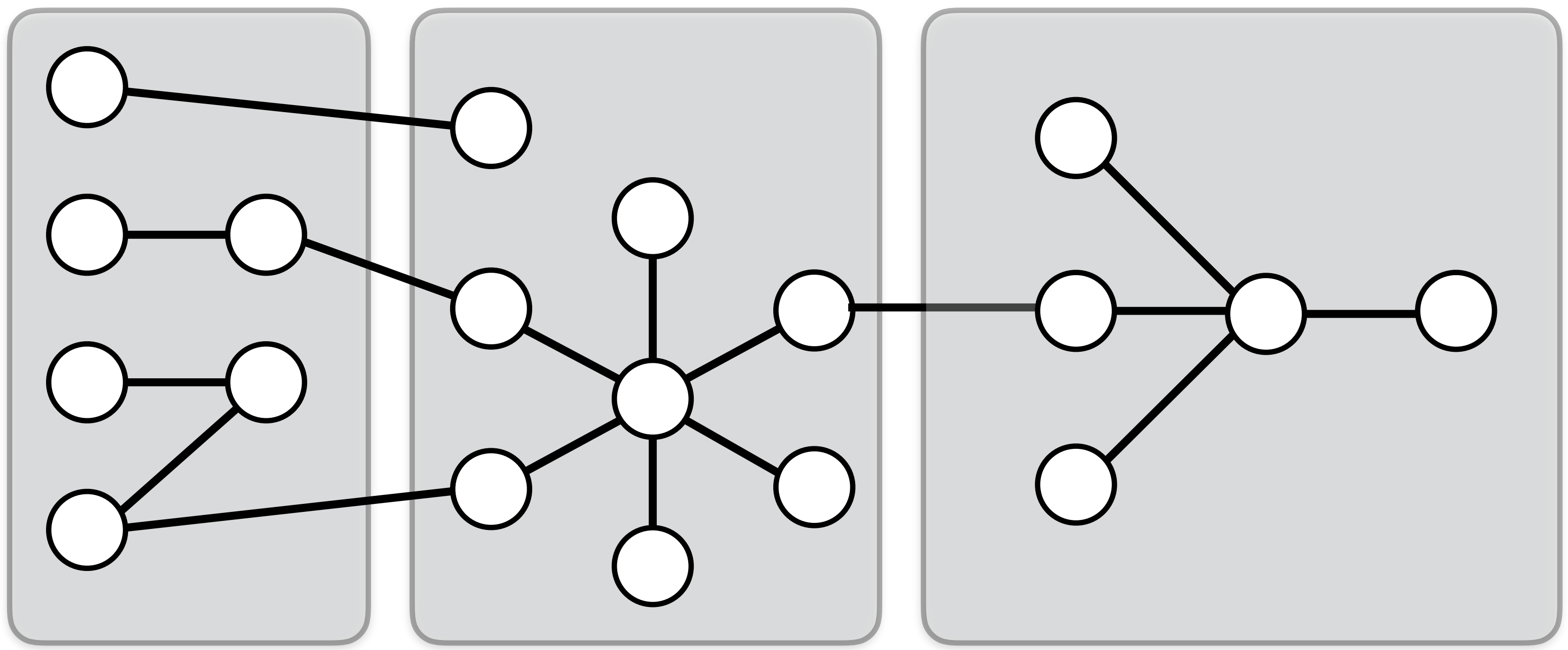
Edge = Transmission

One person's value = One node + adjacent edges

Talk Outline

- The Problem: Private HIV Epidemiology
- Privacy Definition: Node Differential Privacy
- Challenges

Problem: Continual Graph Statistics Release



Given: (Growing) graph G

At time t, nodes and adjacent edges $(\partial V_t, \partial E_t)$ arrive

Goal: At time t, release f(G_t), where f = graph statistic, and

$$G_t = (\cup_{s \leq t} \partial V_s, \cup_{s \leq t} \partial E_s)$$

with node differential privacy and high accuracy

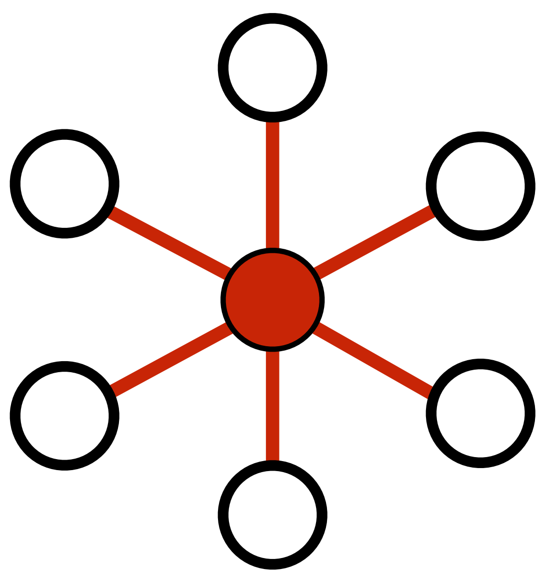
Why is Continual Release of Graphs with Node Differential Privacy hard?

1. Node DP challenging in static graphs [KNRS13, BBDS13]

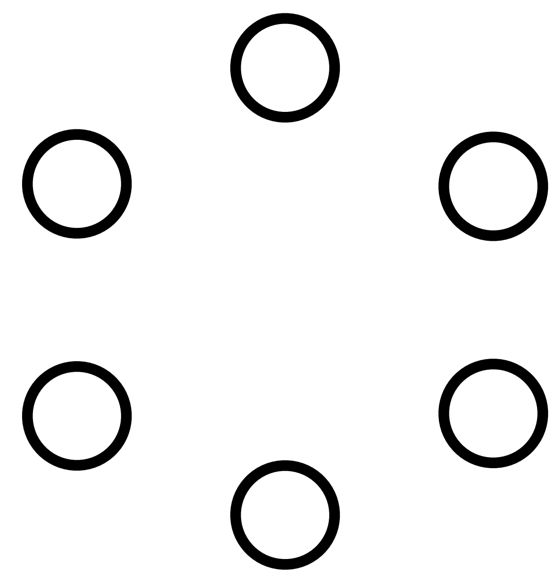
2. Continual release of graph data has extra challenges

Challenge 1: Node DP

Removing one node can change properties by a lot (even for static graphs)



#edges = 6 (size of V)



#edges = 0

Hiding one node needs high noise → low accuracy

Prior Work: Node DP in Static Graphs

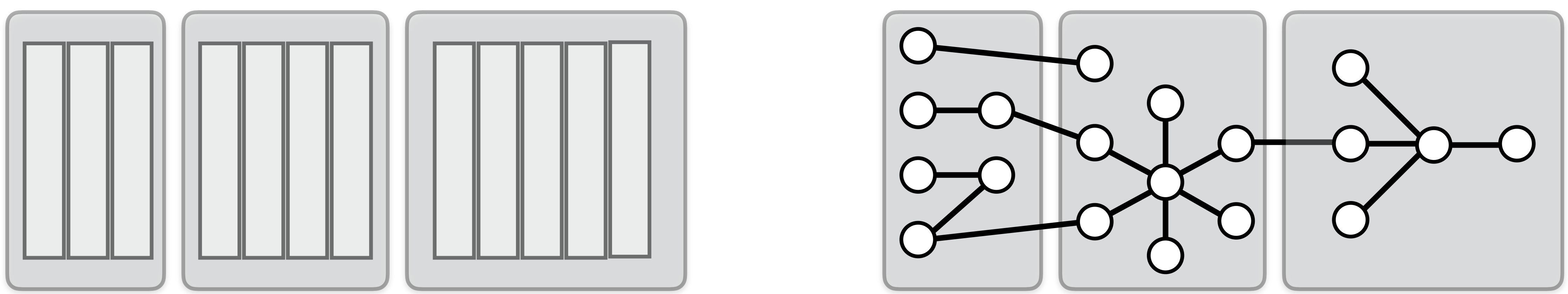
Approach 1 [BCS15]:

- Assume bounded max degree

Approach 2 [KNRS13, RS15]:

- Project to low degree graph G’ and use node DP on G’
- Projection algorithm needs to be “smooth” and computationally efficient

Challenge 2: Continual Release of Graphs

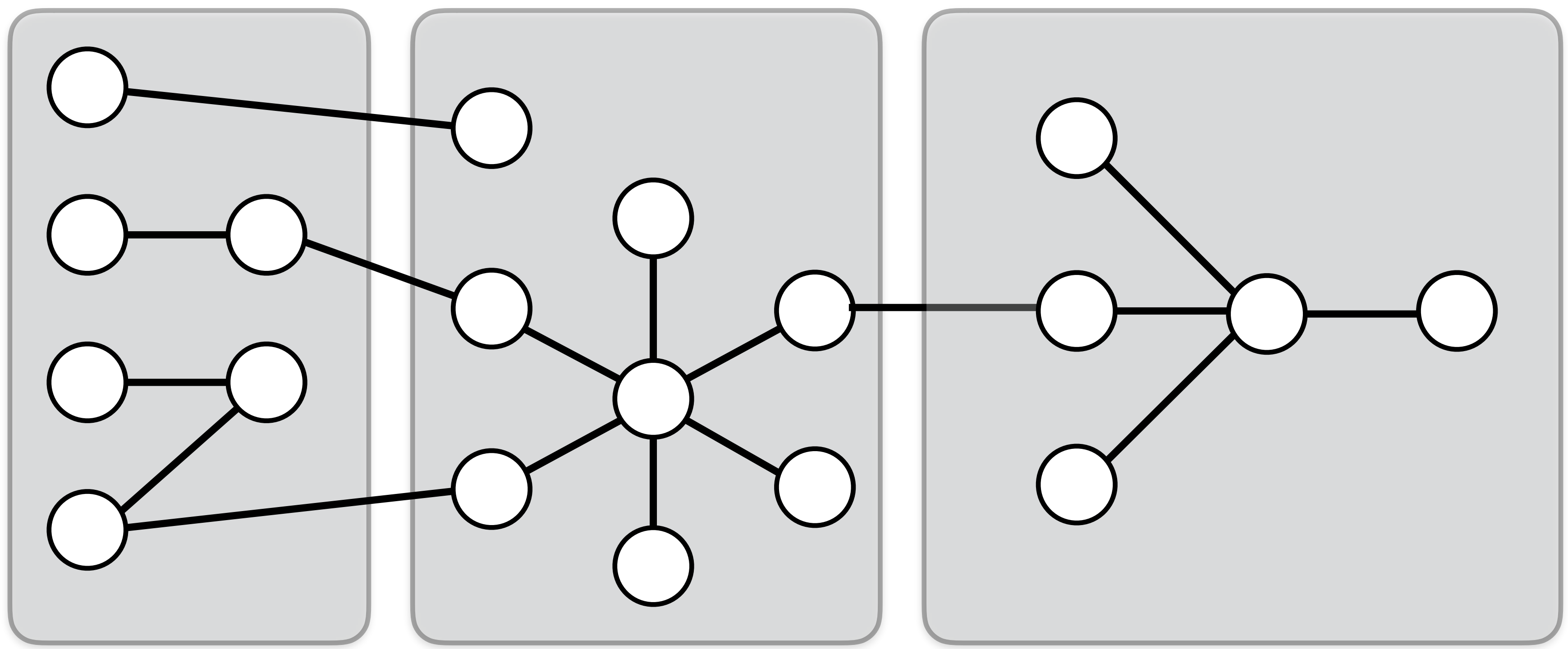


- Methods for tabular data [DNPR10, CSS10] do not apply
- Sequential composition gives poor utility
- Graph projection methods are not "smooth" over time

Talk Outline

- The Problem: Private HIV Epidemiology
- Privacy Definition: Node Differential Privacy
- Challenges
- Approach

Algorithm: Main Ideas

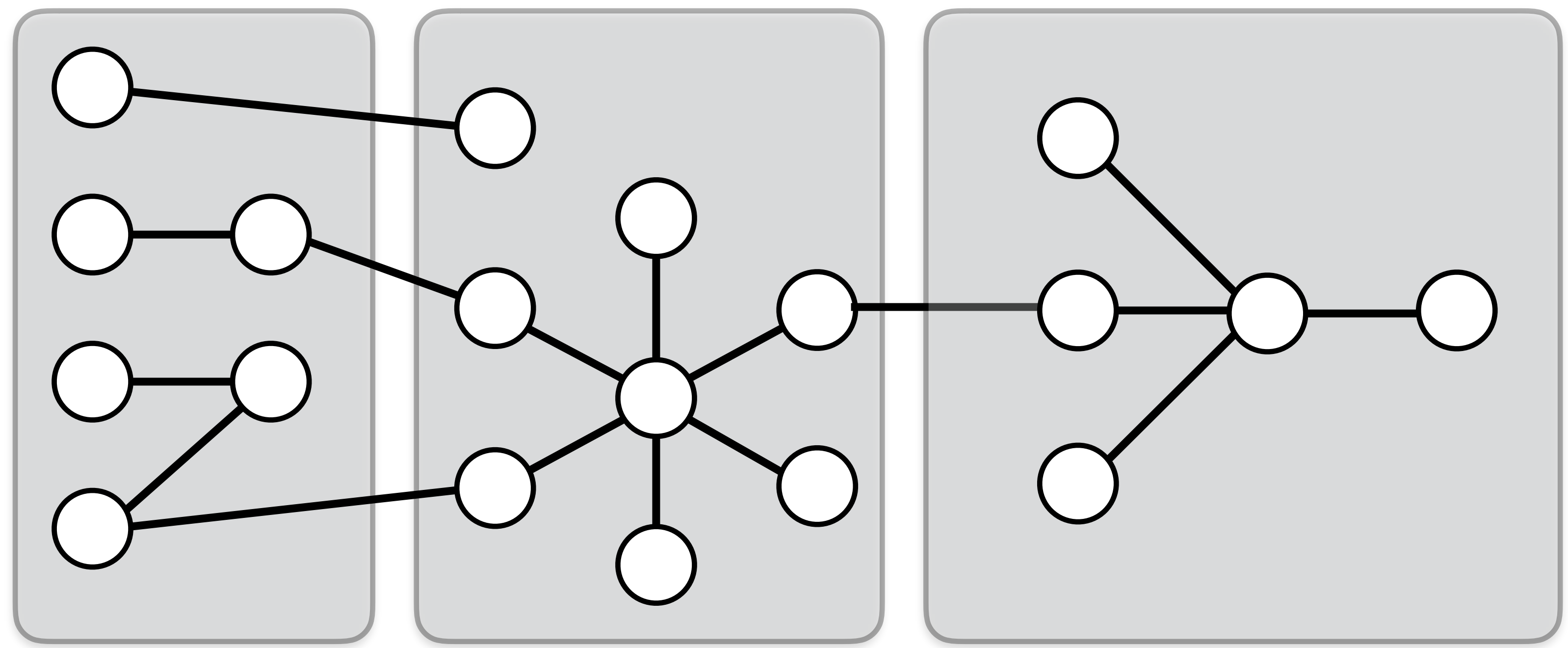


Strategy 1: Assume bounded max degree of G (from domain)

Strategy 2: Privately release "**difference sequence**" of statistic (instead of the direct statistic)

Difference Sequence

Graph Sequence:



G_1 G_2 G_3

Statistic Sequence:	$f(G_1)$	$f(G_2)$	$f(G_3)$
Difference Sequence:	$f(G_1)$	$f(G_2) - f(G_1)$	$f(G_3) - f(G_2)$

Key Observation

Key Observation: For many graph statistics, when G is degree bounded, the **difference sequence** has low sensitivity

Example Theorem:
If max degree(G) = D, then sensitivity of the difference sequence for #high degree nodes is at most 2D + 1.

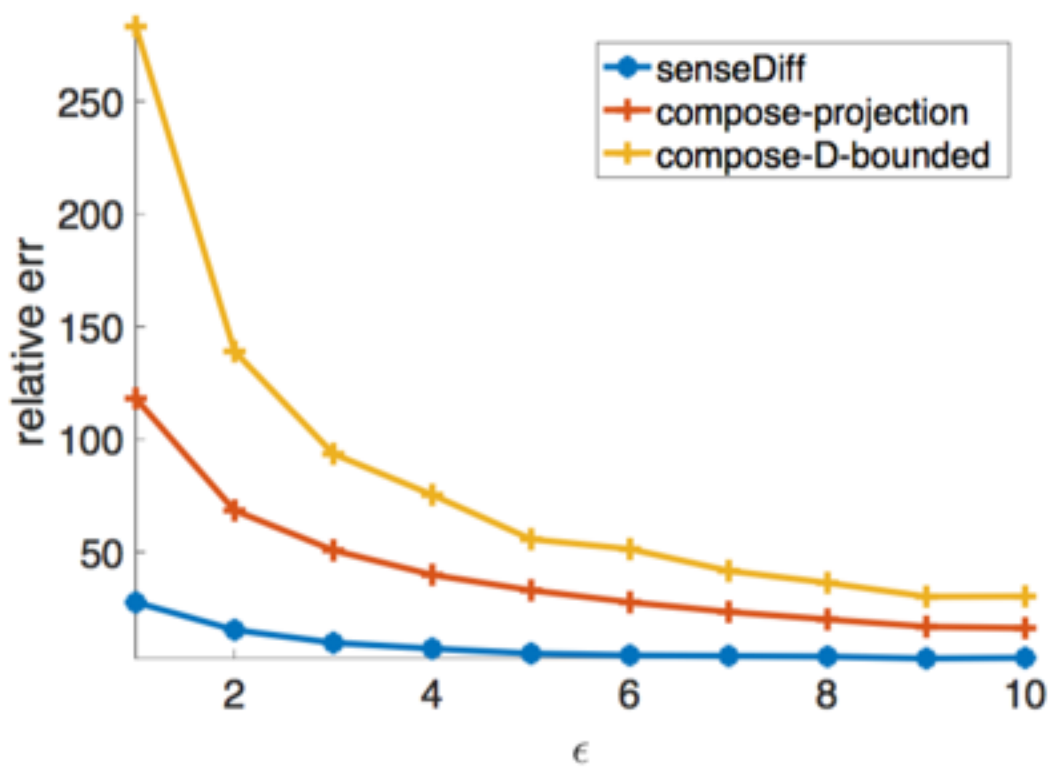
From Observation to Algorithm

Algorithm:

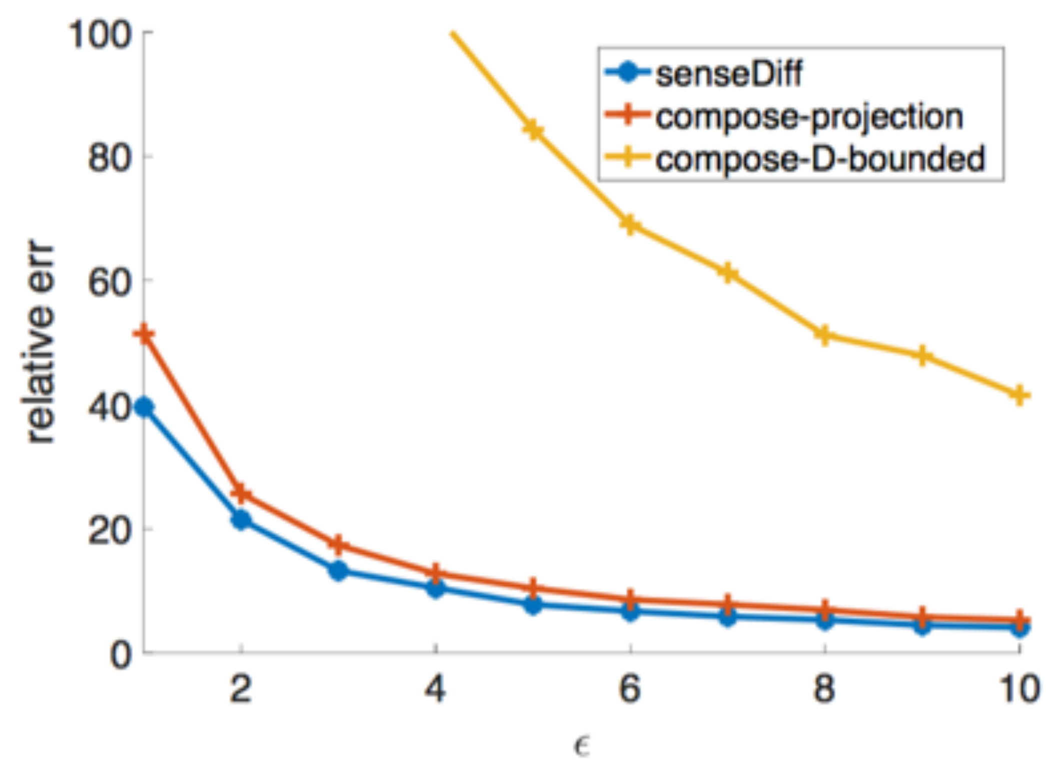
1. Add noise to each item of difference sequence to hide effect of single node and publish
2. Reconstruct private statistic sequence from private difference sequence

How does this work?

Experiments - Privacy vs. Utility



#edges

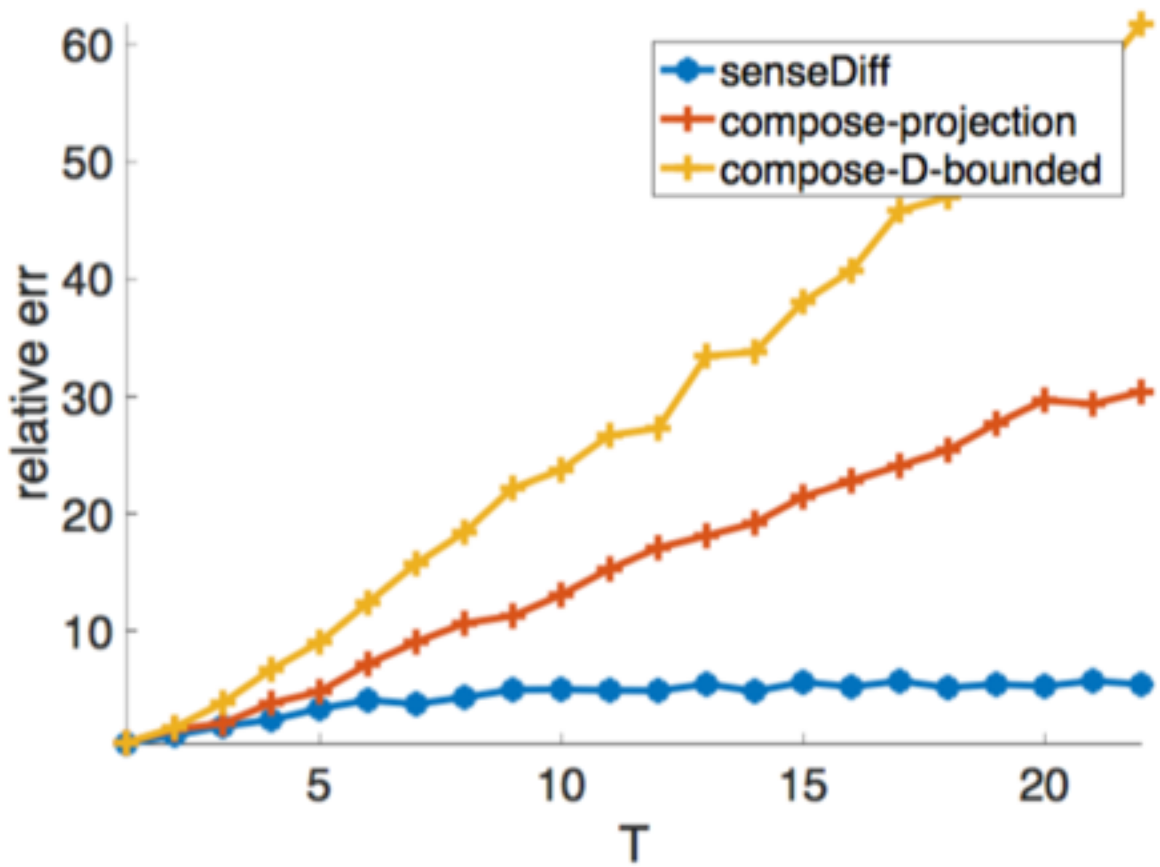


#high degree nodes

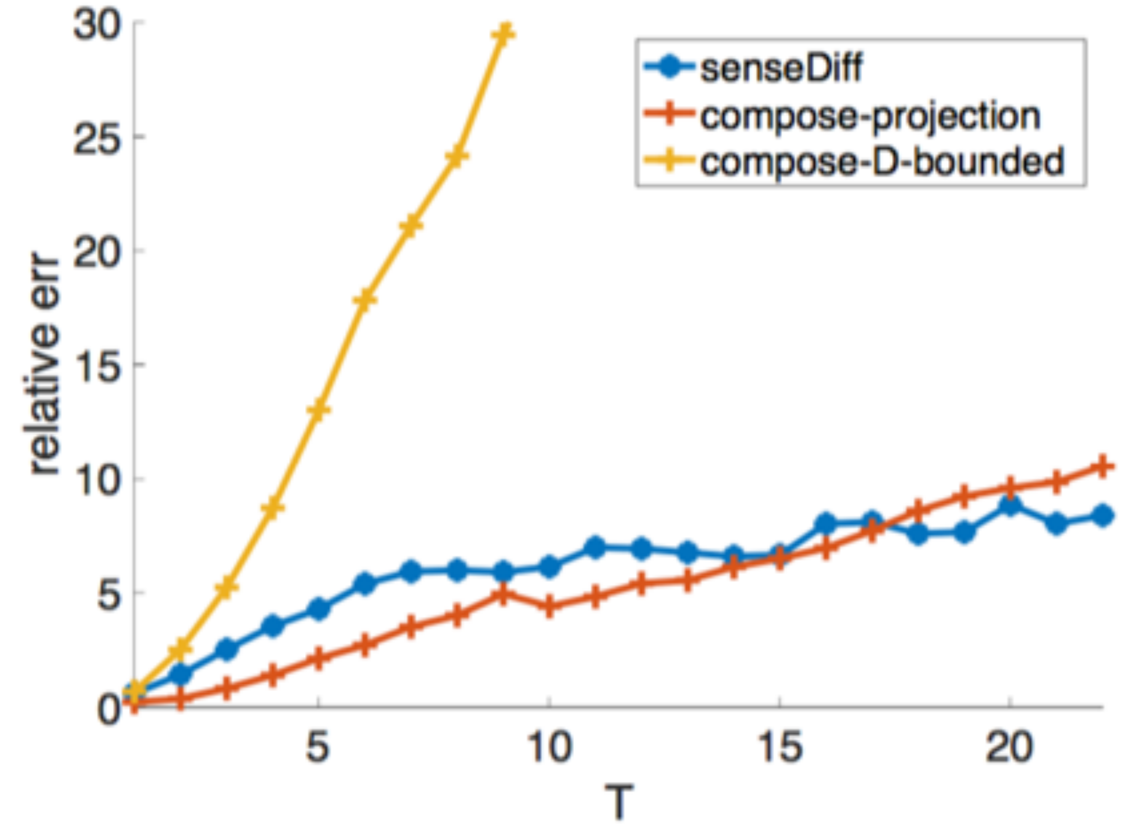
Baselines:

Our Algorithm, DP Composition 1, DP Composition 2

Experiments - #Releases vs. Utility



#edges



#high degree nodes

Baselines:

Our Algorithm, DP Composition 1, DP Composition 2

Talk Agenda

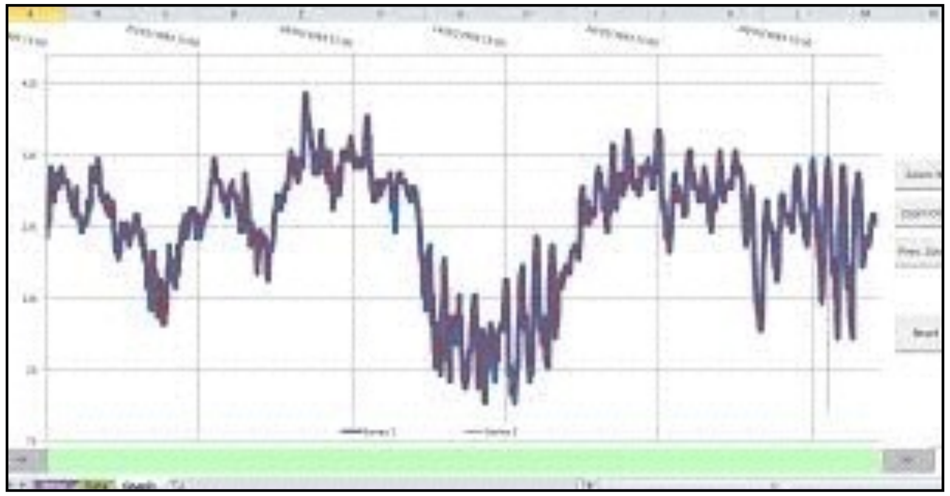
Privacy is application-dependent!

Two applications:

1. HIV Epidemiology
2. Privacy of time-series data - activity monitoring, power consumption, etc

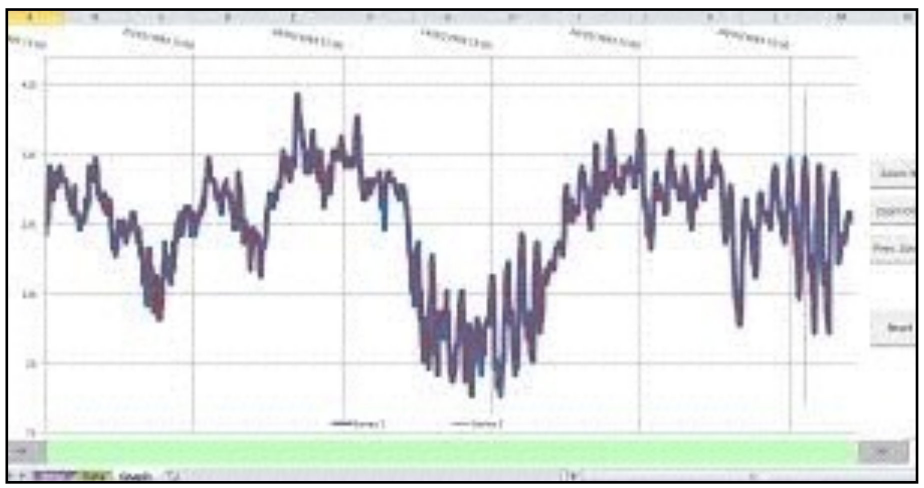
Time Series Data

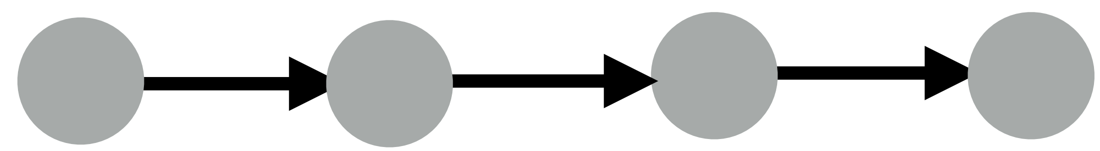
Physical Activity Monitoring



Location traces

Example: Activity Monitoring





Data: Activity trace of a subject

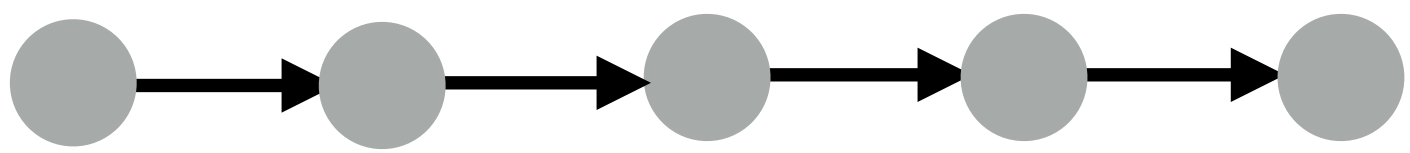
Hide: Activity at each time against adversary with prior knowledge

Release: (Approximate) aggregate activity

Why is Differential Privacy not Right for Correlated data?

Example: Activity Monitoring

$D = (x_1, .., x_T)$, x_t = activity at time t



Correlation Network

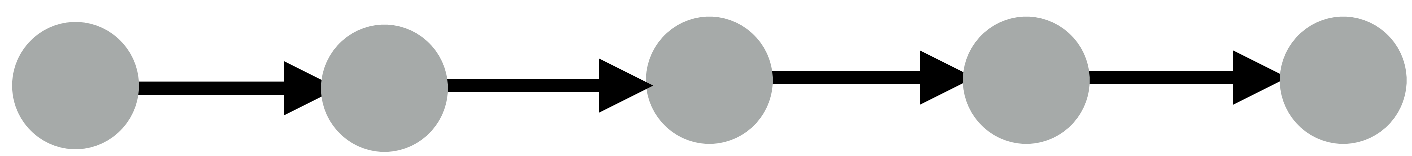
Data from a single subject

1-DP: Output histogram of activities + noise with stdev T

Too much noise - no utility!

Example: Activity Monitoring

$D = (x_1, .., x_T)$, x_t = activity at time t



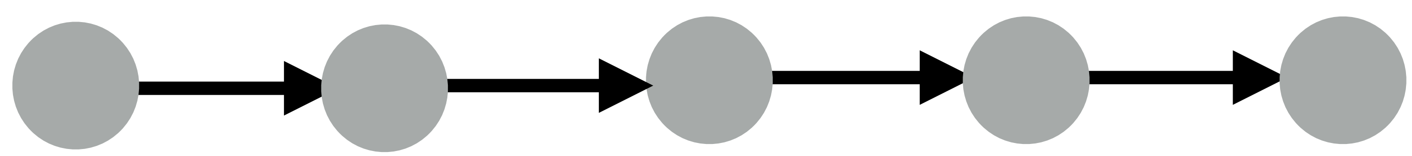
Correlation Network

1-entry-DP: Output activity histogram + noise with stdev 1

Not enough noise - activities across time are correlated!

Example: Activity Monitoring

$D = (x_1, .., x_T)$, x_t = activity at time t



Correlation Network

1-entry-group DP:

Output activity histogram + noise with stdev T

Too much noise - no utility!

How to define privacy for Correlated Data ?

Pufferfish Privacy [KM12]

Secret Set S

S: Information to be protected

e.g: Alice's age is 25, Bob has a disease

Pufferfish Privacy [KM12]

Secret Set S

Secret Pairs Set Q

Q: Pairs of secrets we want to be indistinguishable

e.g: (Alice's age is 25, Alice's age is 40)

(Bob is in dataset, Bob is not in dataset)

Pufferfish Privacy [KM12]

Secret Set S	Secret Pairs Set Q	Distribution Class Θ

Θ: A set of distributions that plausibly generate the data

e.g: (connection graph G, disease transmits w.p [0.1, 0.5])

(Markov Chain with transition matrix in set **P**)

May be used to model correlation in data

Pufferfish Privacy [KM12]

Secret Set S

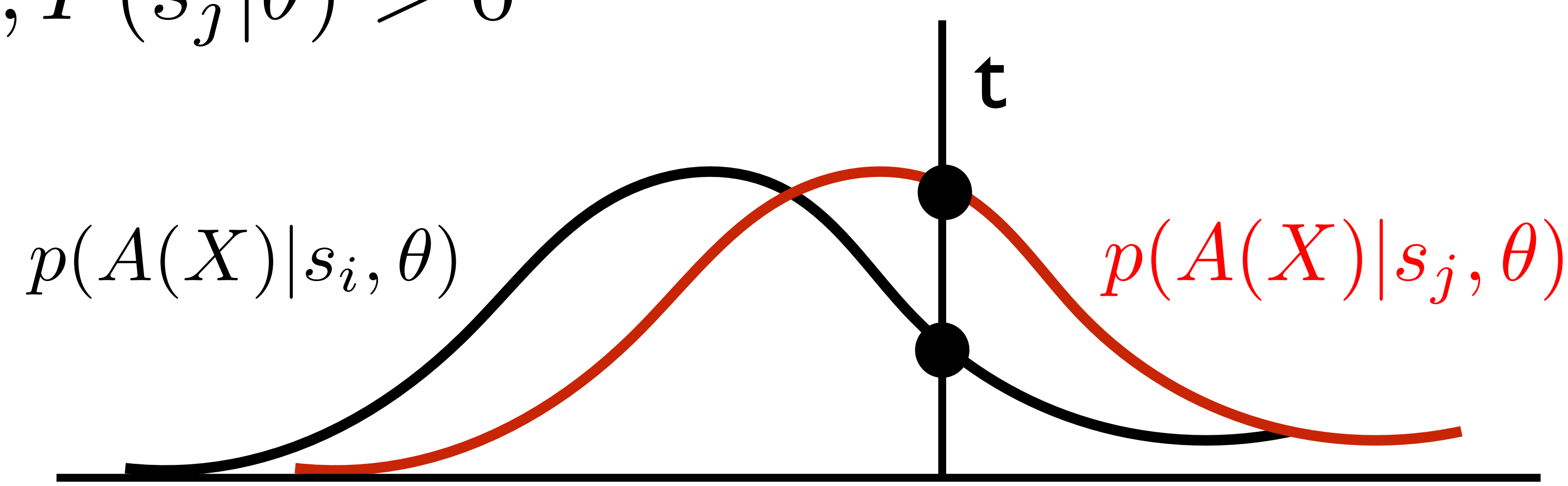
Secret Pairs Set Q

Distribution Class Θ

An algorithm A is ϵ-Pufferfish private with parameters (S, Q, Θ) if for all (s_i, s_j) in Q, for all $\theta \in \Theta$, $X \sim \theta$, all t,

$$p_{\theta,A}(A(X) = t | s_i, \theta) \leq e^{\epsilon} \cdot p_{\theta,A}(A(X) = t | s_j, \theta)$$

whenever $P(s_i|\theta), P(s_j|\theta) > 0$



Pufferfish Interpretation of DP

Theorem: Pufferfish = Differential Privacy when:

S = { $s_{i,a}$:= Person i has value a, for all i, all a in domain X }

Q = { ($s_{i,a}$ $s_{i,b}$), for all i and (a, b) pairs in X x X }

Θ = { Distributions where each person i is independent }

Pufferfish Interpretation of DP

Theorem: Pufferfish = Differential Privacy when:

S = { $s_{i,a}$:= Person i has value a, for all i, all a in domain X }

Q = { ($s_{i,a}$ $s_{i,b}$), for all i and (a, b) pairs in X x X }

Θ = { Distributions where each person i is independent }

Theorem: No utility possible when:

Θ = { All possible distributions }

How to get Pufferfish privacy?

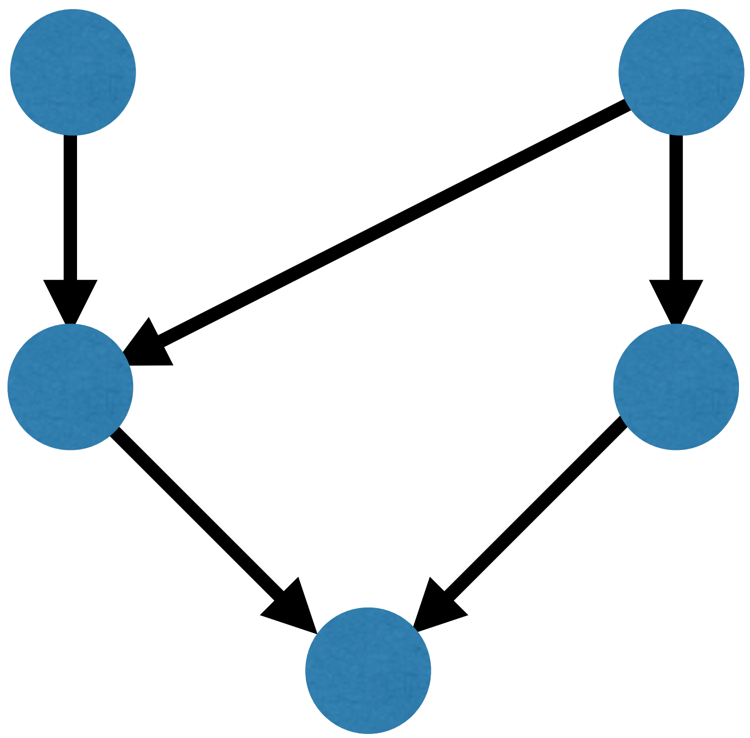
Special case mechanisms [KM12, HMD12]

Is there a **more general** Pufferfish mechanism for a large class of correlated data?

Our work: Yes, the **Markov Quilt Mechanism**

(Also concurrent work [GK16])

Correlation Measure: Bayesian Networks



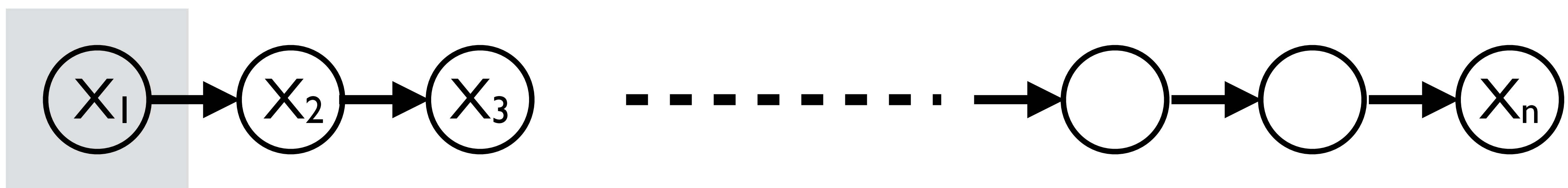
Node: variable

Directed Acyclic Graph

Joint distribution of variables:

$$\Pr(X_1, X_2, \ldots, X_n) = \prod_i \Pr(X_i | \text{parents}(X_i))$$

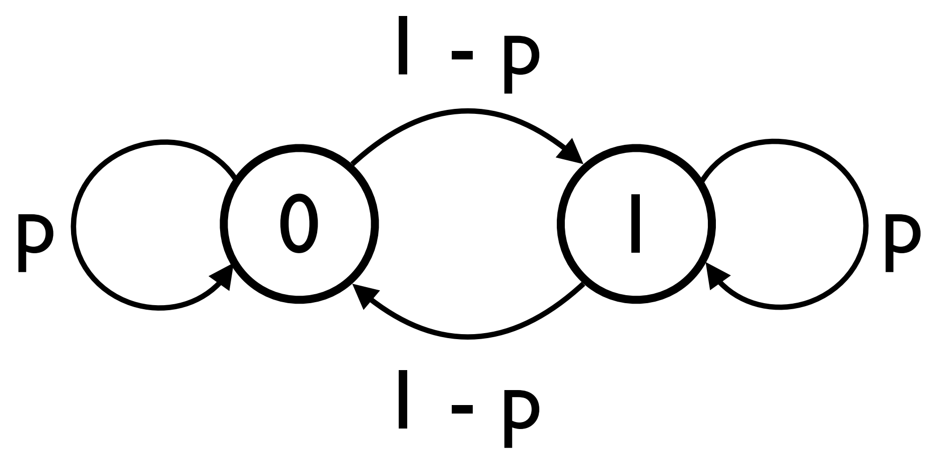
A Simple Example



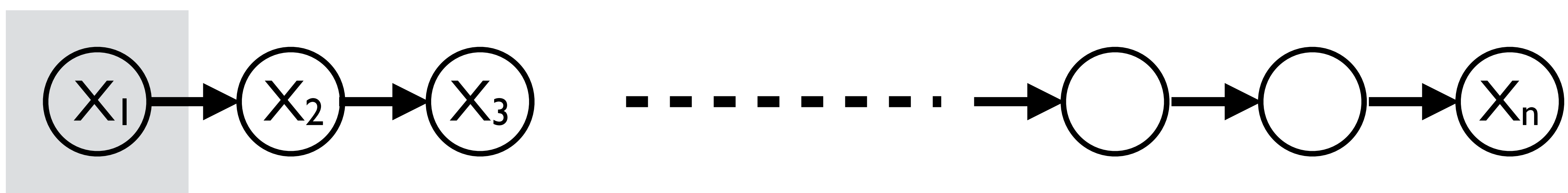
Model:

X_i in {0, 1}

State Transition Probabilities:



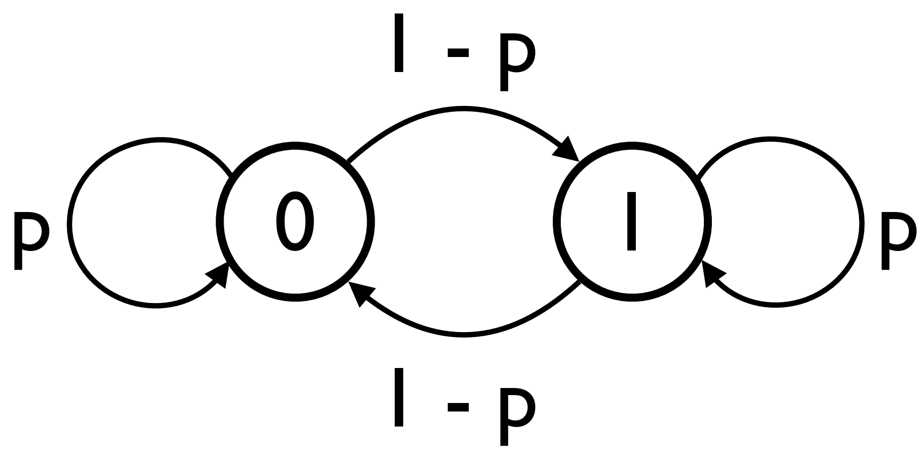
A Simple Example



Model:

X_i in {0, 1}

State Transition Probabilities:

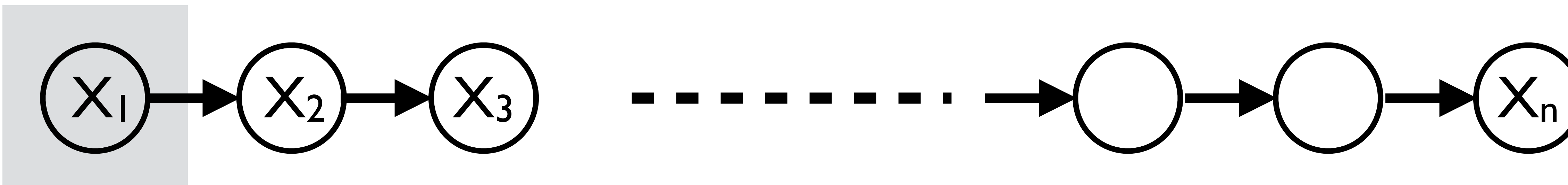


$Pr(X_2 = 0 | X_1 = 0) = p$

$Pr(X_2 = 0 | X_1 = 1) = 1 - p$

....

A Simple Example



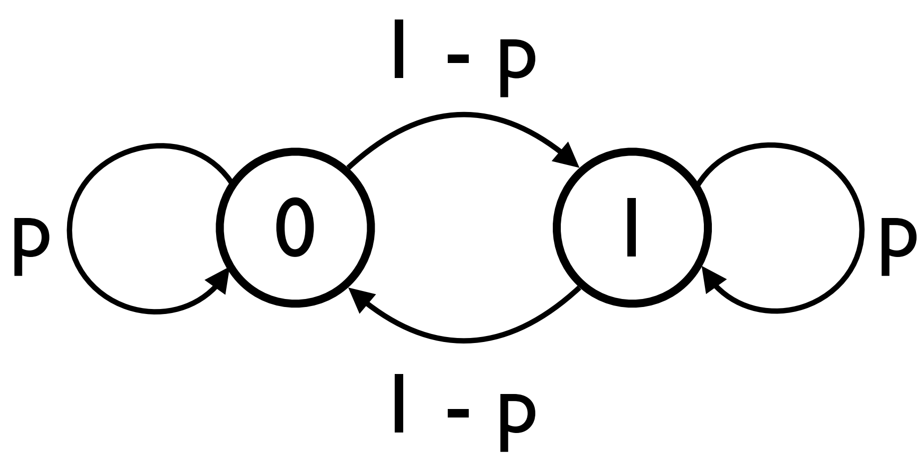
Model:

X_i in {0, 1}

$Pr(X_2 = 0 | X_1 = 0) = p$

$Pr(X_2 = 0 | X_1 = 1) = 1 - p$

State Transition Probabilities:



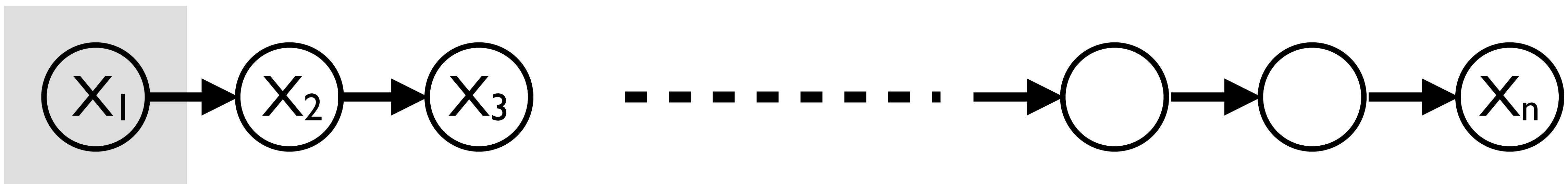
....

$Pr(X_i = 0 | X_1 = 0) = \frac{1}{2} + \frac{1}{2}(2p-1)^{i-1}$

$Pr(X_i = 0 | X_1 = 1) = \frac{1}{2} - \frac{1}{2}(2p-1)^{i-1}$

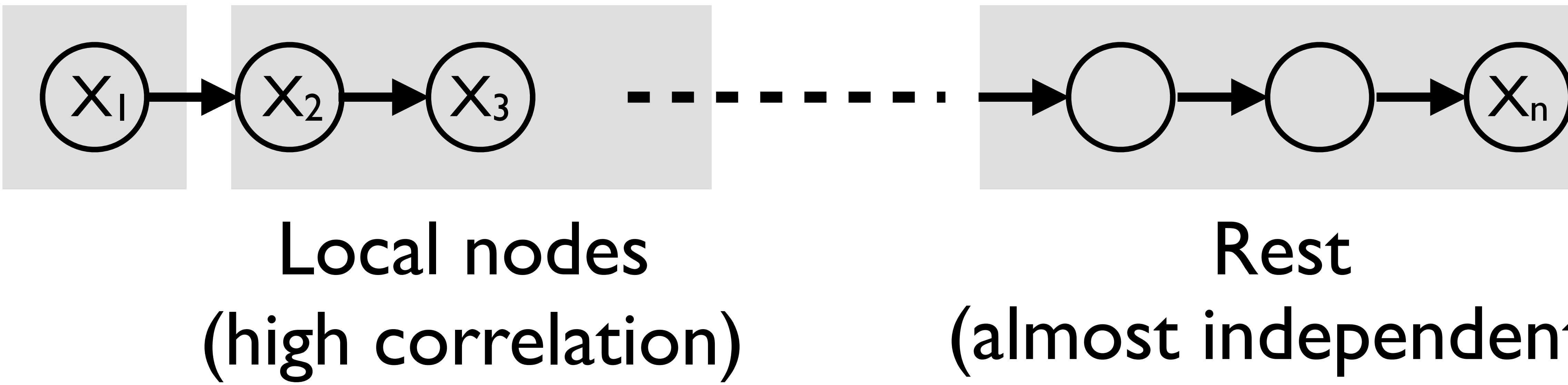
Influence of X_1 diminishes with distance

Algorithm: Main Idea



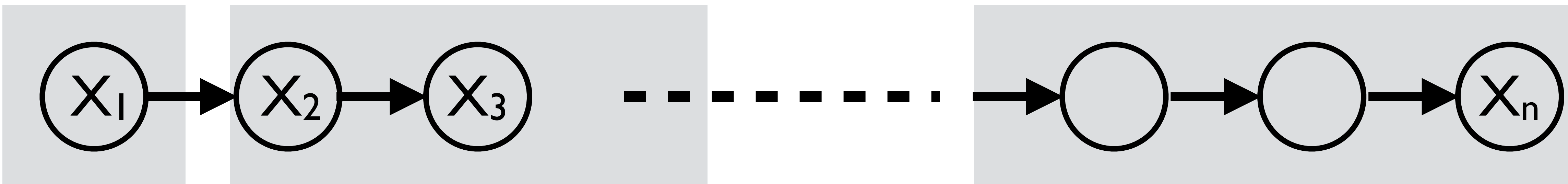
Goal: Protect X_1

Algorithm: Main Idea



Goal: Protect X_1

Algorithm: Main Idea



Local nodes
(high correlation)

Rest
(almost independent)

Goal: Protect X_1

Add noise to hide local nodes **+** Small correction for rest

Measuring "Independence"

Max-influence of X_i on a set of nodes X_R:

$$e(X_R|X_i) = \max_{a,b} \sup_{\theta \in \Theta} \max_{x_R} \log \frac{\Pr(X_R = x_R | X_i = a, \theta)}{\Pr(X_R = x_R | X_i = b, \theta)}$$

Low $e(X_R|X_i)$ means X_R is almost independent of X_i

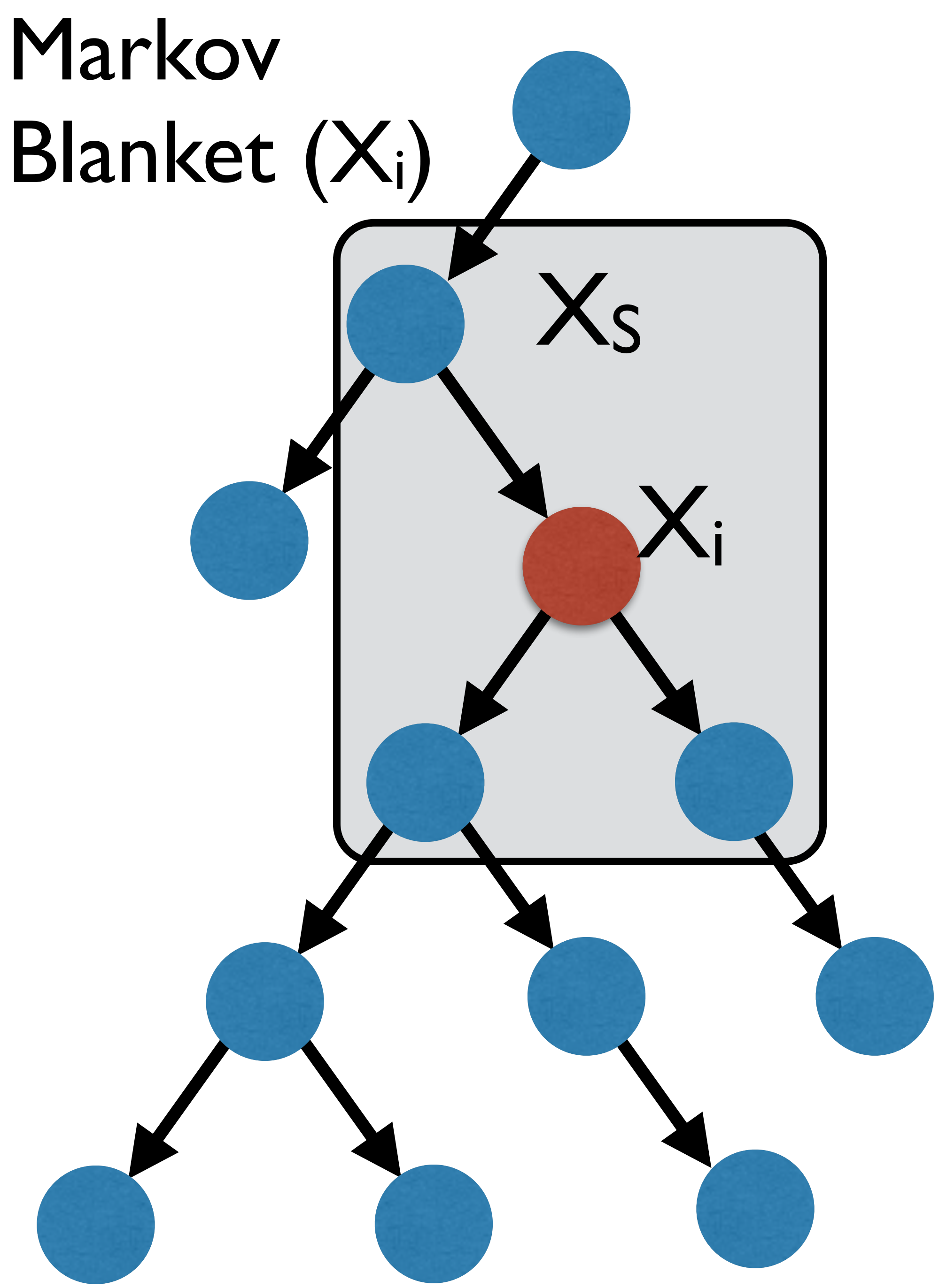
To protect X_i, correction term needed for X_R is $\exp(e(X_R|X_i))$

How to find large “almost independent” sets

Brute force search is expensive

Use structural properties of the Bayesian network

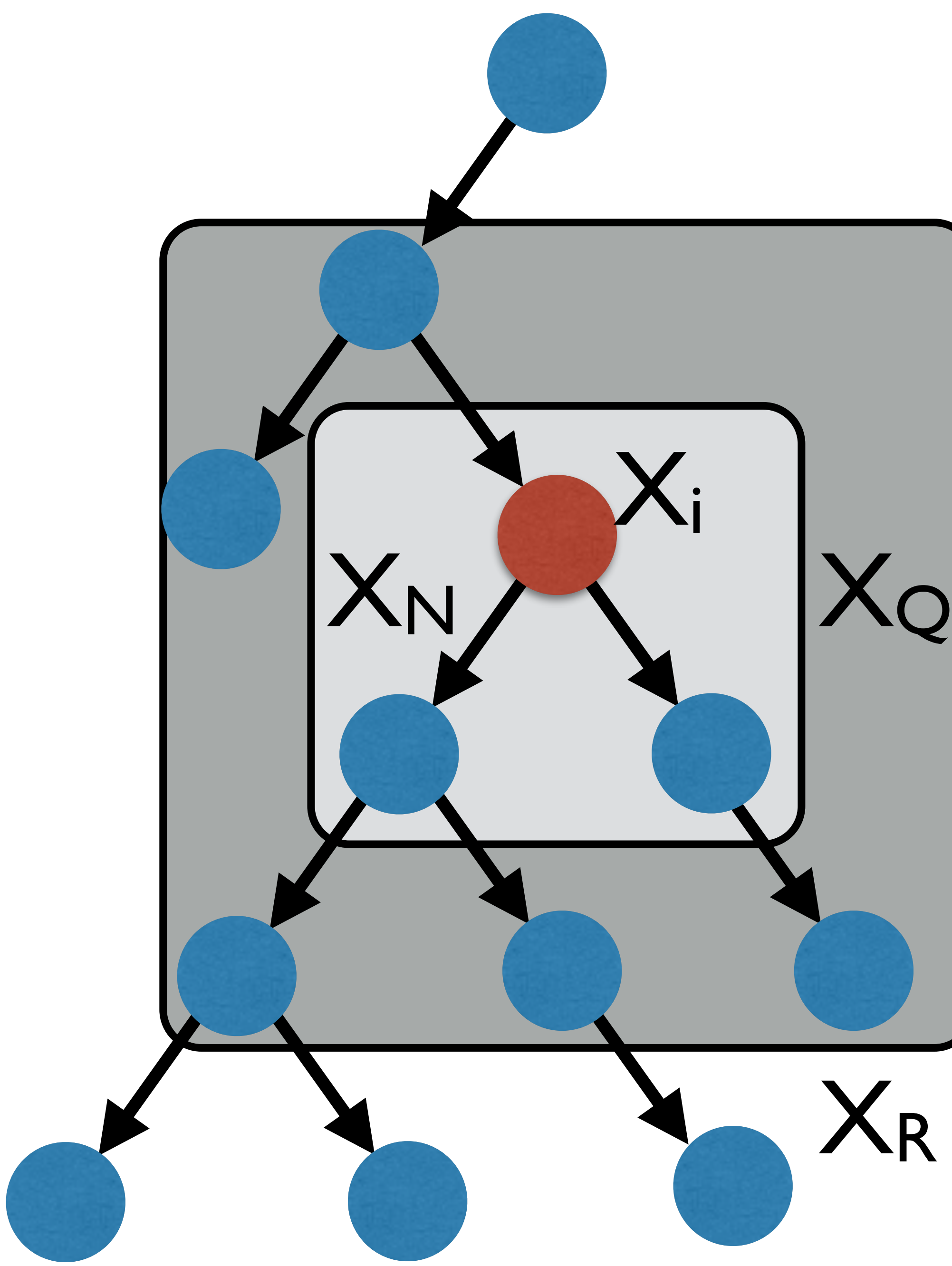
Markov Blanket



Markov Blanket(X_i) =
Set of nodes X_S s.t X_i is
independent of $X \backslash (X_i \cup X_S)$
given X_S

(usually, parents, children,
other parents of children)

Define: Markov Quilt

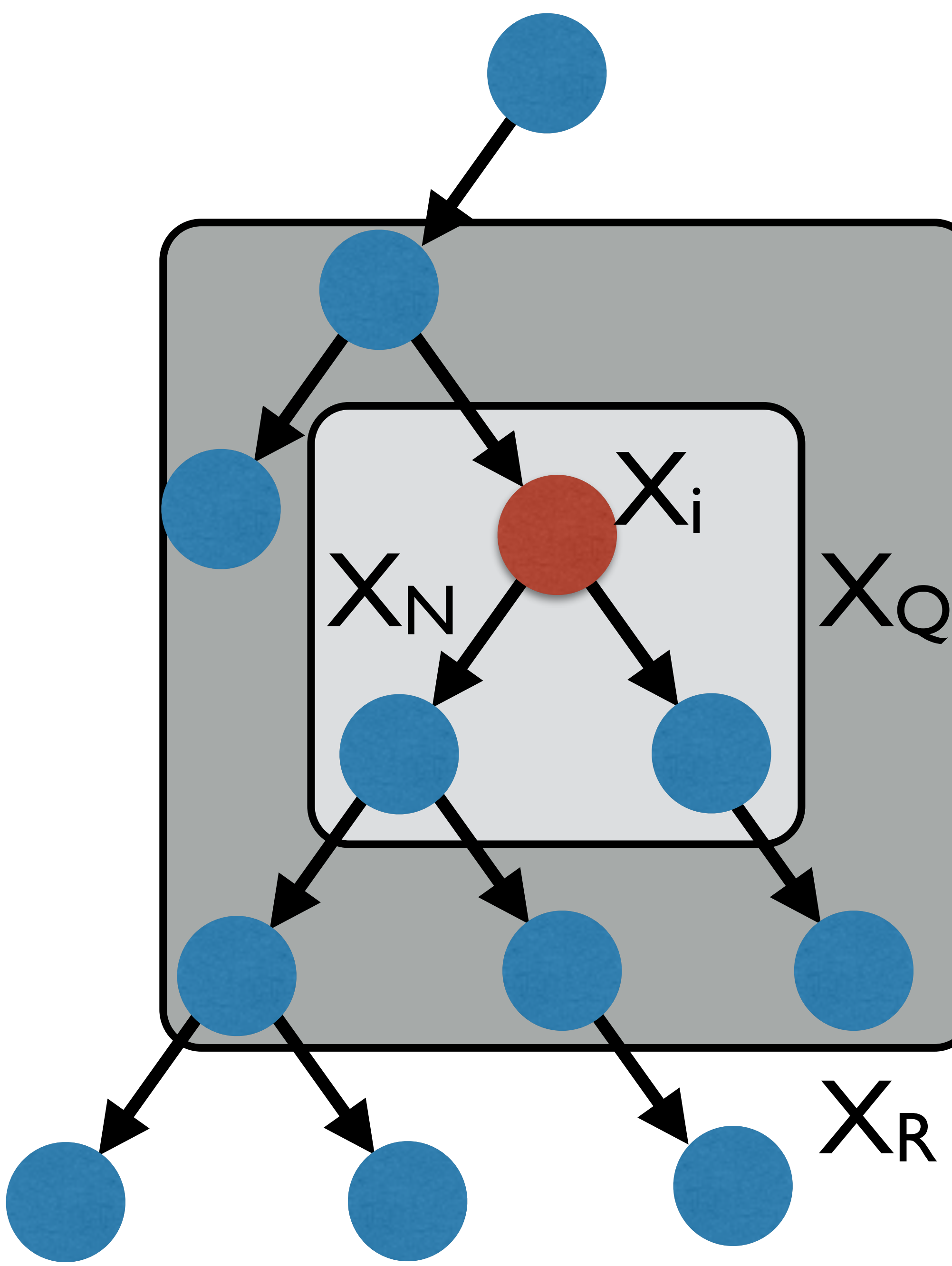


X_Q is a Markov Quilt of X_i if:

1. Deleting X_Q breaks graph into X_N and X_R
2. X_i lies in X_N
3. X_R is independent of X_i given X_Q

(For Markov Blanket $X_N = X_i$)

Why do we need Markov Quilts?

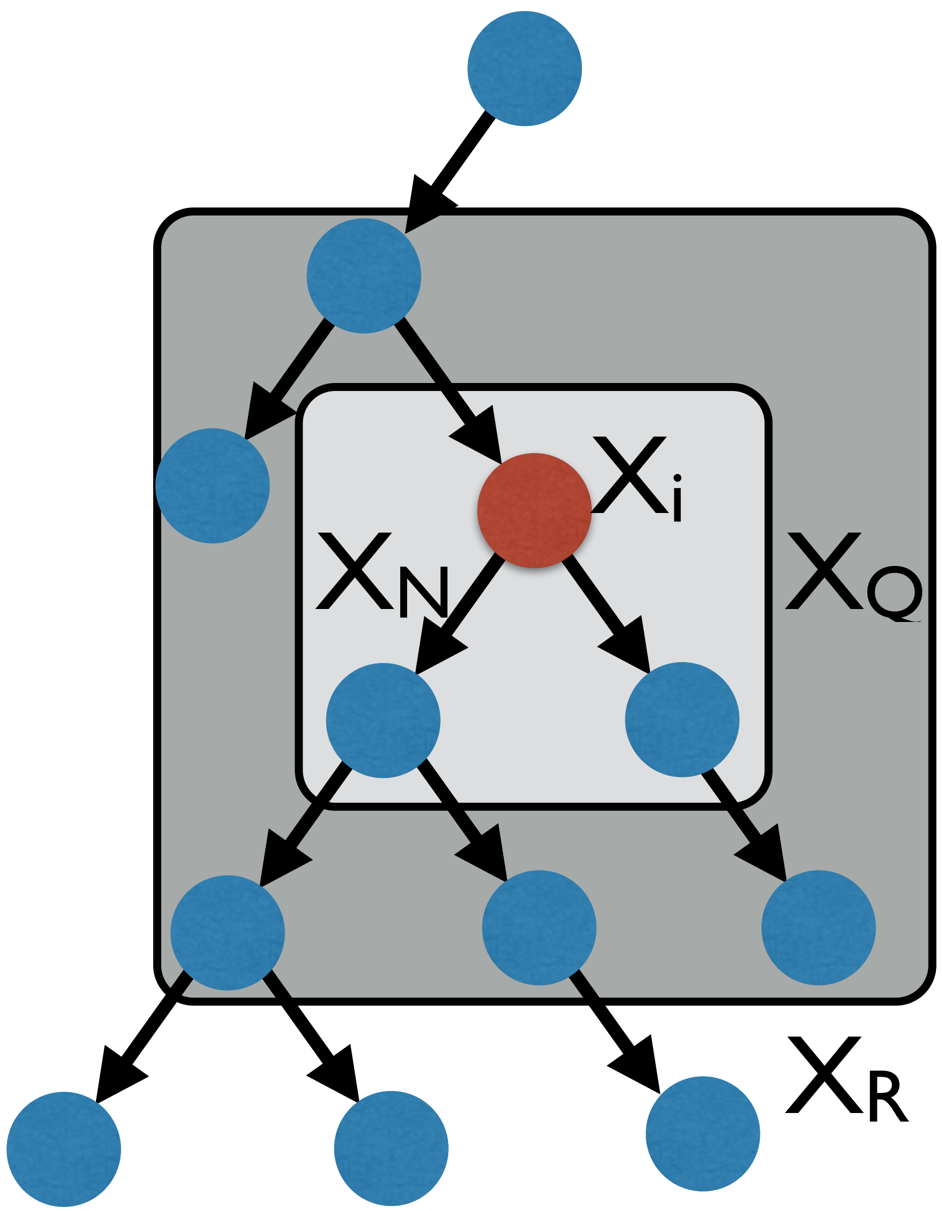


Given a Markov Quilt,

X_N = local nodes for X_i

X_Q U X_R = rest

From Markov Quilts to Amount of Noise



Let X_Q = Markov Quilt for X_i

Stdev of noise to protect X_i:

Noise due to X_N

$$\text{Score}(X_Q) = \frac{card(X_N)}{\epsilon - e(X_Q|X_i)}$$

Correction for $X_Q \cup X_R$

Search all Markov Quilts to find one that needs min noise

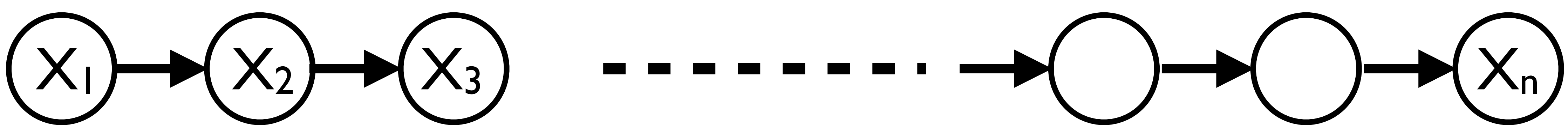
Privacy Properties

Privacy: MQM is ϵ-Pufferfish private

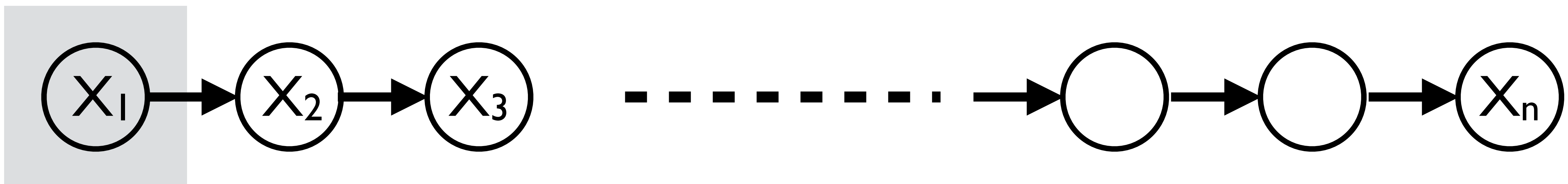
Graceful Composition

MQM for Markov Chains has:

- **Additive** sequential composition
- Parallel composition with a correction term



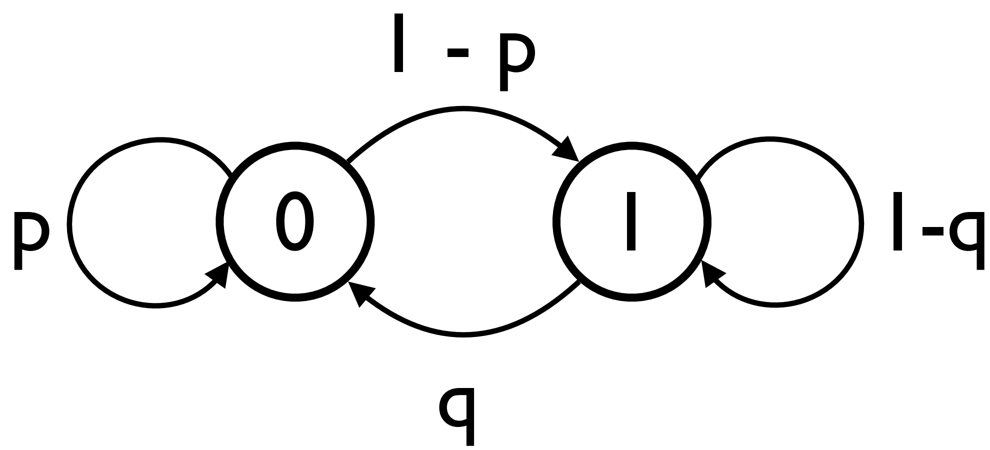
Simulations - Task



Model:

X_i in {0, 1}

State Transition Probabilities:



Model Class:

$$\Theta = [\ell, 1 - \ell]$$

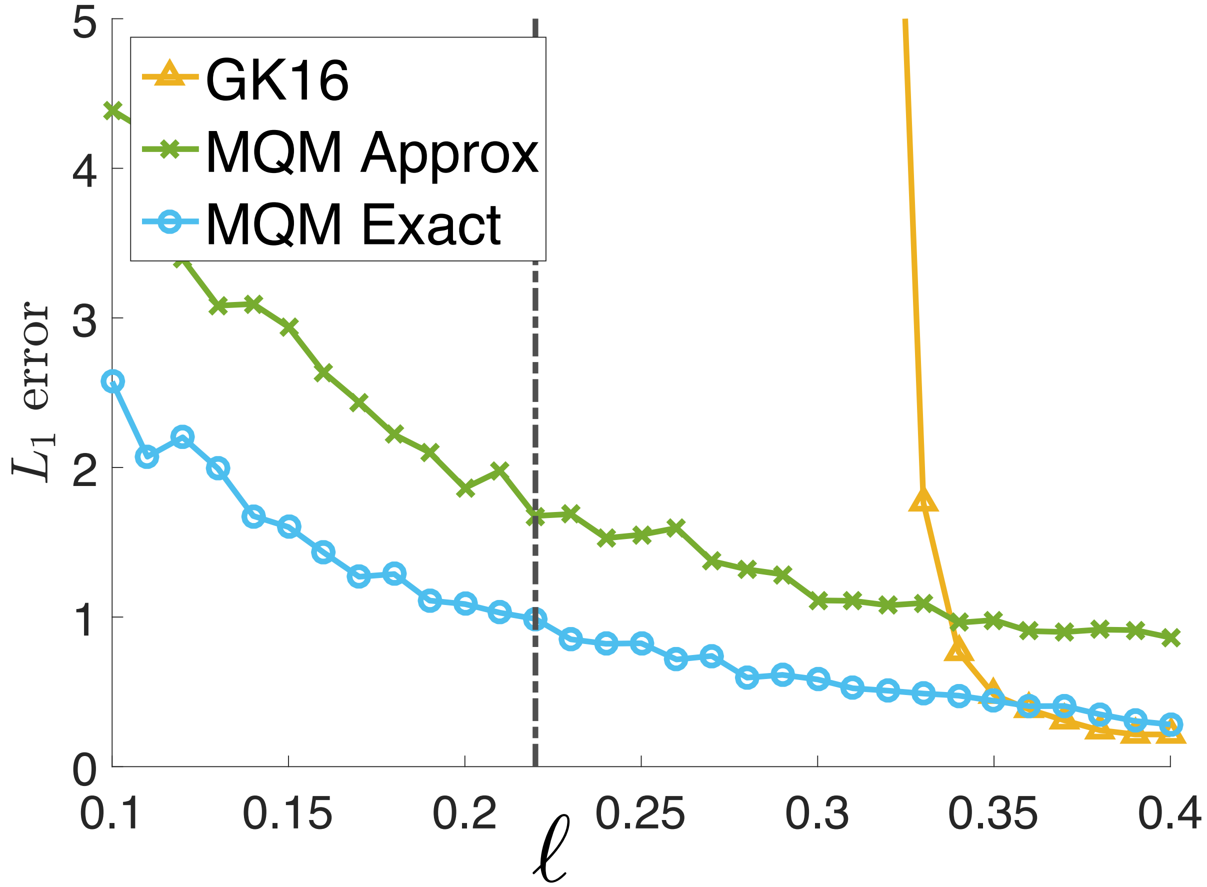
(implies p and q can lie anywhere in Θ)

Sequence length = 100

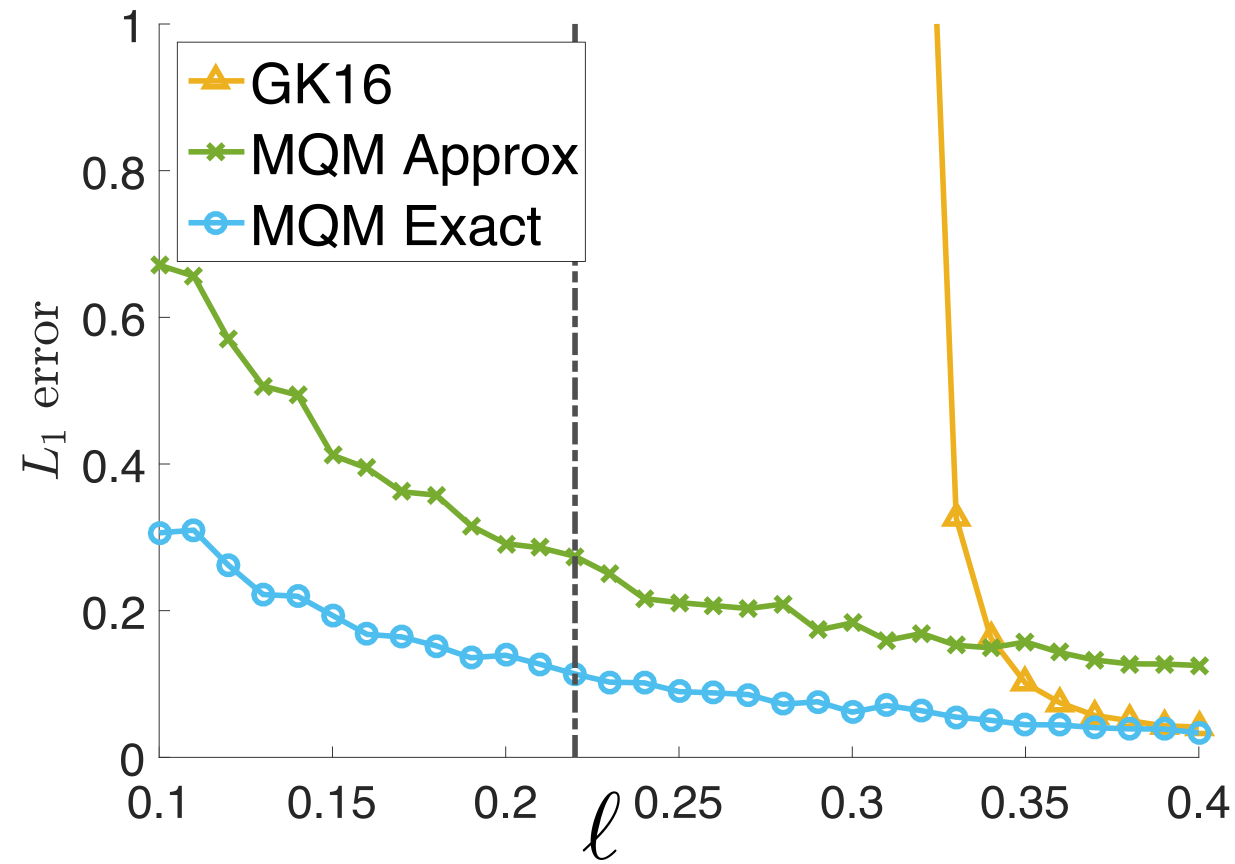
Simulations - Results

Methods:

- Two versions of Markov Quilt Mechanism (MQMExact, MQMApprox)
- GK16



$\epsilon = 0.2$



$\epsilon = 1$

Real Data - Activity Measurement

Dataset on physical activity by three groups of subjects:
40 cyclists, 16 older women and 36 overweight women

4 states (active, standing still, standing moving, sedentary)

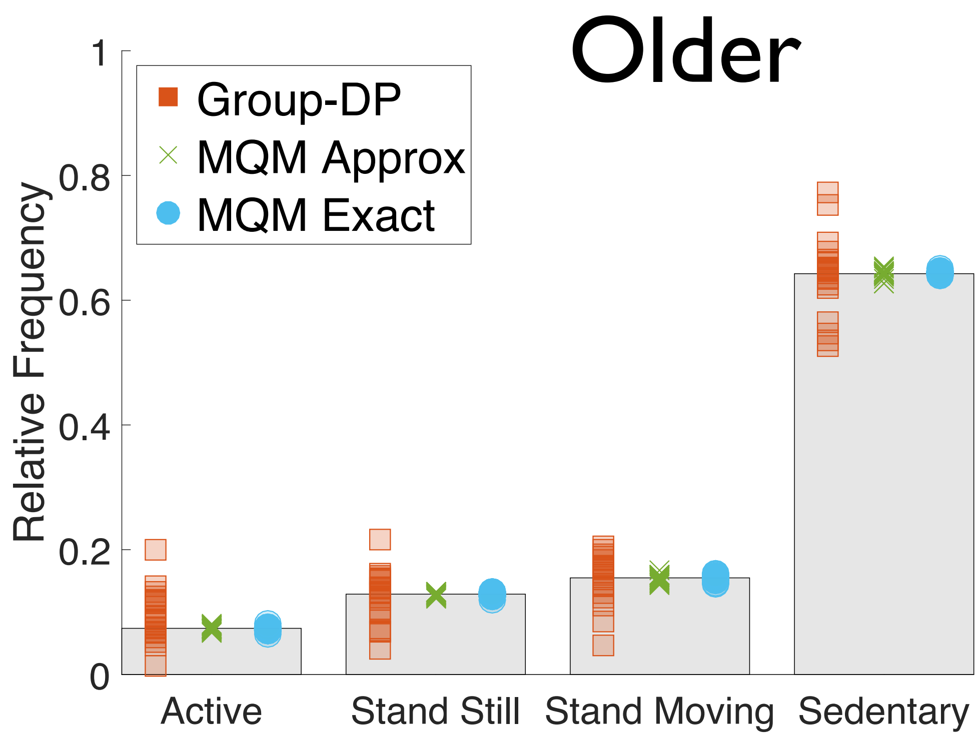
Over 9,000 observations per subject

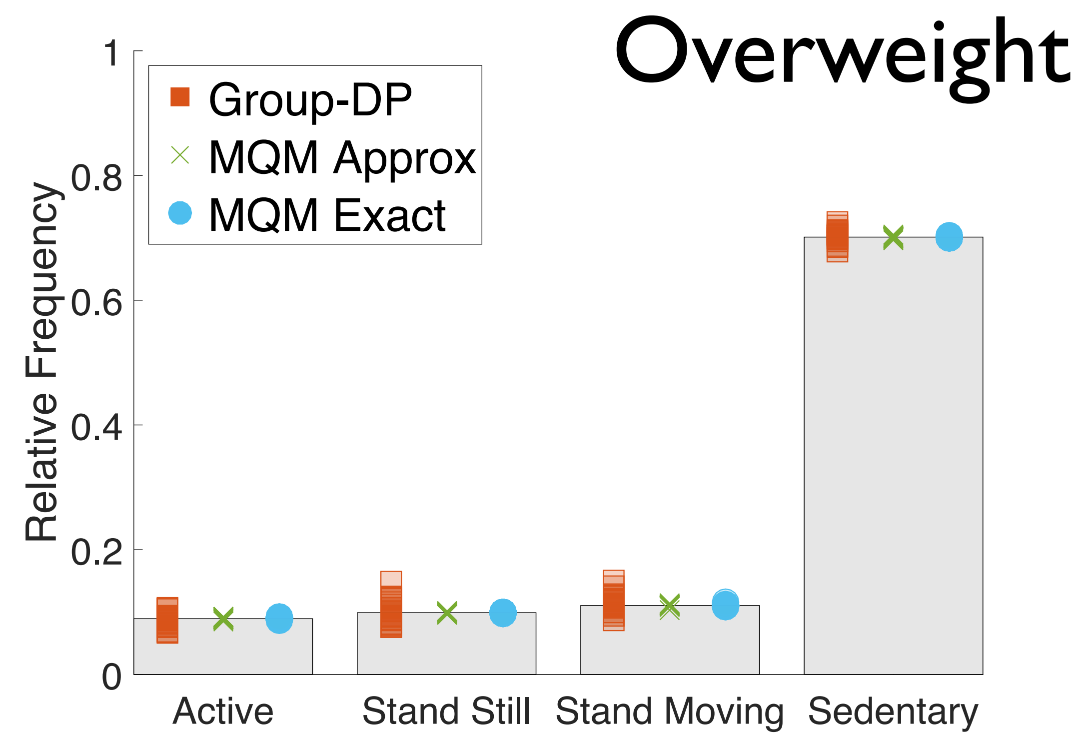
Θ = { Empirical data generating distribution }

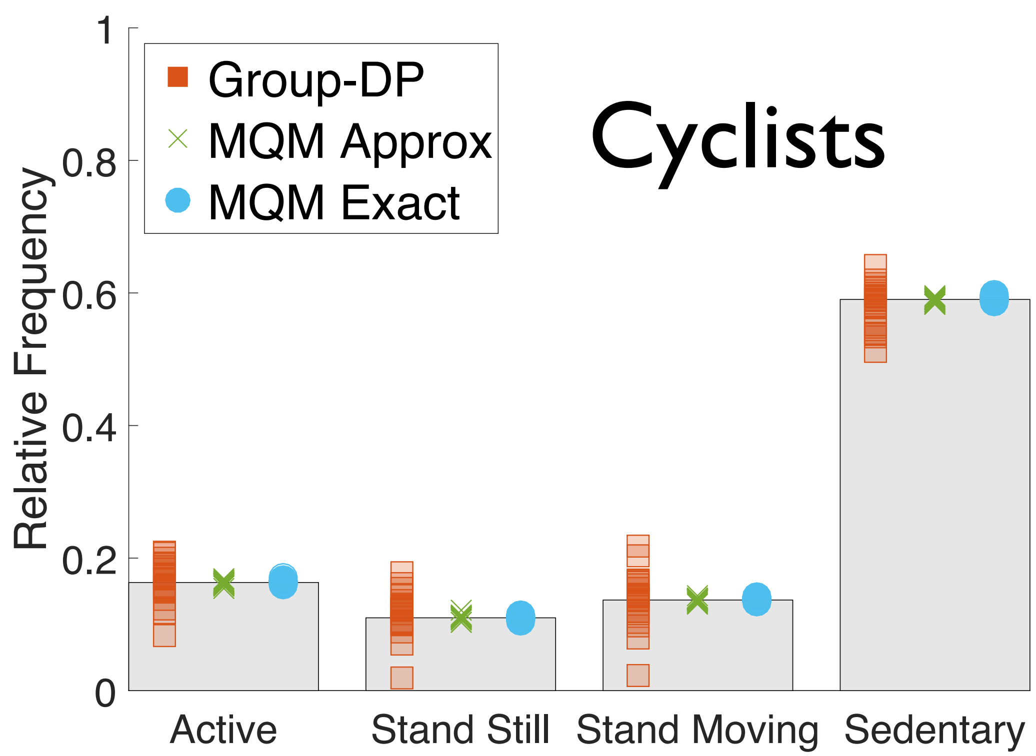
Methods:

- MQMExact and MQMApprox
- GroupDP
- GK16 does not apply

Real Data - Activity Measurement







Aggregated results
(over groups)

$$\epsilon = 1$$

Real Data - Power Consumption

Dataset on power consumption in a single household

Power consumption discretized to 51 levels (51 states)

Over 1 million observations

Θ = { Empirical data generating distribution }

Methods:

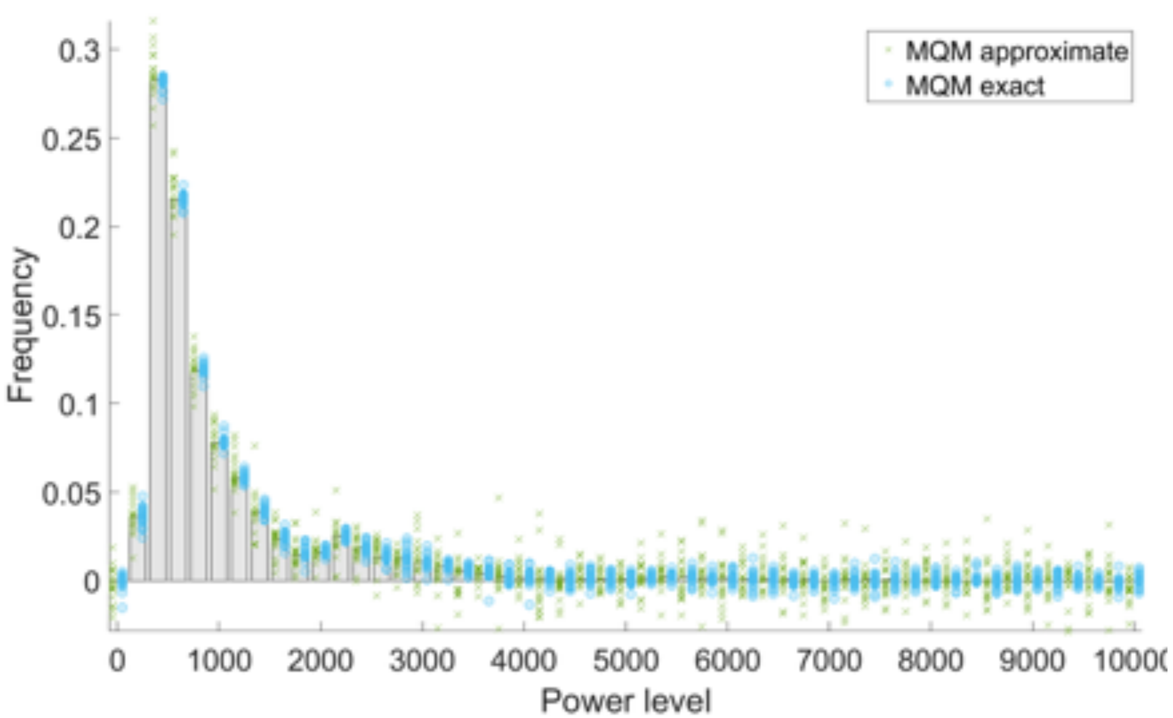
MQMExact vs. MQMApprox

GK16 does not apply

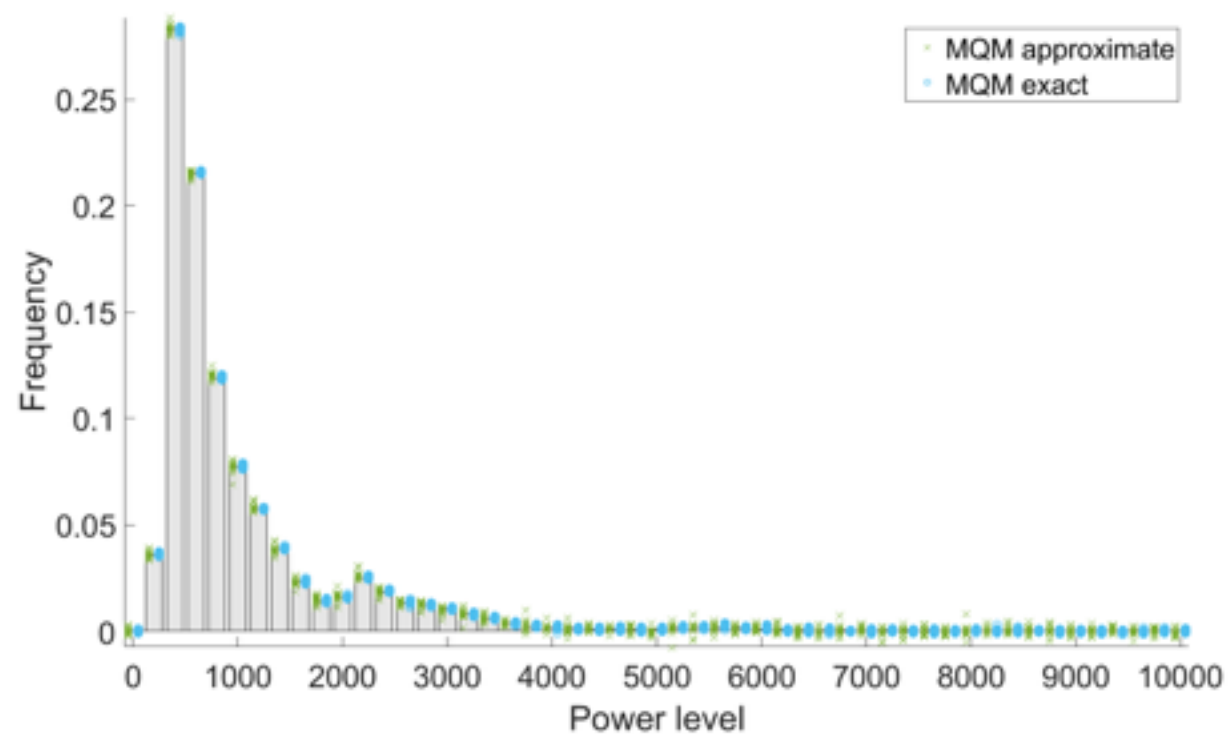
GroupDP has too little utility

Real Data - Power Consumption

Methods:
Two versions of Markov Quilt Mechanism (MQMExact, MQMApprox)



$\epsilon = 0.2$



$\epsilon = 1$

Conclusion

- Real problems have complex privacy challenges
- Rigorous privacy definitions are available
- For any privacy problem, important to think:
  - What do we need to hide?
  - What do we need to reveal?

References

- *"Differentially Private Continual Release of Graph Statistics"*, S. Song, S. Mehta, S. Vinterbo, S. Little and K. Chaudhuri, Arxiv, 2018
- *"Pufferfish Privacy Mechanisms for Correlated Data"*, S. Song, Y. Wang and K. Chaudhuri, SIGMOD 2018.
- *"Composition Properties of Inferential Privacy on Time-Series Data"*, S. Song and K. Chaudhuri, Allerton 2018.

Thanks!