Distributed size estimation in anonymous networks

Damiano Varagnolo, Gianluigi Pillonetto, Luca Schenato

Department of Information Engineering, University of Padova

24th October 2011
Table of Contents

1. Introduction
2. General estimation scheme
3. Continuous distributions
4. Discrete distributions
5. Robustness
Table of Contents

1 Introduction
2 General estimation scheme
3 Continuous distributions
4 Discrete distributions
5 Robustness
Focus of this talk:

- distributed estimation of the size $S$ of a network

$\rightarrow$ i.e. let the agents know how many they are
Motivations (1/3): network maintenance purposes
Motivations (2/3): smart buildings management
Motivations (3/3): estimation purposes

(also $S^{-1}$ may be interesting!!)
Problem definition

hypotheses

- $S :=$ network size
- $S$ deterministic and constant in time
- agents have \textit{limited computational / memory / communication capabilities}
- network is \textit{anonymous}
  (no IDs or IDs not assured to be unique)
Problem definition

hypotheses

- $S := \text{network size}$
- $S$ deterministic and constant in time
- agents have \textit{limited computational / memory / communication capabilities}
- network is \textit{anonymous}
  (no IDs or IDs not assured to be unique)

Goal: develop a distributed estimator $\hat{S}$ of $S$ satisfying the constraints
network size estimation = not a new problem!!
Literature review

network size estimation = not a new problem!!

Deterministic scenario: theoretical limit for anonymous networks

\[ \exists \text{ algorithm (with bounded average bit complexity) guaranteed to return the correct answer for every (finite) execution} \]

Cidon, Shavitt (1995), Information Processing Letters
Literature review

network size estimation = not a new problem!!

Deterministic scenario: theoretical limit for anonymous networks

∀ algorithm (with bounded average bit complexity) guaranteed to return the correct answer for every (finite) execution

Cidon, Shavitt (1995), Information Processing Letters

Stochastic scenario: some existing approaches
Literature review

network size estimation = not a new problem!!

Deterministic scenario: theoretical limit for anonymous networks

∀ algorithm (with bounded average bit complexity) guaranteed to return the correct answer for every (finite) execution

Cidon, Shavitt (1995), Information Processing Letters

Stochastic scenario: some existing approaches

• random walk strategies
Literature review

network size estimation = not a new problem!!

Deterministic scenario: theoretical limit for anonymous networks

∀ algorithm (with bounded average bit complexity) guaranteed to return the correct answer for every (finite) execution

Cidon, Shavitt (1995), Information Processing Letters

Stochastic scenario: some existing approaches

- random walk strategies
- capture-recapture strategies
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)

Peer counting and sampling in overlay networks: random walk methods

ACM symposium on Principles of distributed computing

Algorithm
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)
Peer counting and sampling in overlay networks: random walk methods
ACM symposium on Principles of distributed computing

Algorithm
1. generate a “seed”
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)

Peer counting and sampling in overlay networks: random walk methods

ACM symposium on Principles of distributed computing

Algorithm

1. generate a “seed”
2. randomly propagate it
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)

Peer counting and sampling in overlay networks: random walk methods

ACM symposium on Principles of distributed computing

Algorithm

1. generate a “seed”
2. randomly propagate it
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)

Peer counting and sampling in overlay networks: random walk methods

ACM symposium on Principles of distributed computing

Algorithm

1. generate a “seed”
2. randomly propagate it
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)
Peer counting and sampling in overlay networks: random walk methods
ACM symposium on Principles of distributed computing

Algorithm
1. generate a “seed”
2. randomly propagate it
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)
Peer counting and sampling in overlay networks: random walk methods
ACM symposium on Principles of distributed computing

Algorithm
1. generate a “seed”
2. randomly propagate it
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)
Peer counting and sampling in overlay networks: random walk methods
ACM symposium on Principles of distributed computing

Algorithm
1. generate a “seed”
2. randomly propagate it
3. # of jumps → statistically dependent on $S$
Random walks

Massoulié, Le Merrer, Kermarrec, Ganesh (2006)

Peer counting and sampling in overlay networks: random walk methods

ACM symposium on Principles of distributed computing

Algorithm

1. generate a “seed”
2. randomly propagate it
3. # of jumps $\rightarrow$ statistically dependent on $S$
4. variance of the error:
   \[ \propto \left(\# \text{ of generated seeds}\right)^{-1} \]
Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters

London: Charles Griffin & Co.

---

Algorithm
Introduction

Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm

1. generate $N$ seeds

varagnolo@dei.unipd.it (DEI - UniPD)
Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm

1. generate $N$ seeds
2. propagate them
Capture-recapture

Seber (1982)
The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm
1. generate $N$ seeds
2. propagate them
Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters

London: Charles Griffin & Co.

Algorithm

1. generate $N$ seeds
2. propagate them
Capture-recapture

Seber (1982)
The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm
1. generate $N$ seeds
2. propagate them
Capture-recapture

Seber (1982)
The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm
1. generate $N$ seeds
2. propagate them
Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm

1. generate $N$ seeds
2. propagate them
3. capture and infer

varagnolo@dei.unipd.it (DEI - UniPD)
Capture-recapture

Seber (1982)

The estimation of animal abundance and related parameters
London: Charles Griffin & Co.

Algorithm

1. generate $N$ seeds
2. propagate them
3. capture and infer
4. variance of the error: $\propto \# \text{ of captured seeds}$ (polynomially)
Our algorithm

several peculiarities w.r.t. existing literature
Our algorithm

several peculiarities w.r.t. existing literature

- full parallelism → *every agent will have an estimate at the same time*
Our algorithm

several peculiarities w.r.t. existing literature

- full parallelism → every agent will have an estimate at the same time
- easily implementable in anonymous networks
Our algorithm

several peculiarities w.r.t. existing literature

- full parallelism \(\rightarrow\) every agent will have an estimate at the same time
- easily implementable in anonymous networks
- nice mathematical properties
Our algorithm

several peculiarities w.r.t. existing literature

- full parallelism → every agent will have an estimate at the same time
- easily implementable in anonymous networks
- nice mathematical properties

the idea: generate random numbers → combine them with consensus → exploit statistical inference

Cohen (1997), Journal of Computer and System Sciences, Size-estimation framework with applications to transitive closure and reachability
Table of Contents

1 Introduction
2 General estimation scheme
3 Continuous distributions
4 Discrete distributions
5 Robustness
Block representation of our strategy

$p(\cdot)$

\[ y_{1,1}, \ldots, y_{1,M}, y_{2,1}, \ldots, y_{2,M}, \ldots, y_{S,1}, \ldots, y_{S,M} \]

\[ F \quad F \quad F \]

\[ f_1, f_2, f_M \]

\[ \psi \rightarrow \hat{S} \quad (or \ \hat{S}^{-1}) \]
every agent $i$ generates a $M$-tuple $\{y_{i,1}, \ldots, y_{i,M}\}$, $y_{i,m} \sim p(\cdot)$
General estimation scheme

Block representation of our strategy

The $S$-tuples $\{y_{1,m}, \ldots, y_{S,m}\}$ are converted into a scalar $f_m$ through $F$ (e.g. $F = \text{average}$, $F = \text{max}$)

\[ \text{local} \quad \begin{array}{c} y_{1,1} \\ y_{2,1} \\ \vdots \\ y_{S,1} \\ y_{1,2} \\ y_{2,2} \\ \vdots \\ y_{S,2} \\ y_{1,M} \\ y_{2,M} \\ \vdots \\ y_{S,M} \end{array} \quad \begin{array}{c} F \\ F \\ \vdots \\ F \\ F \\ \vdots \\ F \end{array} \quad \Psi \quad \tilde{S} \quad \text{(or } \tilde{S}^{-1} \text{)} \quad \text{local} \]

\[ f_1 \\ f_2 \\ \vdots \\ f_M \]
the $M$-tuple $\{f_1, \ldots, f_M\}$ is converted into an estimate $\hat{S}$ through $\Psi$
(e.g. $\Psi = \text{Maximum Likelihood}$)
Block representation of our strategy

General estimation scheme

Cost function: \[ J(p, F, \Psi) := \mathbb{E} \left[ (S - \hat{S})^2 \right] \]
Algorithm \((M = 1)\):
An example

Algorithm ($M = 1$):

Local generation with $\rho = \mathcal{N}(0, 1)$

$y_1 \sim \mathcal{N}(0, 1)$

$y_2 \sim \mathcal{N}(0, 1)$

$y_3 \sim \mathcal{N}(0, 1)$

$y_4 \sim \mathcal{N}(0, 1)$

$y_5 \sim \mathcal{N}(0, 1)$
General estimation scheme

An example

Algorithm ($M = 1$):

Local generation with $p = \mathcal{N}(0, 1)$

$F = \text{average consensus}$

$y_5 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i$

$y_2 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i$

$y_3 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i$

$y_4 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i$
An example

Algorithm ($M = 1$):

- **Local generation** with $p = \mathcal{N}(0, 1)$

$$y_{\text{ave}} \sim \mathcal{N}(0, \frac{1}{S})$$

$F = \text{average consensus}$

$$y_{\text{ave}} \sim \mathcal{N}(0, \frac{1}{S})$$
An example

Algorithm ($M = 1$):

- **Local generation** with $p = \mathcal{N}(0, 1)$
- $F = \text{average consensus}$
- $\Psi = \text{Maximum Likelihood}$
- $\hat{S} = y_{\text{ave}}^{-2}$

$y_{\text{ave}} \sim \mathcal{N}(0, \frac{1}{\hat{S}})$
A formidable infinite-dimensional problem

\[ \text{arg min}_{\rho, F, \Psi} J(\rho, F, \Psi) = ??? \]

\[ J(\rho, F, \Psi) := \mathbb{E}\left[ (S - \hat{S})^2 \right] \]
Our case studies

Case 1:

Gaussian distribution

\[ F = \text{ave.} \]

\[ y_1,1, \quad y_2,1, \quad \ldots, \quad y_{S,1} \]

\[ y_1,2, \quad y_2,2, \quad \ldots, \quad y_{S,2} \]

\[ y_1,M, \quad y_2,M, \quad \ldots, \quad y_{S,M} \]

\[ F = \text{ave.} \]

\[ f_1 \]

\[ f_2 \]

\[ f_M \]

\[ \psi = \text{ML} \]

\[ \hat{S} \]
Our case studies

Case 2:

Absolutely continuous distribution

\[ F = \max \]

\[ y_{1,1}, y_{2,1}, \ldots, y_{S,1} \]

\[ y_{1,2}, y_{2,2}, \ldots, y_{S,2} \]

\[ y_{1,M}, y_{2,M}, \ldots, y_{S,M} \]

\[ F = \max \]

\[ f_1, f_2, \ldots, f_M \]

\[ \Psi = \text{ML} \]

\[ \hat{S} \]
Our case studies

Case 3:

\[
\begin{align*}
F &= \text{ave.} \\
\Psi &= \text{ML} \\
\hat{S} &= \text{dist. est.}
\end{align*}
\]
An historical case study

The German Tank problem

infer tanks production from serial numbers analysis
(June 1940 → September 1942)

<table>
<thead>
<tr>
<th>intelligence</th>
<th>statisticians</th>
<th>actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>1400</td>
<td>256</td>
<td></td>
</tr>
</tbody>
</table>
An historical case study

The German Tank problem

infer tanks production from serial numbers analysis

(June 1940 → September 1942)

<table>
<thead>
<tr>
<th>intelligence</th>
<th>statisticians</th>
<th>actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>1400</td>
<td>256</td>
<td></td>
</tr>
</tbody>
</table>

$F = \max \text{ID \\ 
& \# of tanks}$

$\Psi = \text{MVUE}$
General estimation scheme

An historical case study

The German Tank problem

Infer tanks production from serial numbers analysis
(June 1940 → September 1942)

<table>
<thead>
<tr>
<th>intelligence</th>
<th>statisticians</th>
<th>actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>1400</td>
<td>256</td>
<td>255</td>
</tr>
</tbody>
</table>

\[ F = \max \text{ID} \& \#\text{of tanks} \]

\[ \psi = \text{MVUE} \]
Case 1: \((p \text{ Gaussian}) + (F = \text{average}) + (\Psi = \text{ML})\)

\[
p = \mathcal{N}(\mu, \sigma^2)
\]

\[
\{y_{1,m}\}, \{y_{2,m}\}, \ldots, \{y_{S,m}\}
\]

\[
F = \text{ave. cons.}
\]

\[
\hat{S}
\]

\[
\hat{S} \sim \text{Inv}-\chi^2(M)
\]

\[
E[\hat{S}^2] = M - \frac{2\text{var}(\hat{S} - S)}{S}
\]
Case 1: \( (p \text{ Gaussian}) + (F = \text{average}) + (\Psi = \text{ML}) \)

\[
p = \mathcal{N}(\mu, \sigma^2)
\]

\[
\{y_{1,m}\} \quad \{y_{2,m}\} \quad \cdots \quad \{y_{S,m}\}
\]

\[
F = \text{ave. cons.} \quad \Psi = \text{ML}
\]

\[
\hat{S} = \left( \frac{1}{M} \sum_{m=1}^{M} y_{\text{ave},m}^2 \right)^{-1}
\]

\[
(MS)^{-1} \hat{S} \sim \text{Inv} - \chi^2(M)
\]

\[
\mathbb{E} \left[ \frac{\hat{S}}{S} \right] = \frac{M}{M - 2}
\]

\[
\text{var} \left( \frac{\hat{S} - S}{S} \right) \approx \frac{2}{M}
\]

Results: (1 / 2) (independent of \( \mu \) and \( \sigma^2 \))

\( \hat{S} \)
Continuous distributions

Case 1: \((p\ \text{Gaussian}) + (F = \text{average}) + (\Psi = \text{ML})\)

\[ p = \mathcal{N}(\mu, \sigma^2) \]

Results: (2 / 2)

- \( (\hat{S})^{-1} = \overline{S}^{-1} \) and \( \overline{S}^{-1} \) is MVUE for \( S^{-1} \)

- for generic regular \( p(\cdot), \ S \uparrow \Rightarrow \frac{1}{\hat{S}} \sum y_i \xrightarrow{\text{dist.}} \mathcal{N}(0, \frac{1}{\hat{S}}) \)

implication: performances tend to become independent of \( p(\cdot) \)
Case 2: \((p \text{ continuous}) + (F = \text{max}) + (\Psi = \text{ML})\)
Case 2: \((p \text{ continuous}) + (F = \text{max}) + (\Psi = \text{ML})\)

 absolutes continuous distribution

\[
\{y_{1,m}\} \quad \{y_{2,m}\} \quad \vdots \quad \{y_{S,m}\}
\]

\(F = \text{max cons.}\) \quad \Psi = \text{ML} \quad \hat{S}

**Results:**  \textit{independent of } \(p(\cdot)\)

- \(\hat{S} = \left(\frac{1}{M} \sum_{m=1}^{M} - \log(P[y_{\text{ave},m}])\right)^{-1} (MS)^{-1} \hat{S} \sim \text{Inv} - \Gamma(M, 1)\)

- \(\mathbb{E} \left[\frac{\hat{S}}{S}\right] = \frac{M}{M - 1}\quad \text{var} \left(\frac{\hat{S} - S}{S}\right) \approx \frac{1}{M} \quad (\times \frac{1}{2} \text{ w.r.t. average})\)

- \((\hat{S})^{-1} = \hat{S}^{-1}\quad \text{and}\quad \hat{S}^{-1}\text{ is MVUE for } S^{-1}\)
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

is it possible to do better using discrete distributions?
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

\( (\text{abs. cont. dist.}) \)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\( (S_1) \)

\( \mathcal{L} \quad \mathcal{U} \quad \mathcal{N} \)

is it possible to do better using discrete distributions?
Continuous distributions

A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\( \mathcal{L} \quad \mathcal{U} \quad \mathcal{N} \)

\( (S_2) \)
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\( (S_3) \)

\[ \mathcal{L} \quad \mathcal{U} \quad \mathcal{N} \]
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\[ J(p, F = \text{ave}, \Psi = \text{ML}) \]

varagnolo@dei.unipd.it (DEI - UniPD) Distributed size estimation 24th October 2011 23 / 47
Continuous distributions

A graphical summary

\[ J(p, F = \{\text{ave.}, \max\}, \Psi) \]

(\text{abs. cont. dist.})

\[ J(p, F = \max, \Psi = \text{ML}) \]

\[ J(p, F = \text{ave}, \Psi = \text{ML}) \]

\( S_3 \)

\[ p \]

\[ \psi \]

\[ \mathcal{L} \quad \mathcal{U} \quad \mathcal{N} \]
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]  
\[ (S_3) \]

\[ J(p, F = \text{ave}, \Psi = \text{ML}) \]  
\[ (S_1) \]

is it possible to do better using discrete distributions?
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\[ J(p, F = \text{ave}, \Psi = \text{ML}) \]

\( \mathcal{L} \quad \mathcal{U} \quad \mathcal{N} \)

varagnolo@dei.unipd.it (DEI - UniPD)

Distributed size estimation

24th October 2011 23 / 47
Continuous distributions

A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

\[ J(p, F = \text{max}, \Psi = \text{ML}) \]

\[ J(p, F = \text{ave}, \Psi = \text{ML}) \]

\( S_3 \)

Distributed size estimation

varagnolo@dei.unipd.it (DEI - UniPD)
A graphical summary

\[ J(p, F = \{\text{ave.}, \text{max}\}, \Psi) \]

(abs. cont. dist.)

is it possible to do better using discrete distributions?
Table of Contents

1. Introduction
2. General estimation scheme
3. Continuous distributions
4. Discrete distributions
5. Robustness
Example with Bernoulli trials

**disclaimer:** finite precision will be handled later
Example with Bernoulli trials

Algorithm \((M = 1)\):
Example with Bernoulli trials

Algorithm ($M = 1$):

\[
y_1 \sim B(0.5)
\]

\[
y_2 \sim B(0.5)
\]

\[
y_3 \sim B(0.5)
\]

\[
y_4 \sim B(0.5)
\]

\[
y_5 \sim B(0.5)
\]

Local generation with $p = B(0.5)$
Example with Bernoulli trials

Algorithm ($M = 1$):

local generation with $p = \mathcal{B}(0.5)$
Example with Bernoulli trials

Algorithm \((M = 1)\):

- **local generation with** \(p = B(0.5)\)

\[
y_1 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i
\]

\[
y_2 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i
\]

\[
y_3 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i
\]

\[
y_4 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i
\]

\[
y_5 \rightarrow \frac{1}{S} \sum_{i=1}^{S} y_i
\]

\(F = \text{average consensus}\)
Example with Bernoulli trials

Algorithm ($M = 1$):

- Local generation with $p = \mathcal{B}(0.5)$

$$F = \text{average consensus}$$
Example with Bernoulli trials

Algorithm ($M = 1$):

1. **Local generation** with $p = B(0.5)$

2. **$F = average consensus$**

**idea:** estimator $\hat{S} = denominator$!
Example with Bernoulli trials - insights

\[ y_{\text{ave}} = \frac{2}{5} \]

\[ y_{\text{ave}} = \frac{2}{5} \]

\[ y_{\text{ave}} = \frac{2}{5} \]
Example with Bernoulli trials - insights

\[ y_{\text{ave}} = \frac{2}{6} \]

\[ y_{\text{ave}} = \frac{2}{6} \]

\[ y_{\text{ave}} = \frac{2}{6} \]

\[ y_{\text{ave}} = \frac{2}{6} \]
Example with Bernoulli trials - insights

\[ y_{\text{ave}} = \frac{1}{3} = \frac{2}{6} = \ldots \]

\[ y_{\text{ave}} = \frac{1}{3} = \frac{2}{6} = \ldots \]

\[ y_{\text{ave}} = \frac{1}{3} = \frac{2}{6} = \ldots \]

\[ y_{\text{ave}} = \frac{1}{3} = \frac{2}{6} = \ldots \]

\[ y_{\text{ave}} = \frac{1}{3} = \frac{2}{6} = \ldots \]
Example with Bernoulli trials - insights

$y_{ave} = \frac{1}{3}$

assumption: agents compute only coprime representations
Example with Bernoulli trials - insights

\[ y_{ave} = \frac{1}{3} \]

assumption: agents compute only coprime representations

is denominator a good estimator?
Proposition

Hypotheses:

\( y_i \sim \mathcal{B}(p) \)

\[ y_{\text{ave}} = \frac{1}{S} \sum_{i=1}^{S} y_i = \frac{k}{S} \text{ coprime} \]
Statistical characterization of the estimator

Proposition

Hypotheses:
- \( y_i \sim \mathcal{B}(p) \)
- \( y_{\text{ave}} = \frac{1}{S} \sum_{i=1}^{S} y_i = \frac{k}{\hat{S}} \) \textit{coprime}

Thesis:
\( \hat{S} = \text{ML estimate of } S \) for every \( p \)
Ockham’s razor  
(William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”
Intuition behind the ML property

Ockham’s razor  
(William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”

\[ y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} = \frac{2\hat{k}}{2\hat{S}} = \frac{3\hat{k}}{3\hat{S}} = \ldots \]
Intuition behind the ML property

Ockham’s razor

(William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”

\[ y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} = \frac{2\hat{k}}{2\hat{S}} = \frac{3\hat{k}}{3\hat{S}} = \ldots \]

\( \hat{S} \) agents, \( \hat{k} \) generated “1”
Ockham’s razor  (William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”

\[
\hat{\text{y}}_{\text{ave}} = \frac{\hat{k}}{\hat{S}} = \frac{2\hat{k}}{2\hat{S}} = \frac{3\hat{k}}{3\hat{S}} = \ldots
\]

2\hat{S} agents, 2\hat{k} generated “1”
Intuition behind the ML property

Ockham’s razor

(William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”

\[
y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} = \frac{2\hat{k}}{2\hat{S}} = \frac{3\hat{k}}{3\hat{S}} = \ldots
\]

- \(3\hat{S}\) agents, \(3\hat{k}\) generated “1”
Intuition behind the ML property

Ockham’s razor

(William of Ockham, c. 1288 - c. 1348)

“select from among competing hypotheses the one that makes the fewest new assumptions”

\[
y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} = \frac{2\hat{k}}{2\hat{S}} = \frac{3\hat{k}}{3\hat{S}} = \ldots
\]

\[\uparrow\]

the simplest network / hypothesis
An historical and related question

The Newton-Pepys problem (Isaac Newton, 1643 - 1727; Samuel Pepys, 1633 - 1703)

Which one is the most likely event?
1. have at least 1 six when rolling 6 dice
2. have at least 2 sixes when rolling 12 dice
3. have at least 3 sixes when rolling 18 dice

Our result:

\[ P \left[ \text{have exactly } k \text{ sixes when rolling } kN \text{ dice} \right] \]

decreases when increasing \( k \)
Essential question: performances?

Recap

\[ \text{measured } y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} \text{ coprime, estimator } = \hat{S} \]
Essential question: performances?

measured $y_{\text{ave}} = \frac{\hat{k}}{\hat{S}}$ coprime, estimator $= \hat{S}$

is this a good estimator?
Essential question: performances?

Recap:

\[ \text{measured } y_{\text{ave}} = \frac{\hat{k}}{\hat{S}} \text{ coprime, estimator } = \hat{S} \]

Will develop intuitions

Is this a good estimator?
The nonlinear behavior of the estimator

assumption:
$S$ known,
$S = 6$
The nonlinear behavior of the estimator

assumption:
$S$ known, $S = 6$
The nonlinear behavior of the estimator

assumption: $S$ known, $S = 6$
The nonlinear behavior of the estimator

assumption:
$S$ known,
$S = 7$
Connections with number theory

Definition: totative of an integer $S$

A positive integer $k \leq S$ which is also relatively prime to $S$
Connections with number theory

Definition: **totative of an integer** $S$

A positive integer $k \leq S$ which is also relatively prime to $S$

Definition: **Euler’s $\phi$-function**

$$\phi(S) := \text{number of totatives of } S$$
Connections with number theory

Definition: totative of an integer $S$

A positive integer $k \leq S$ which is also relatively prime to $S$

Definition: Euler’s $\phi$-function

$\phi(S) := \text{number of totatives of } S$

For our purposes, $\phi(S) = \text{number of good values}$
Totatives’ characteristics (1/2)

**Distribution:** $\approx$ uniform on $\mathbb{N}$

- $S = 10$: $\ldots$ (40%)
- $S = 50$: $\ldots$ (40%)
- $S = 100$: $\ldots$ (40%)
Totatives’ characteristics (1/2)

Distribution: \( \approx \text{uniform on } \mathbb{N} \)

- \( S = 10: \)
  - Range: 0 to 10
  - Percentage: 40%

- \( S = 50: \)
  - Range: 0 to 50
  - Percentage: 40%

- \( S = 100: \)
  - Range: 0 to 100
  - Percentage: 40%

very important: Bernoulli’s \( p \) has not key roles
Totatives’ characteristics (1/2)

Distribution: \( \approx \text{uniform on } \mathbb{N} \)

<table>
<thead>
<tr>
<th>( S ) = 10:</th>
<th>( S = 50: )</th>
<th>( S = 100: )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>100</td>
</tr>
</tbody>
</table>

very important: Bernoulli’s \( p \) has not key roles
Totatives’ characteristics (1/2)

Distribution: $\approx$ uniform on $\mathbb{N}$

- $S = 10$: $0$ to $10$ (40%)
- $S = 50$: $0$ to $50$ (40%)
- $S = 100$: $0$ to $100$ (40%)

very important: Bernoulli’s $p$ has not key roles
Totatives’ characteristics (1/2)

Distribution: \( \approx \) uniform on \( \mathbb{N} \)

\[
\begin{align*}
S = 10: & \quad 0 \quad 10 \quad (40\%) \\
S = 50: & \quad 0 \quad 50 \quad (40\%) \\
S = 100: & \quad 0 \quad 100 \quad (40\%)
\end{align*}
\]

very important: Bernoulli’s \( p \) has not key roles
Totatives’ characteristics (2/2)

How many?

\[ \phi(S) > \frac{S}{e^\gamma \log \log S + \frac{3}{\log \log S}} \quad \text{i.e.} \quad \frac{\phi(S)}{S} > 0.15 \]

\[ \forall S \in [2, 10^{10}] \]

\((\gamma \approx 0.577, \text{ Euler-Mascheroni constant})\)
How many?

\[ \phi(S) > \frac{S}{e^\gamma \log \log S + \frac{3}{\log \log S}} \quad \text{i.e.} \quad \frac{\phi(S)}{S} > 0.15 \]

\( \forall S \in [2, 10^{10}] \)

(\( \gamma \approx 0.577 \), Euler-Mascheroni constant)

an other important result:

at least 15\% of the plausible \( y_{\text{ave}} \) are good ones
Totatives’ characteristics (2/2)

How many?

\[ \phi(S) > \frac{S}{e^\gamma \log \log S + \frac{3}{\log \log S}} \]

\[ \frac{\phi(S)}{S} > 0.15 \]

\[ \forall S \in [2, 10^{10}] \]

(\( \gamma \approx 0.577 \), Euler-Mascheroni constant)

another important result:

at least 15\% of the plausible \( y_{ave} \) are good ones

only 15\%??
Extension to the multiple-generations case

\[ y_1: \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \]

\[ y_2: \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \]

\[ y_3: \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \]

\[ y_4: \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \]

\[ y_5: \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \]
Extension to the multiple-generations case

\[
\begin{align*}
y_1 &: 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \\
y_2 &: 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \\
y_3 &: 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \\
y_4 &: 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \\
y_5 &: 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0
\end{align*}
\]

locally generated
(size = \( M \))
Extension to the multiple-generations case

\[\begin{align*}
y_1: & \quad 0 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \\
y_2: & \quad 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \\
y_3: & \quad 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \\
y_4: & \quad 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \\
y_5: & \quad 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \\
\end{align*}\]

component-wise consensus
Extension to the multiple-generations case

<table>
<thead>
<tr>
<th></th>
<th>( y_1 )</th>
<th>( y_2 )</th>
<th>( y_3 )</th>
<th>( y_4 )</th>
<th>( y_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0 1 0 1 1 0 1 0 0 0</td>
<td>1 1 1 0 0 1 1 1 0 1</td>
<td>0 0 1 0 0 0 0 1 1 0</td>
<td>1 1 0 0 1 1 1 1 1 1</td>
<td>0 0 1 1 1 0 0 1 0 0</td>
</tr>
</tbody>
</table>

\( S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10} \)
Extension to the multiple-generations case

\[
\begin{align*}
\hat{S}_1 & : 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\
\hat{S}_2 & : 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\
\hat{S}_3 & : 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
\hat{S}_4 & : 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hat{S}_5 & : 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
\end{align*}
\]

\[\hat{s} = \text{LCM} \left( \left\{ \hat{S}_m \right\} \right)\]
Intuition behind the LCM(·) operation
Intuition behind the LCM(\cdot) operation
Intuition behind the LCM(·) operation
Intuition behind the LCM(·) operation

\[ \text{LCM}(2, 3) = 6 \]
Discrete distributions

Estimation performance

Main result

\[(0.5)^{S_{\max}}M \leq \mathbb{P} \left[ \hat{S} \neq S \mid M \right] \leq (0.85)^M\]
Table of Contents

1 Introduction
2 General estimation scheme
3 Continuous distributions
4 Discrete distributions
5 Robustness
Robustness issues

need to take into account several non-idealities

- quantization errors
- consensus errors

robustness properties of the various strategies are very different
Robustness: Gaussian + average

Assumptions and definitions

1. $y_{\text{ave}}^{\text{actual}} = (1 + \delta)y_{\text{ave}}^{\text{ideal}} + \Delta$

2. $\frac{\Delta \hat{S}}{\hat{S}} := \text{relative error btw. ideal case and actual estimate}$
Robustness: Gaussian + average

Assumptions and definitions

- \( y_{\text{ave}}^{\text{actual}} = (1 + \delta)y_{\text{ave}}^{\text{ideal}} + \Delta \)

- \( \frac{\Delta \hat{S}}{\hat{S}} \) := relative error btw. *ideal case* and *actual estimate*

First-order approximation

\[
\left| \frac{\Delta \hat{S}}{\hat{S}} \right| \lesssim 2\delta_{\text{max}} + 2\sqrt{S}\Delta_{\text{max}}
\]
Robustness: Gaussian + average

Assumptions and definitions

- \( y_{\text{ave}}^{\text{actual}} = (1 + \delta) y_{\text{ave}}^{\text{ideal}} + \Delta \)

- \( \hat{\Delta S} \) := relative error btw. ideal case and actual estimate

First-order approximation

\[
\left| \frac{\hat{\Delta S}}{\hat{S}} \right| \lesssim 2\delta_{\text{max}} + 2\sqrt{S} \Delta_{\text{max}}
\]

well posed map
Robustness: absolutely continuous dist. + max

Assumptions and definitions

- \( y_{\text{actual}}^{\text{ave}} = (1 + \delta) y_{\text{ideal}}^{\text{ave}} + \Delta \)

- \( \frac{\Delta \hat{S}}{\hat{S}} := \text{relative error btw. ideal case and actual estimate} \)

First-order approximation

\[
\left| \frac{\Delta \hat{S}}{\hat{S}} \right| \lesssim S \delta_{\text{max}} + S \Delta_{\text{max}}
\]
Robustness: absolutely continuous dist. + max

Assumptions and definitions

\[ y_{\text{actual}}^{\text{ave}} = (1 + \delta)y_{\text{ideal}}^{\text{ave}} + \Delta \]

\[ \frac{\Delta \hat{S}}{\hat{S}} := \text{relative error btw. ideal case and actual estimate} \]

First-order approximation

\[ \left| \frac{\Delta \hat{S}}{\hat{S}} \right| \lesssim S\delta_{\text{max}} + S\Delta_{\text{max}} \]

\textit{tradeoff robustness vs. performance}
Robustness: Bernoulli + average

Extremely non-linear map (requires $S_{\text{max}}$):

\[ y_{\text{ave}} \]

\[ S \]

\[ \tilde{S} \]
Robustness: Bernoulli + average

Extremely non-linear map (requires $S_{\text{max}}$):
Robustness: Bernoulli + average

Extremely non-linear map \((\text{requires } S_{\text{max}})\):

\[
\begin{array}{cccccccccccc}
0 & 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 & 1 \\
\hat{S} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

small error ⇒ insensitivity
big error ⇒ unreliable estimates

\textit{ill posed map}
Robustness: Bernoulli + average

Extremely non-linear map (requires $S_{\text{max}}$):

\[
\hat{S} \propto \frac{1}{S_{\text{max}}^2}
\]

minimal distance between stems

$\hat{S}$

$y_{\text{ave}}$

0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

0 5 10 15 20

varagnolo@dei.unipd.it (DEI - UniPD)
Summary of discussed points

- proposed various easily implementable distributed estimators
- mathematically characterized their statistical properties
- shown tradeoffs between estimation error performances and robustness to errors
Concluding comments (1/2)

Summary of discussed points

- proposed various easily implementable distributed estimators
- mathematically characterized their statistical properties
- shown tradeoffs between estimation error performances and robustness to errors

Summary of novel contributes

- full statistical descriptions of the estimators
- independence of performances on generation distributions
- novel Bernoulli-based estimator with exponential performance
Concluding comments (2/2)

Future works

- extensions to dynamic networks
- applications to network topology estimation
  - generate some data (locally)
  - transform them (distributedly)
  - compute hypotheses’ likelihood (locally)
develop algorithms able to detect network faults and give indications for self-reconfiguration purposes
Bibliography

Varagnolo, Pillonetto, Schenato (2010)
Distributed statistical estimation of the number of nodes in Sensor Networks
IEEE Conference on Decision and Control

Varagnolo, Pillonetto, Schenato (2012)
Consensus based estimation of anonymous networks size using Bernoulli trials
American Control Conference (submitted)

Varagnolo, Pillonetto, Schenato (20??)
Distributed size estimation in anonymous networks
IEEE Transactions on Automatic Control (submitted)
Distributed size estimation in anonymous networks

Damiano Varagnolo, Gianluigi Pillonetto, Luca Schenato

Department of Information Engineering, University of Padova

24th October 2011

varagnolo@dei.unipd.it
http://automatica.dei.unipd.it/people/varagnolo.html