

Foundations of Privacy and Quantitative Information Flow

Catuscia Palamidessi
INRIA & Ecole Polytechnique
France

Plan of the lecture

- I. Quantitative Information Flow
- II. Differential Privacy
- III. Location Privacy

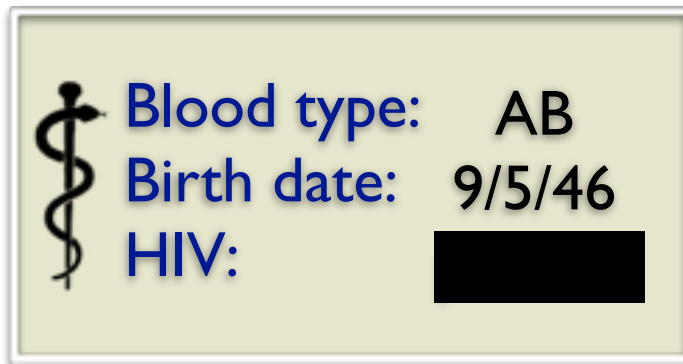
Part I

Quantitative Information Flow

1. Motivations
2. Information-theoretic view
3. Notions of entropy and operational interpretation
4. Focus on Shannon leakage and min-entropy leakage

Protection of sensitive information

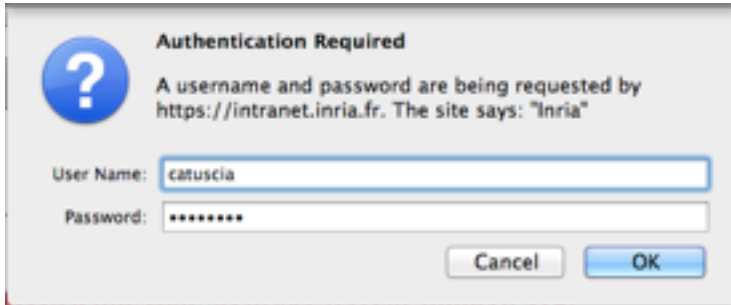
- Protecting the **confidentiality** of sensitive information is a fundamental issue in computer security and in privacy



- Access control and encryption are not sufficient! Systems could leak secret information through correlated observables.
 - The notion of “observable” depends on the situation and adversary
 - Often, secret-leaking observables are public, and therefore available to the adversary

Leakage through correlated observables

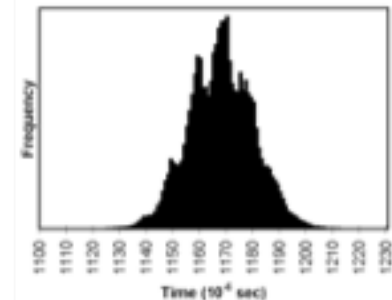
Password checking



Election tabulation



Timings of decryptions



Quantitative Information Flow

Information Flow: Leakage of **secret information** via **correlated observables**

Ideally: No leak

- No interference [Goguen & Meseguer'82]

In practice: There is almost always some leak

- Intrinsic to the system (public observables, part of the design)
- Side channels

⇒ **need quantitative ways to measure the leak**

Example I

Password checker I

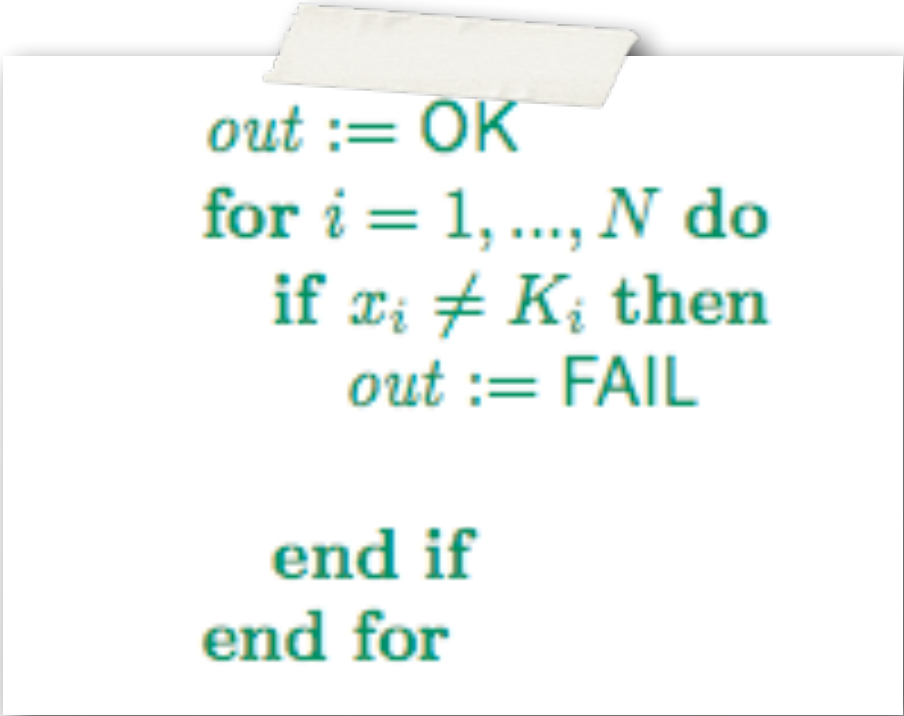
Password: $K_1 K_2 \dots K_N$

Input by the user: $x_1 x_2 \dots x_N$

Output: out (Fail or OK)

Intrinsic leakage

By learning the result of the check the adversary learns something about the secret



```
 $out := OK$   
for  $i = 1, \dots, N$  do  
  if  $x_i \neq K_i$  then  
     $out := FAIL$   
  end if  
end for
```

Example I


Password checker 2

Password: $K_1 K_2 \dots K_N$

Input by the user: $x_1 x_2 \dots x_N$

Output: out (Fail or OK)

More efficient, but what about security?



```
out := OK  
for  $i = 1, \dots, N$  do  
  if  $x_i \neq K_i$  then  
    { out := FAIL  
      exit() }  
  end if  
end for
```


Example I

Password checker 2


Password: $K_1K_2 \dots K_N$

Input by the user: $x_1x_2 \dots x_N$

Output: out (Fail or OK)

Side channel attack

If the adversary can measure the execution time, then he can also learn the longest correct prefix of the password

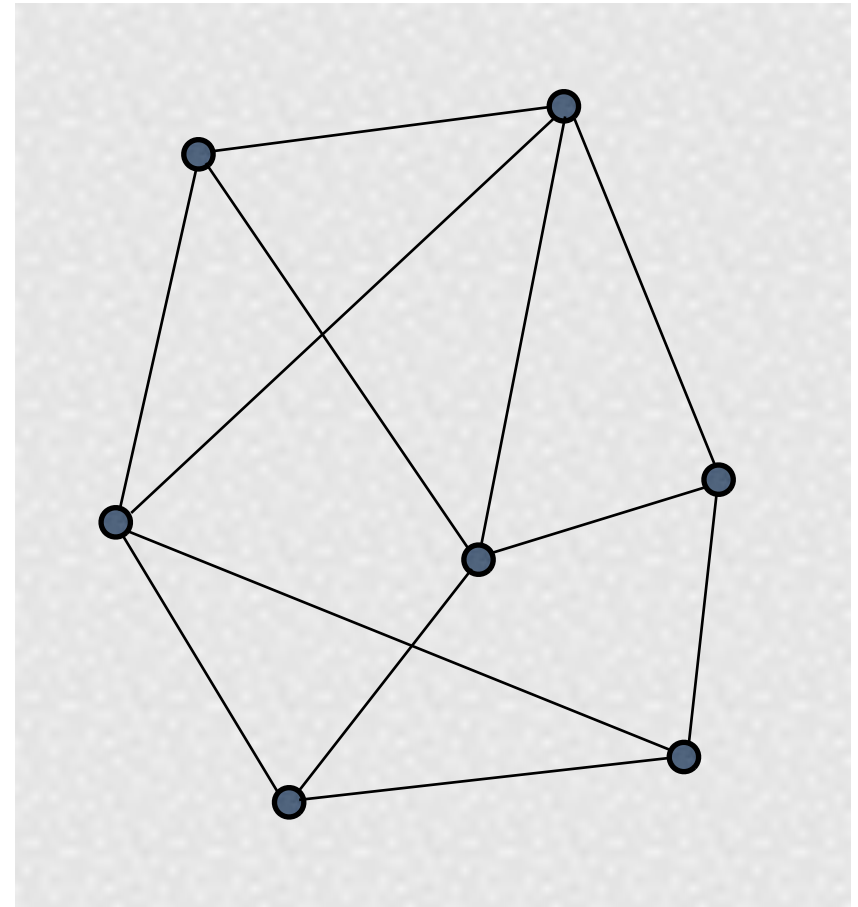


```
 $out := OK$   
for  $i = 1, \dots, N$  do  
  if  $x_i \neq K_i$  then  
    {  $out := FAIL$   
       $exit()$  }  
  end if  
end for
```

Example 2

DC Nets (Extended Dining Cryptographers) [Chaum'88]

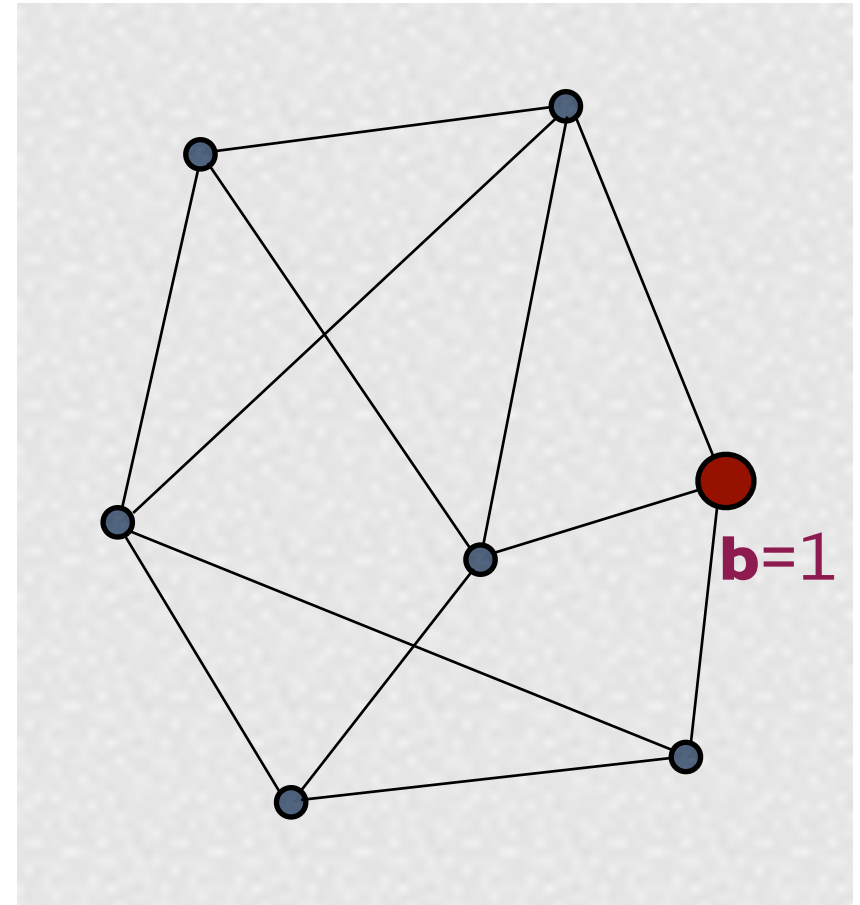
- A set of nodes with some communication channels (edges).
- One of the nodes (source) wants to broadcast one bit **b** of information
- The source (broadcaster) must remain **anonymous**



DC Nets

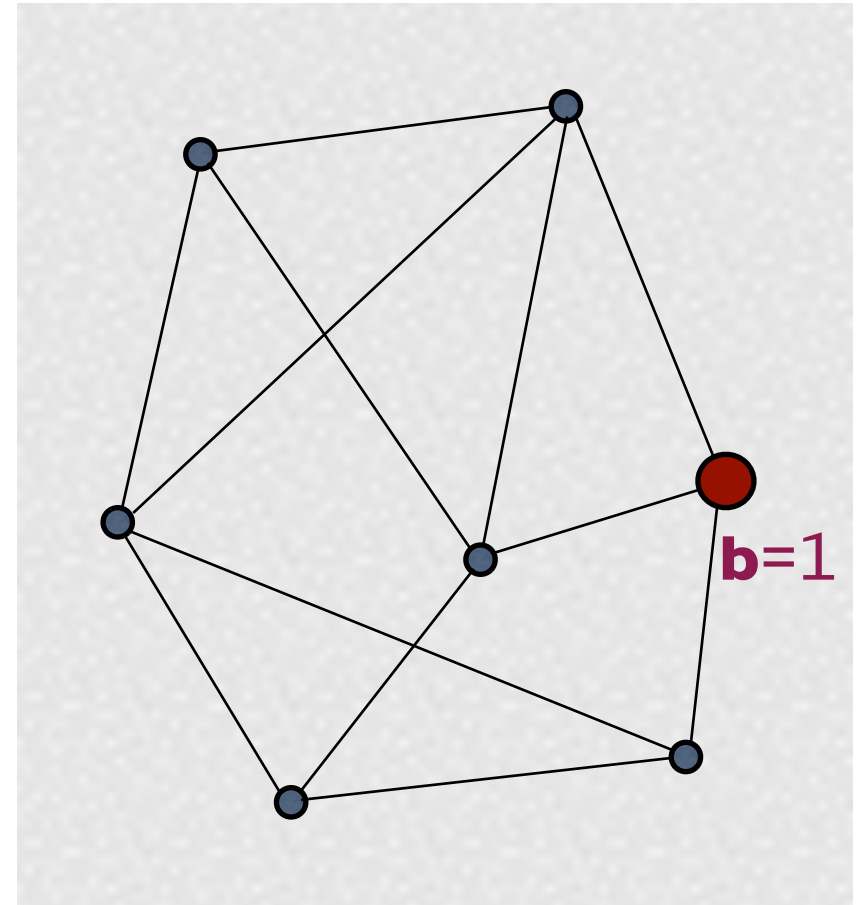
(Extended Dining Cryptographers)
[Chaum'88]

- A set of nodes with some communication channels (edges).
- One of the nodes (source) wants to broadcast one bit **b** of information
- The source (broadcaster) must remain **anonymous**



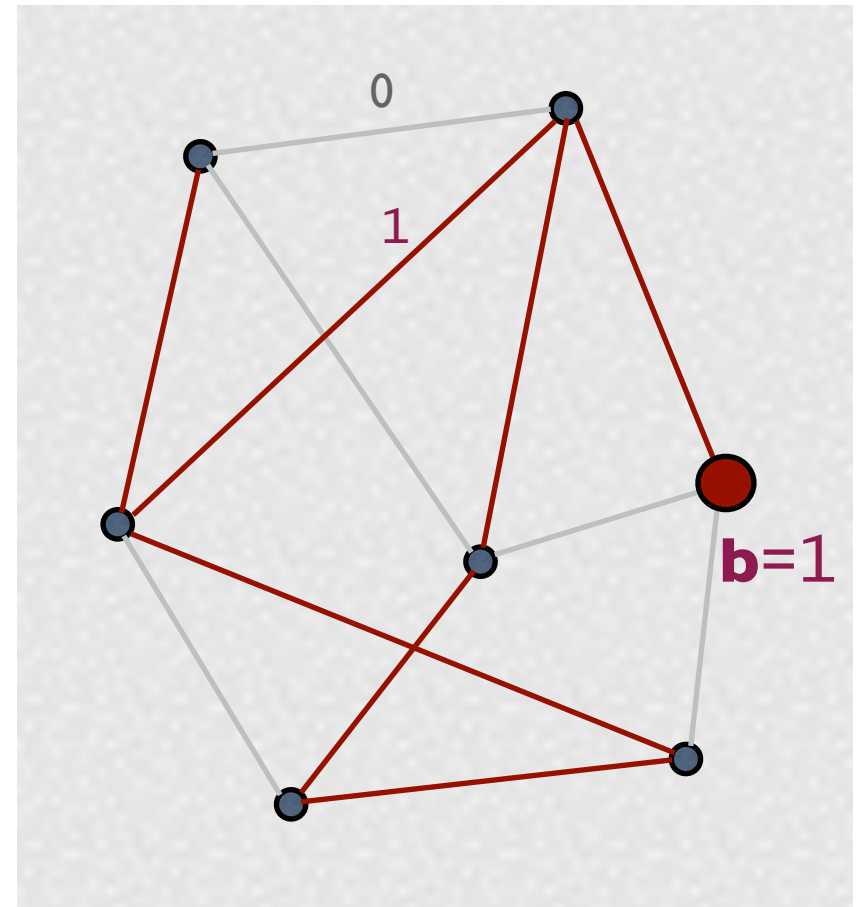
Chaum's solution

- Associate to each edge a fair binary coin



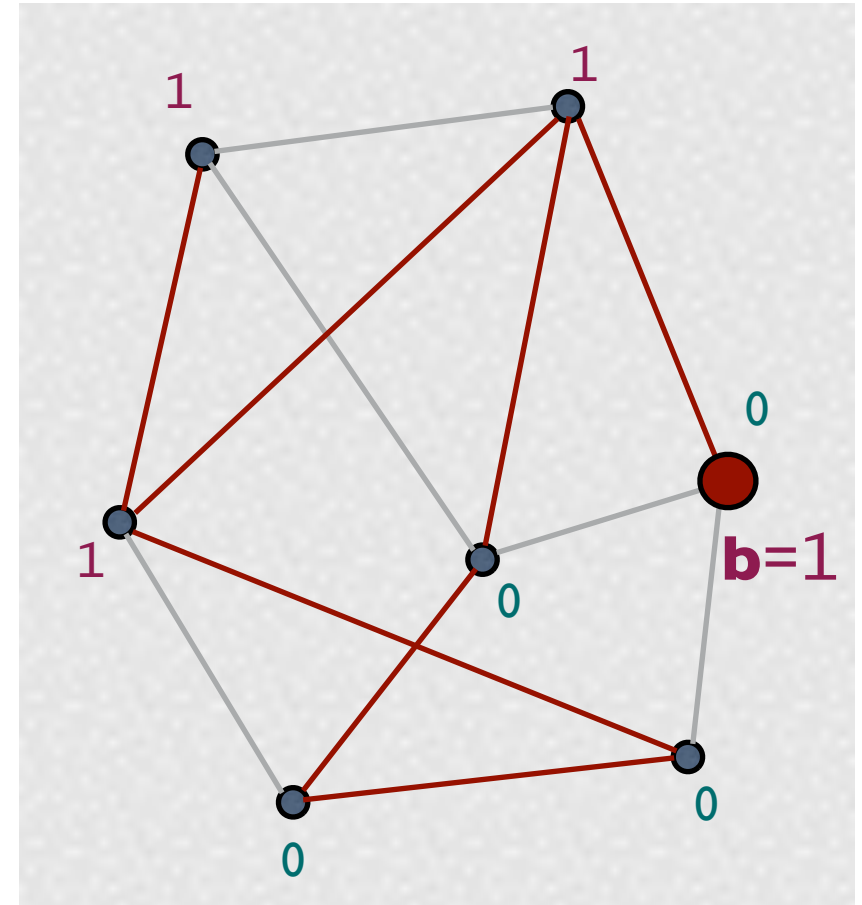
Chaum's solution

- Associate to each edge a fair binary coin
- Toss the coins



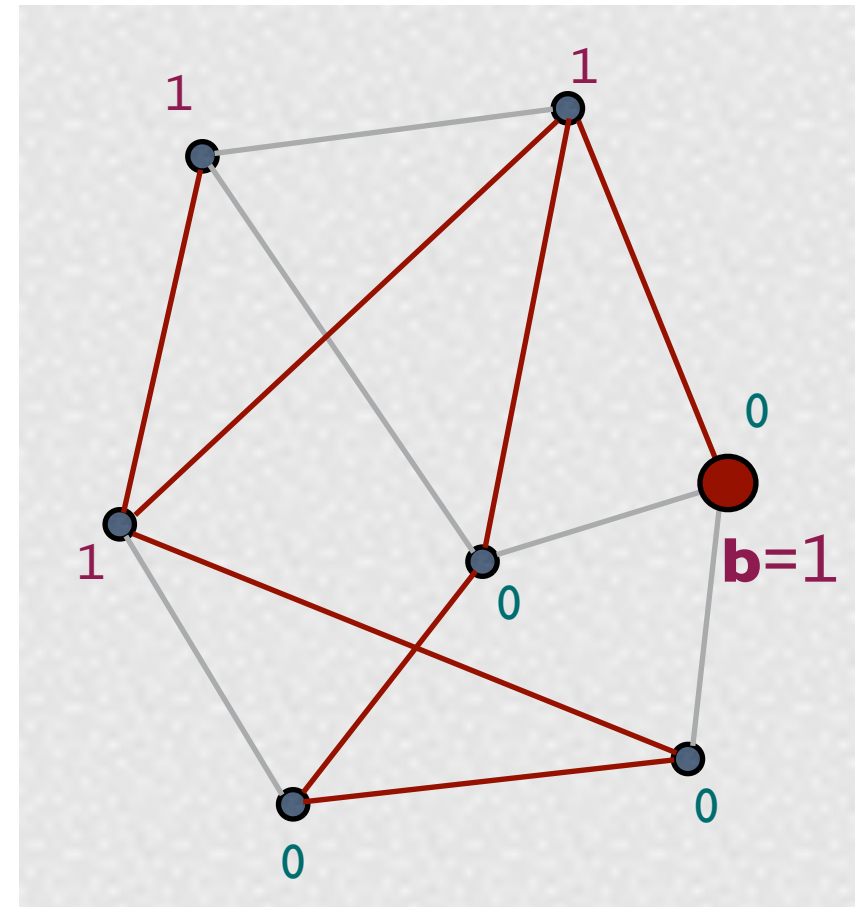
Chaum's solution

- Associate to each edge a fair binary coin
- Toss the coins
- Each node computes the binary sum of the incident edges. The source adds **b**. They all broadcast their results



Chaum's solution

- Associate to each edge a fair binary coin
- Toss the coins
- Each node computes the binary sum of the incident edges. The source adds **b**. They all broadcast their results
- **Achievement of the goal:**
Compute the total binary sum:
it coincides with **b**



Anonymity of DC Nets

Observables: An (external) attacker can only see the declarations of the nodes

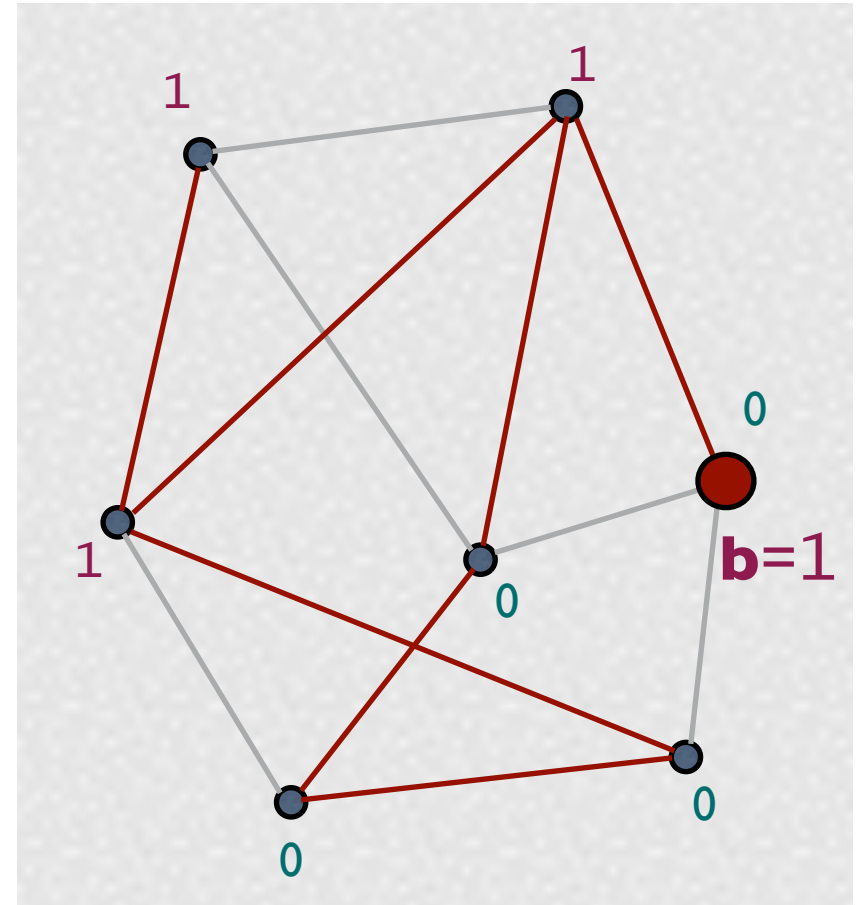
Question: Does the protocol protects the anonymity of the source?

Strong anonymity (Chaum)

- If the graph is **connected** and the coins are **fair**, then for an **external observer**, the protocol satisfies **strong anonymity**:

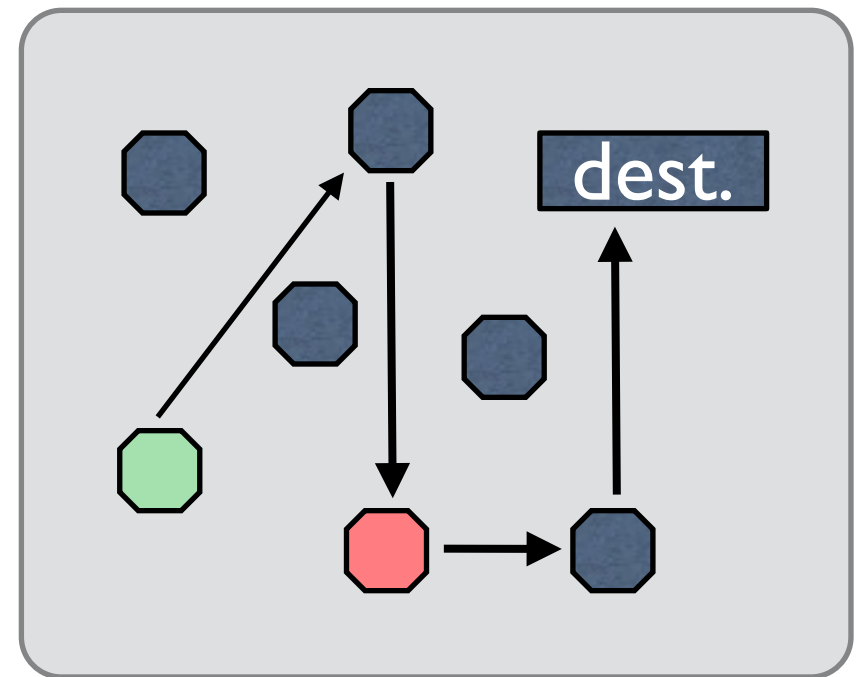
the *a posteriori* probability that a certain node is the source is equal to its *a priori* probability

- A priori / a posteriori = before / after observing the declarations



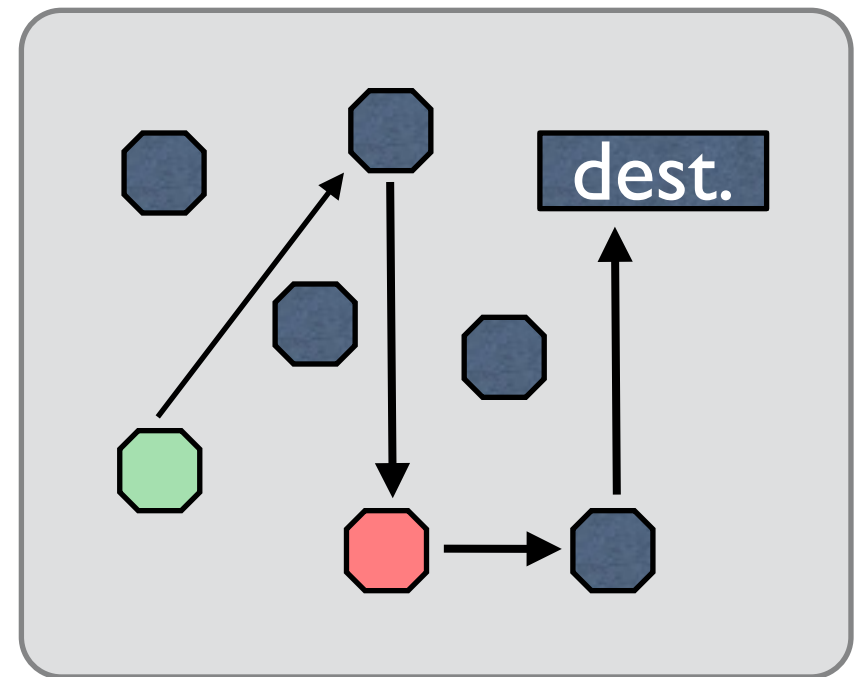
Example 3: Crowds [Rubin and Reiter'98]

- Problem: A user (initiator) wants to send a message anonymously to another user (dest.)
- Crowds: A group of n users who agree to participate in the protocol.
- The initiator selects randomly another user (forwarder) and forwards the request to her
- A forwarder randomly decides whether to send the message to another forwarder or to dest.
- ... and so on



Example 3: Crowds [Rubin and Reiter'98]

- Problem: A user (initiator) wants to send a message anonymously to another user (dest.)
- Crowds: A group of n users who agree to participate in the protocol.
- The initiator selects randomly another user (forwarder) and forwards the request to her
- A forwarder randomly decides whether to send the message to another forwarder or to dest.
- ... and so on



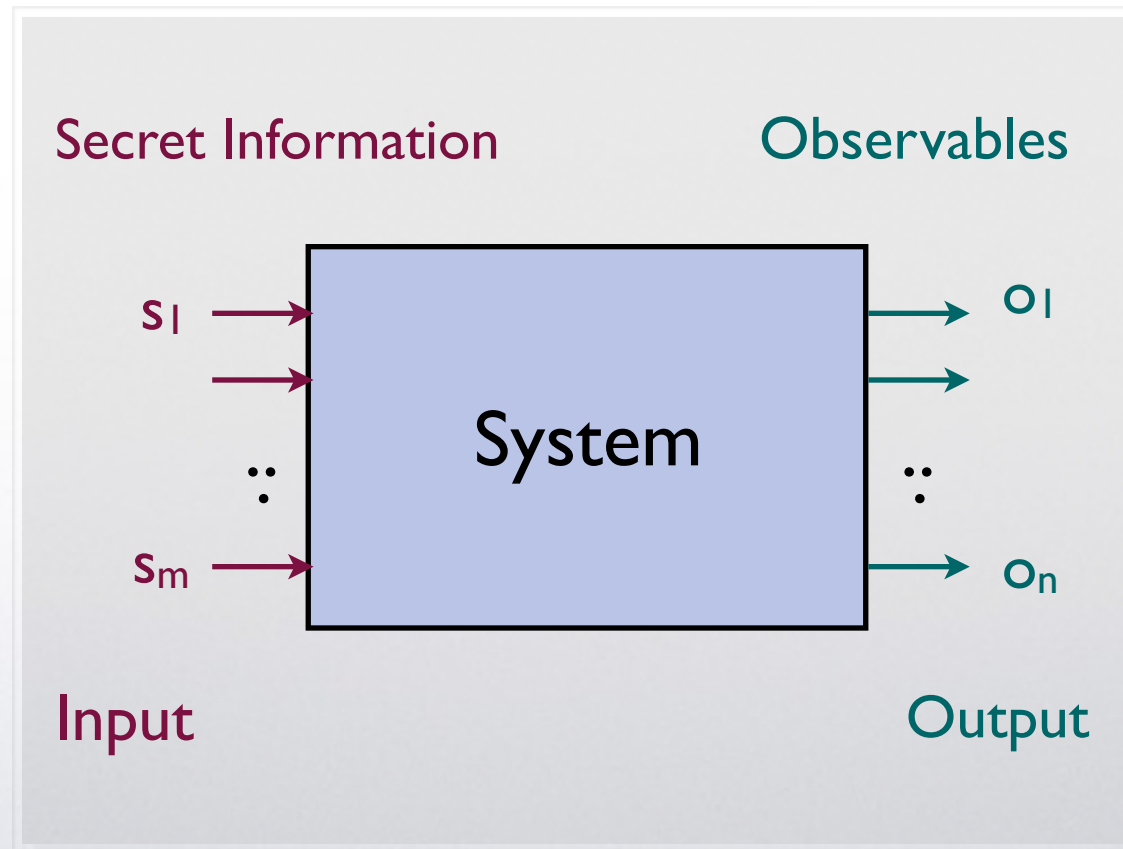
Probable innocence: under certain conditions, an attacker who intercepts the message from x cannot attribute more than 0.5 probability to x to be the initiator

Common features

- Secret information
 - Password checker: The password
 - DC: the identity of the source
 - Crowds: the identity of the initiator
- Public information (Observables)
 - Password checker: The result (OK / Fail) and the execution time
 - DC: the declarations of the nodes
 - Crowds: the identity of the agent forwarding to a corrupted user
- The system may be probabilistic
 - Often the system uses randomization to obfuscate the relation between secrets and observables
 - DC: coin tossing
 - Crowds: random forwarding to another user

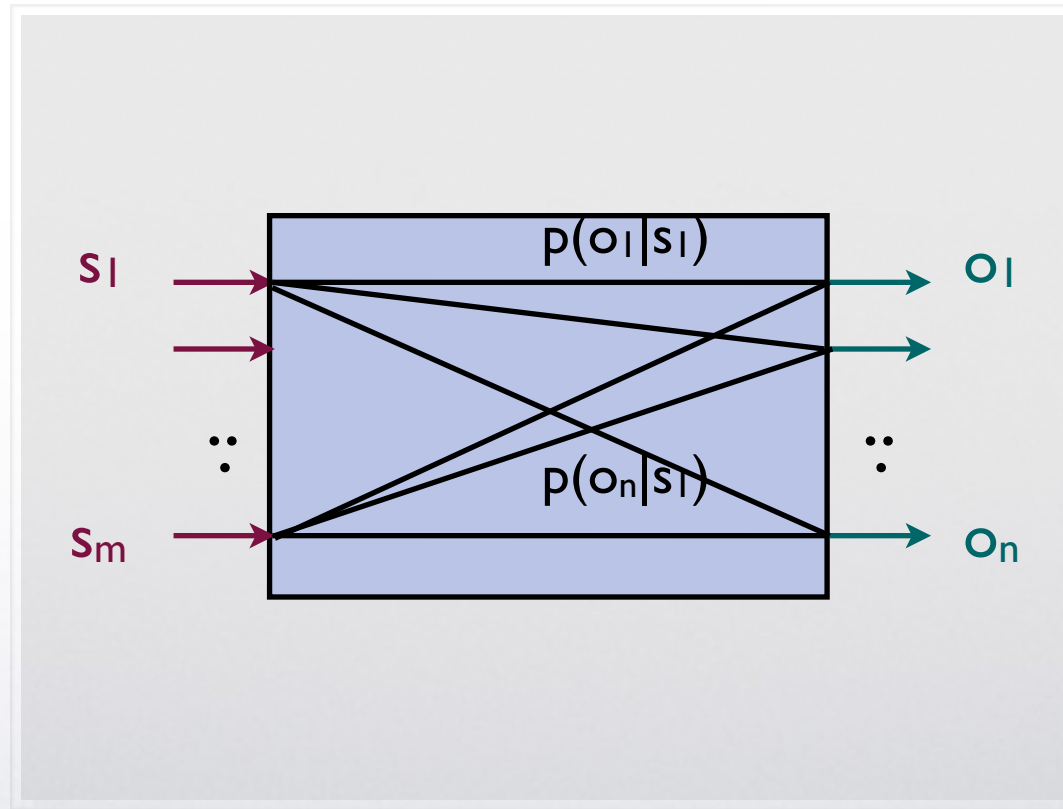
The basic model:

Systems = Information-Theoretic channels



Probabilistic systems are **noisy** channels:

an output can correspond to different inputs, and
an input can generate different outputs, according to a prob. distribution



$p(o_j|s_i)$: the conditional probability to observe o_j given the secret s_i

	O_1	...	O_n
S_1	$p(o_1 s_1)$...	$p(o_n s_1)$
\vdots	\vdots		
S_m	$p(o_1 s_m)$		$p(o_n s_m)$

$$p(o|s) = \frac{p(o \text{ and } s)}{p(s)}$$

A channel is characterized by its matrix: the array of conditional probabilities

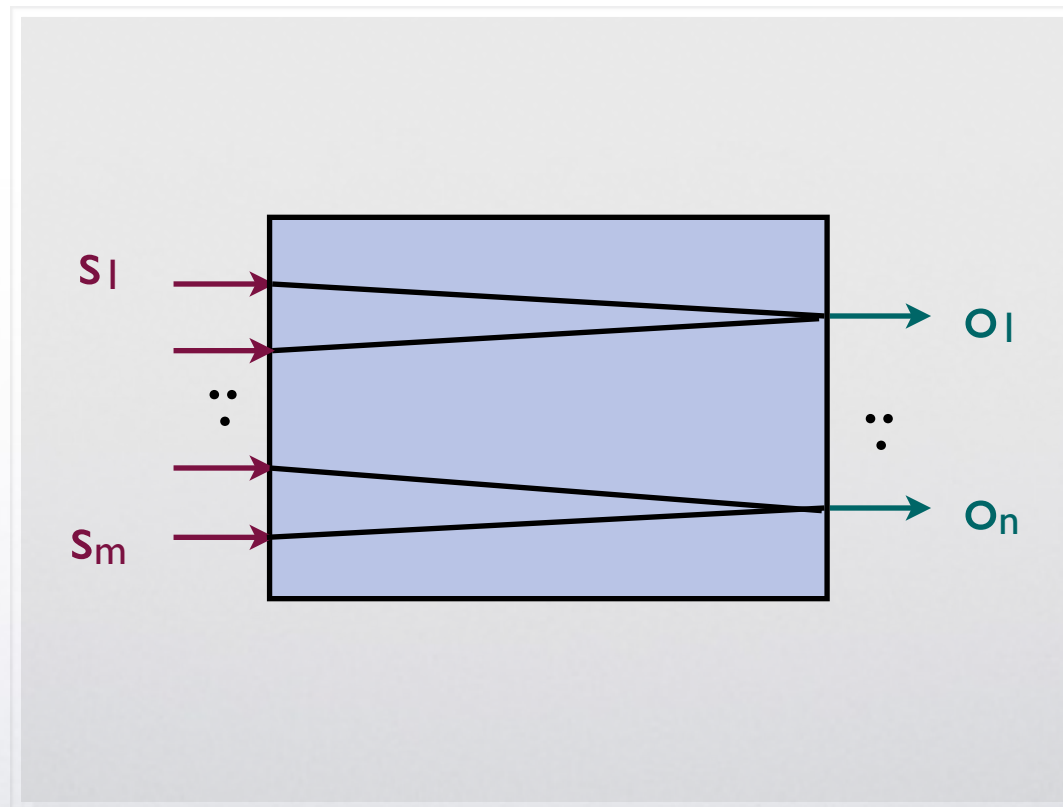
In an information-theoretic channel these conditional probabilities are independent from the input distribution

This means that we can model systems abstracting from the input distribution

Particular case: **Deterministic systems**

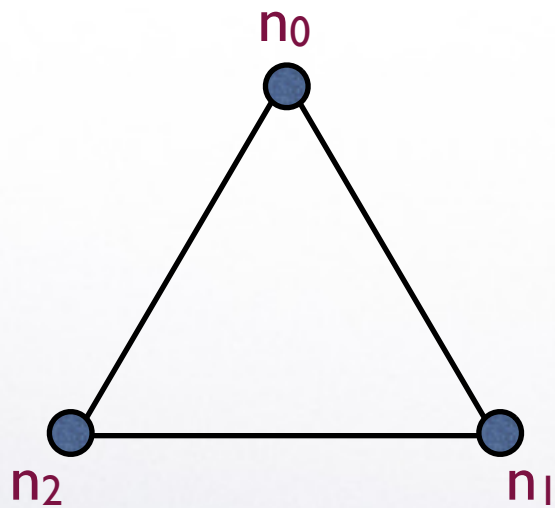
In these systems an input generates only one output

Still interesting: the problem is how to retrieve the input from the output



The entries of the channel matrix can be only 0 or 1

Example: DC nets (ring of 3 nodes, $b=1$)

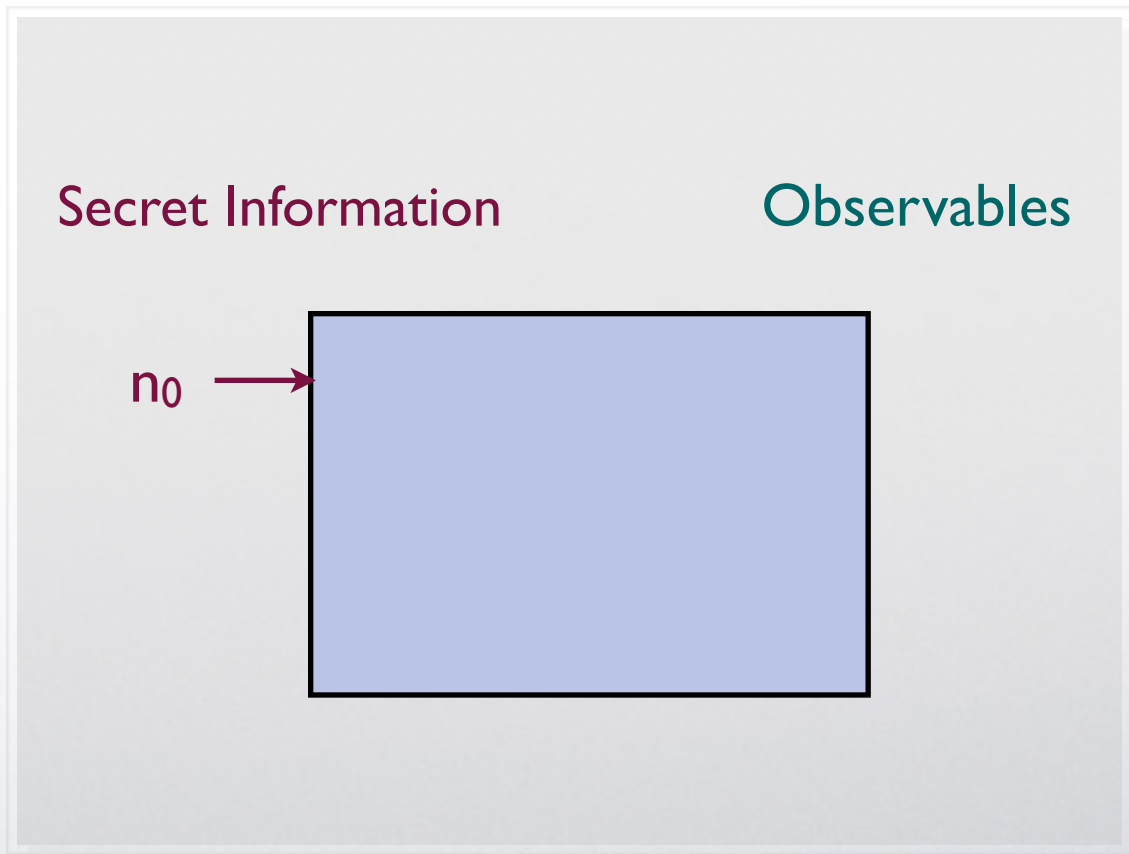
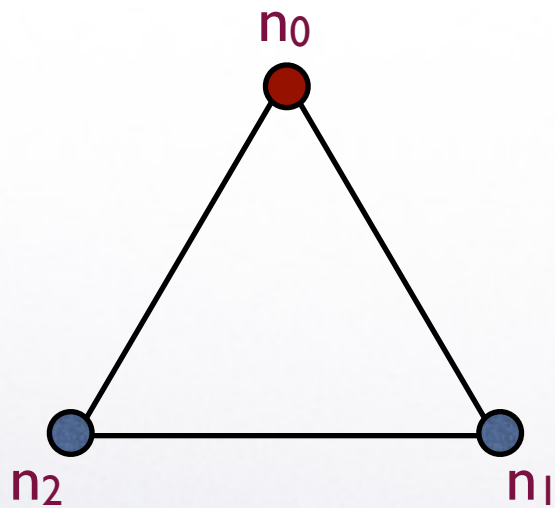


Secret Information

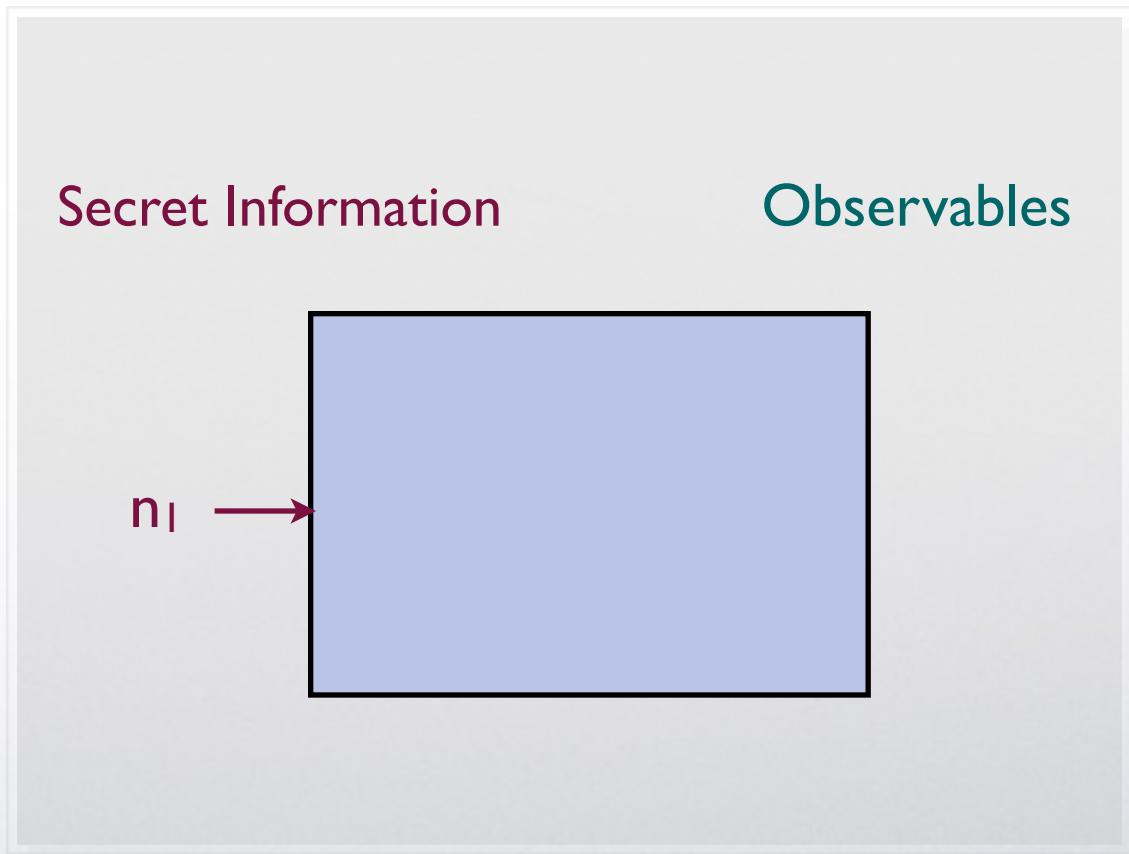
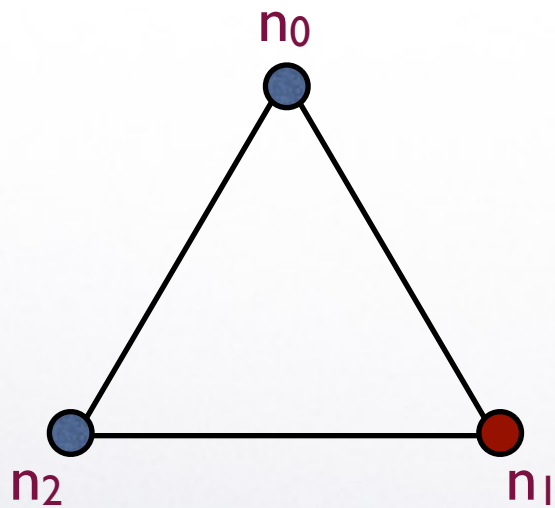
Observables



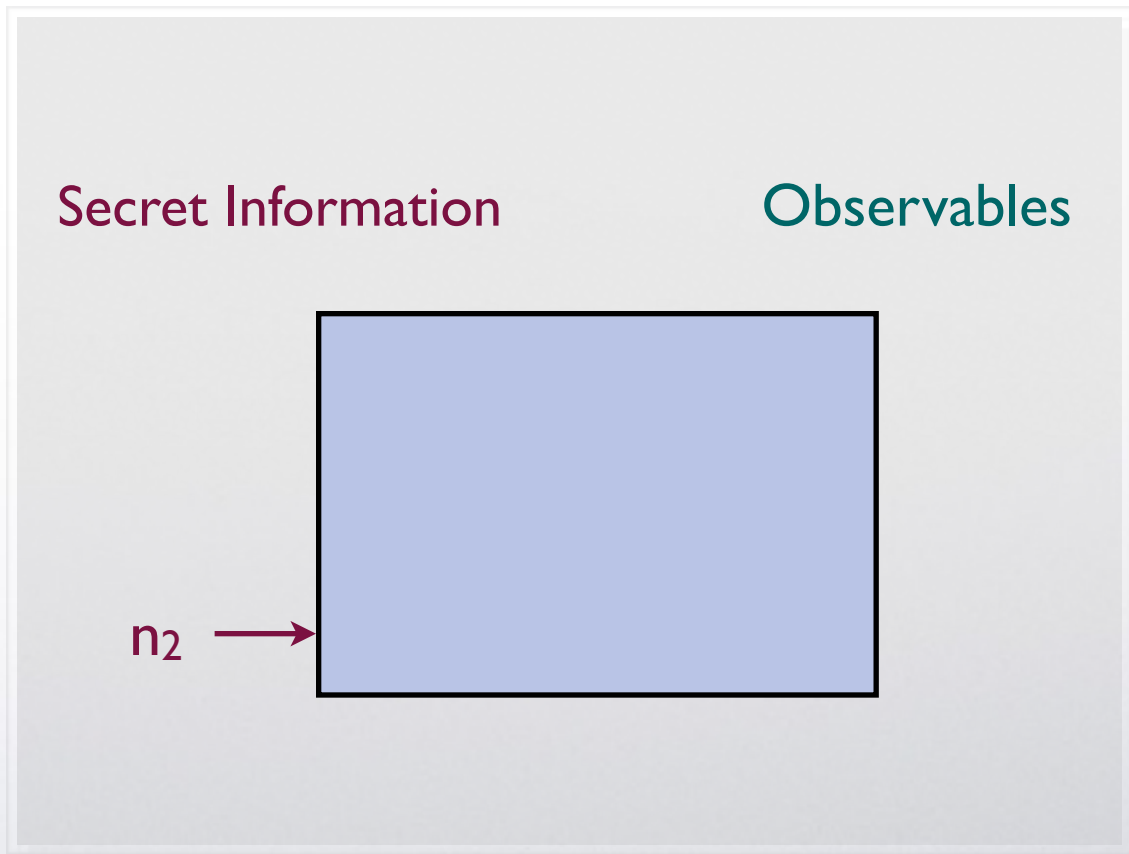
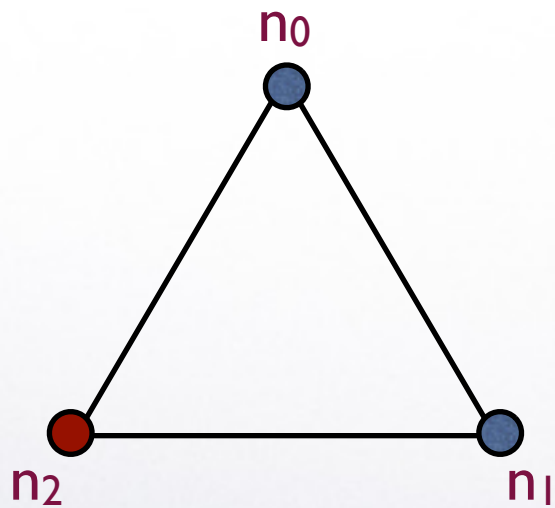
Example: DC nets (ring of 3 nodes, $b=1$)



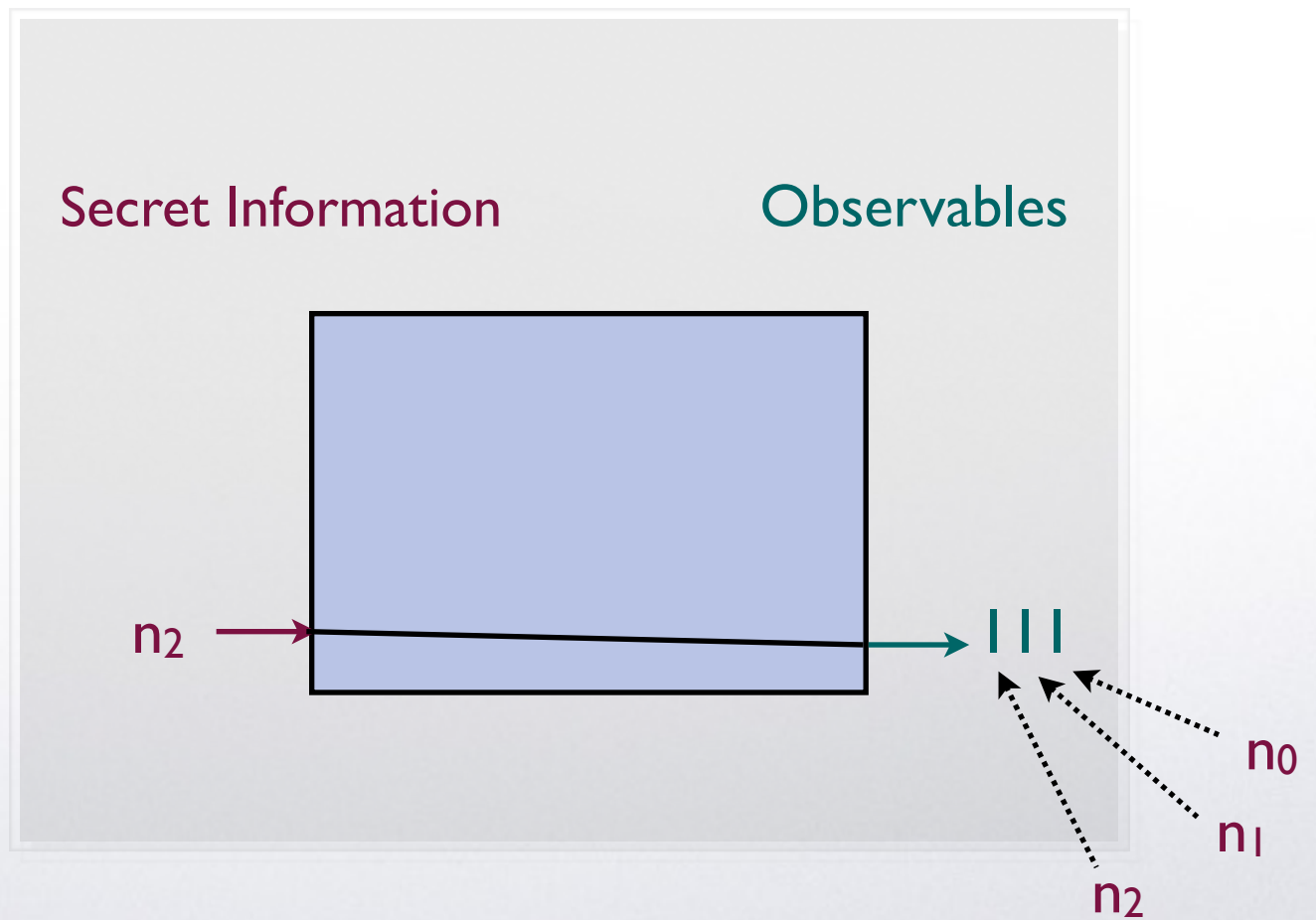
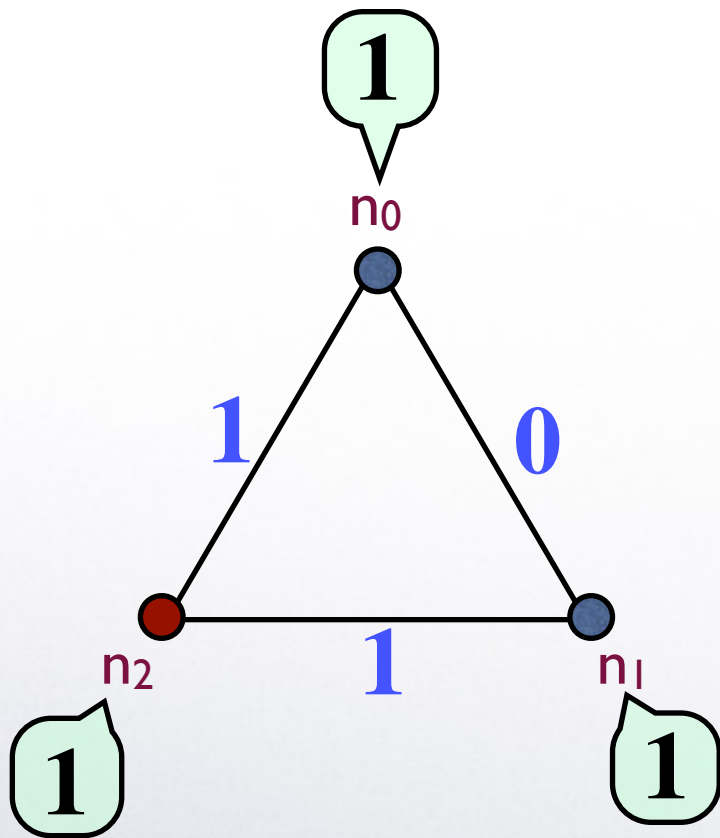
Example: DC nets (ring of 3 nodes, $b=1$)



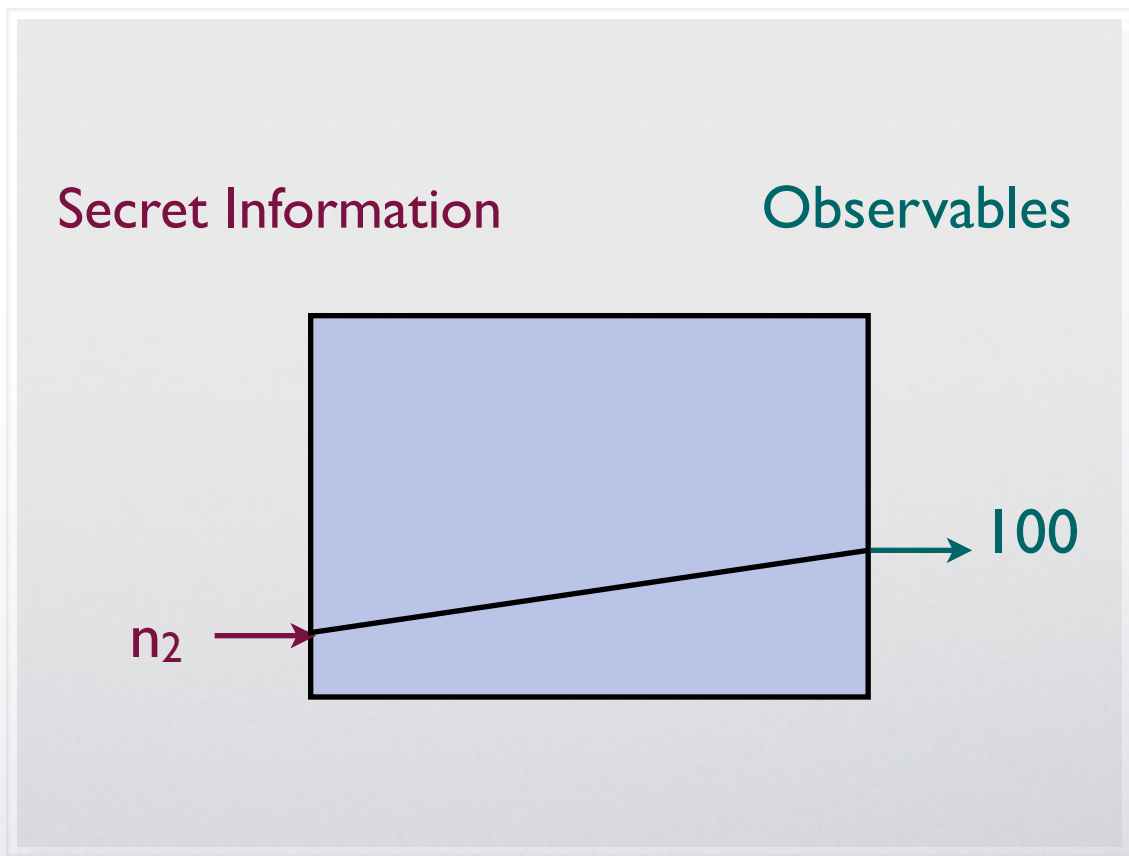
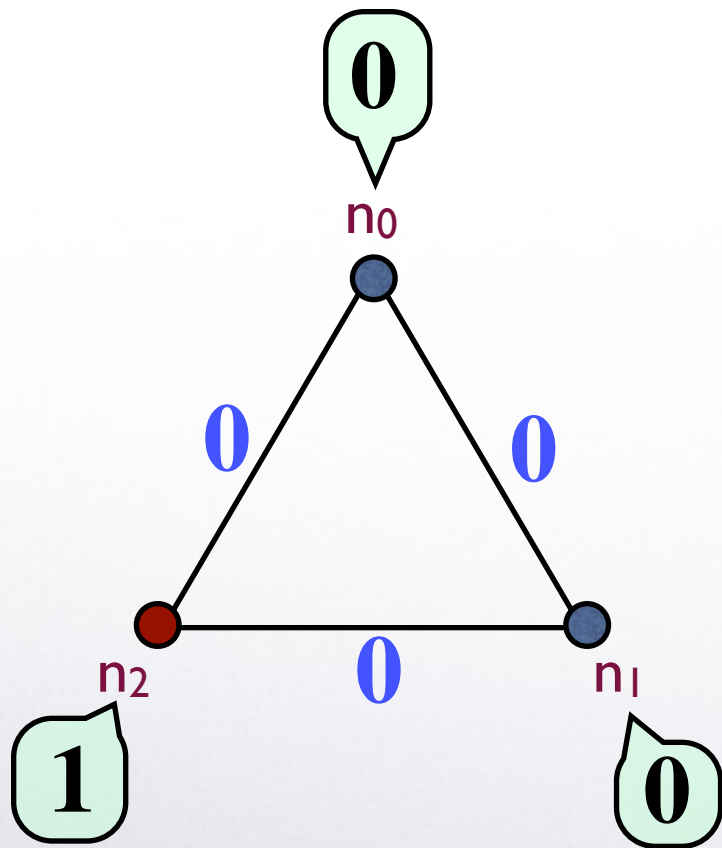
Example: DC nets (ring of 3 nodes, $b=1$)



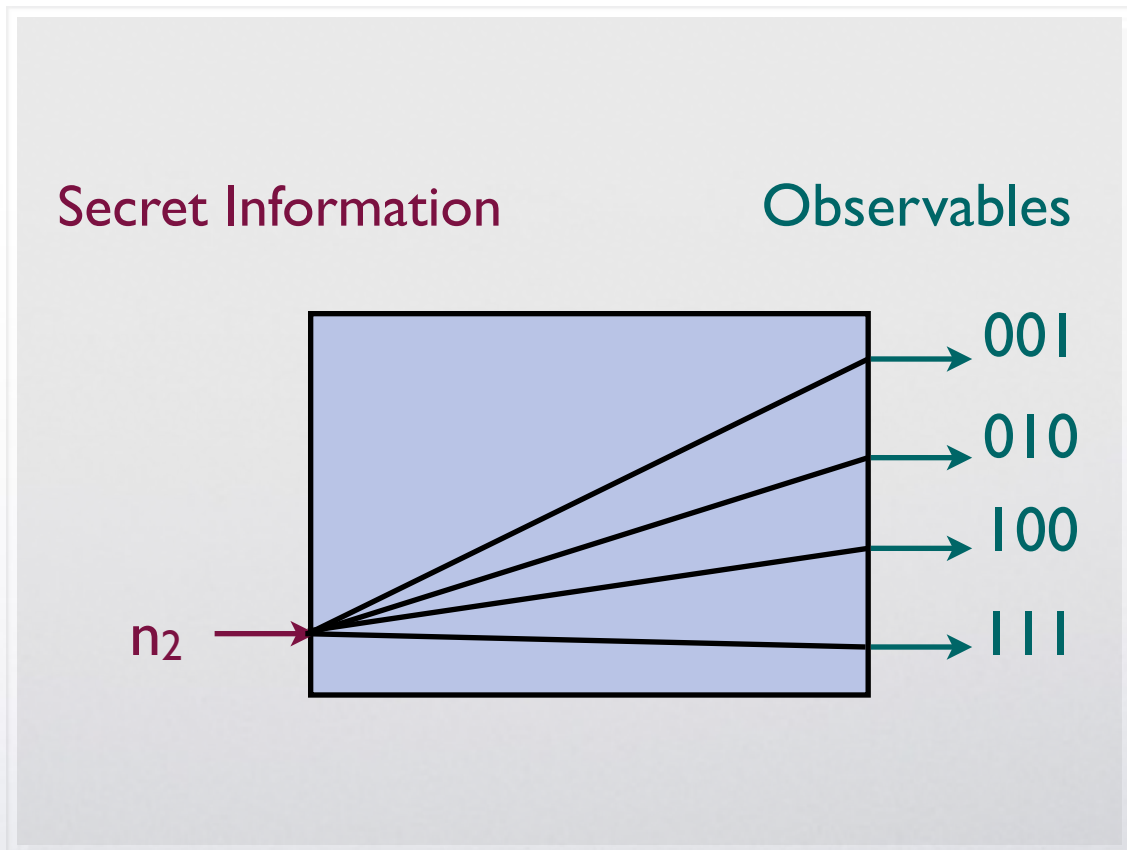
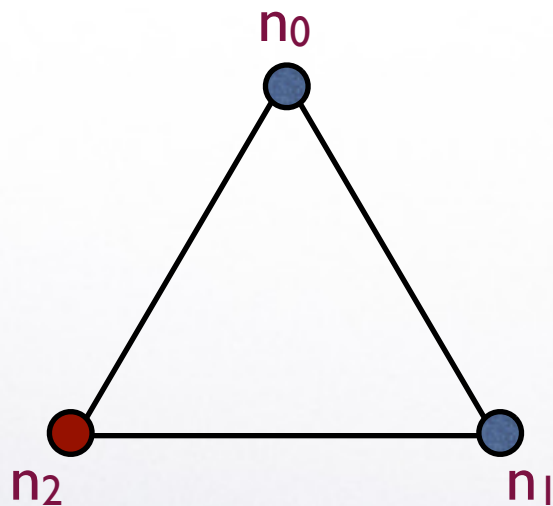
Example: DC nets (ring of 3 nodes, $b=1$)



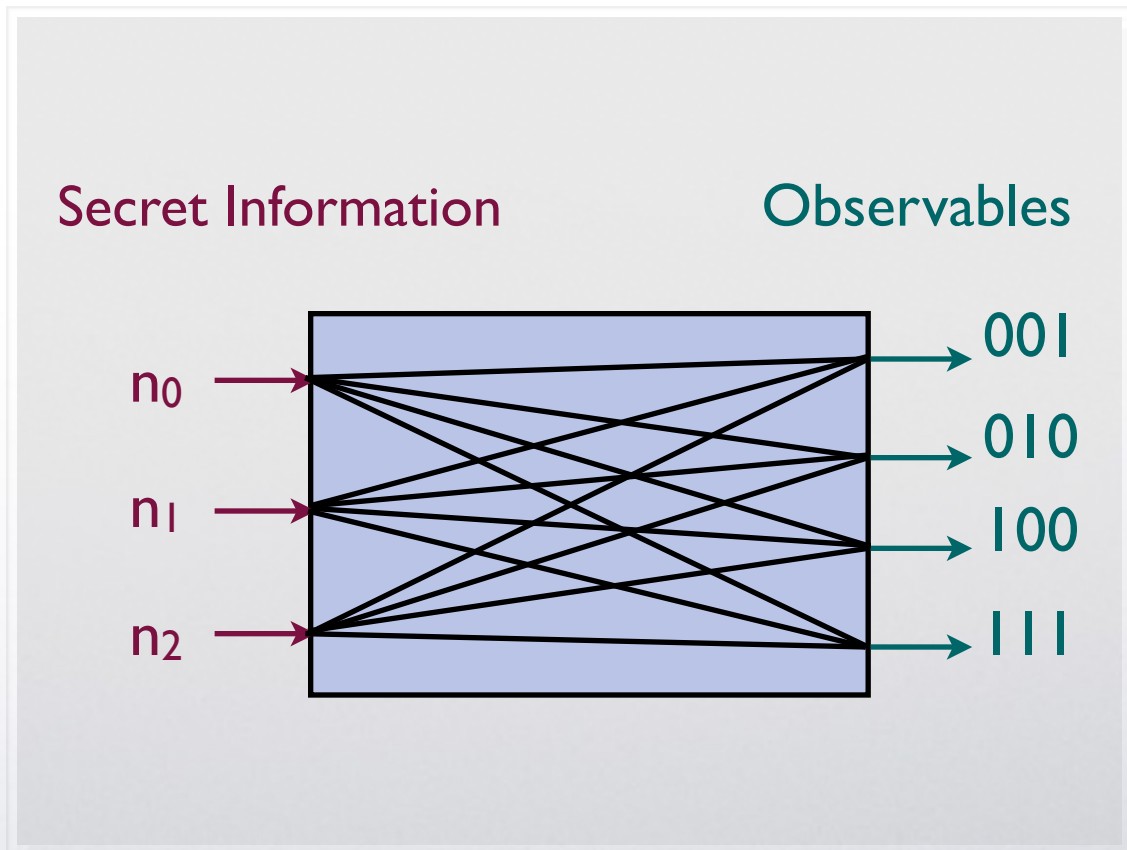
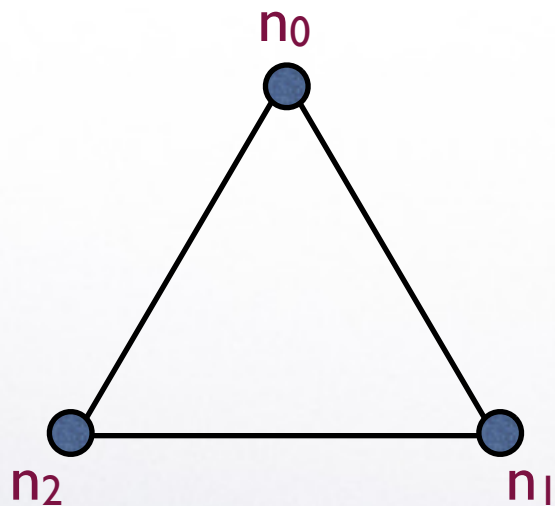
Example: DC nets (ring of 3 nodes, $b=1$)



Example: DC nets (ring of 3 nodes, $b=1$)



Example: DC nets (ring of 3 nodes, $b=1$)



Example: DC nets (ring of 3 nodes, $b=1$)

	001	010	100	111
n_0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
n_1	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
n_2	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

fair coins: $\Pr(0) = \Pr(1) = \frac{1}{2}$
 strong anonymity

	001	010	100	111
n_0	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$
n_1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$
n_2	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$

biased coins: $\Pr(0) = \frac{2}{3}$, $\Pr(1) = \frac{1}{3}$
 The source is more likely to declare 1 than 0

Quantitative Information Flow

- Intuitively, the **leakage** is the (probabilistic) information that the adversary **gains** about the **secret** through the **observables**
- Each observable **changes** the **prior** probability distribution on the secret values into a **posterior** probability distribution according to the **Bayes** theorem (Bayesian update)
- In the average, the posterior probability distribution gives a **better hint** about the actual secret value

Bayesian update: prior \Rightarrow posterior

Bayesian update: prior \Rightarrow posterior

$p(n)$		001	010	100	111
$\frac{1}{2}$	n_0	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_2	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$

prior
secret
prob

$p(o|n)$
conditional prob

Bayesian update: prior \Rightarrow posterior

$p(n)$		001	010	100	111
$\frac{1}{2}$	n_0	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_2	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$

prior
secret
prob

$p(o|n)$
conditional prob

	001	010	100	111
n_0	$\frac{1}{6}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
n_1	$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$	$\frac{1}{18}$
n_2	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$

$p(n,o)$
joint prob

Bayesian update: prior \Rightarrow posterior

$p(n)$		001	010	100	111
$\frac{1}{2}$	n_0	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_1	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{1}{4}$	n_2	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$

prior
secret
prob

$p(o|n)$
conditional prob

$p(o)$	$\frac{5}{18}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{2}{9}$	obs prob
	001	010	100	111	
n_0	$\frac{1}{6}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	
n_1	$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$	$\frac{1}{18}$	
n_2	$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$	

$p(n,o)$
joint prob

Bayesian update: prior \Rightarrow posterior

$$p(n|o) = \frac{p(n, o)}{p(o)}$$

$p(n|001)$

$\frac{3}{5}$

$\frac{1}{5}$

$\frac{1}{5}$

post
secret
prob

n_0

n_1

n_2

001 010 100 111

$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$	$\frac{2}{9}$
$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{3}$	$\frac{2}{9}$

$p(o|n)$

conditional prob

$p(o)$

$\frac{5}{18}$

$\frac{1}{4}$

$\frac{1}{4}$

$\frac{2}{9}$

obs
prob

001 010 100 111

n_0

n_1

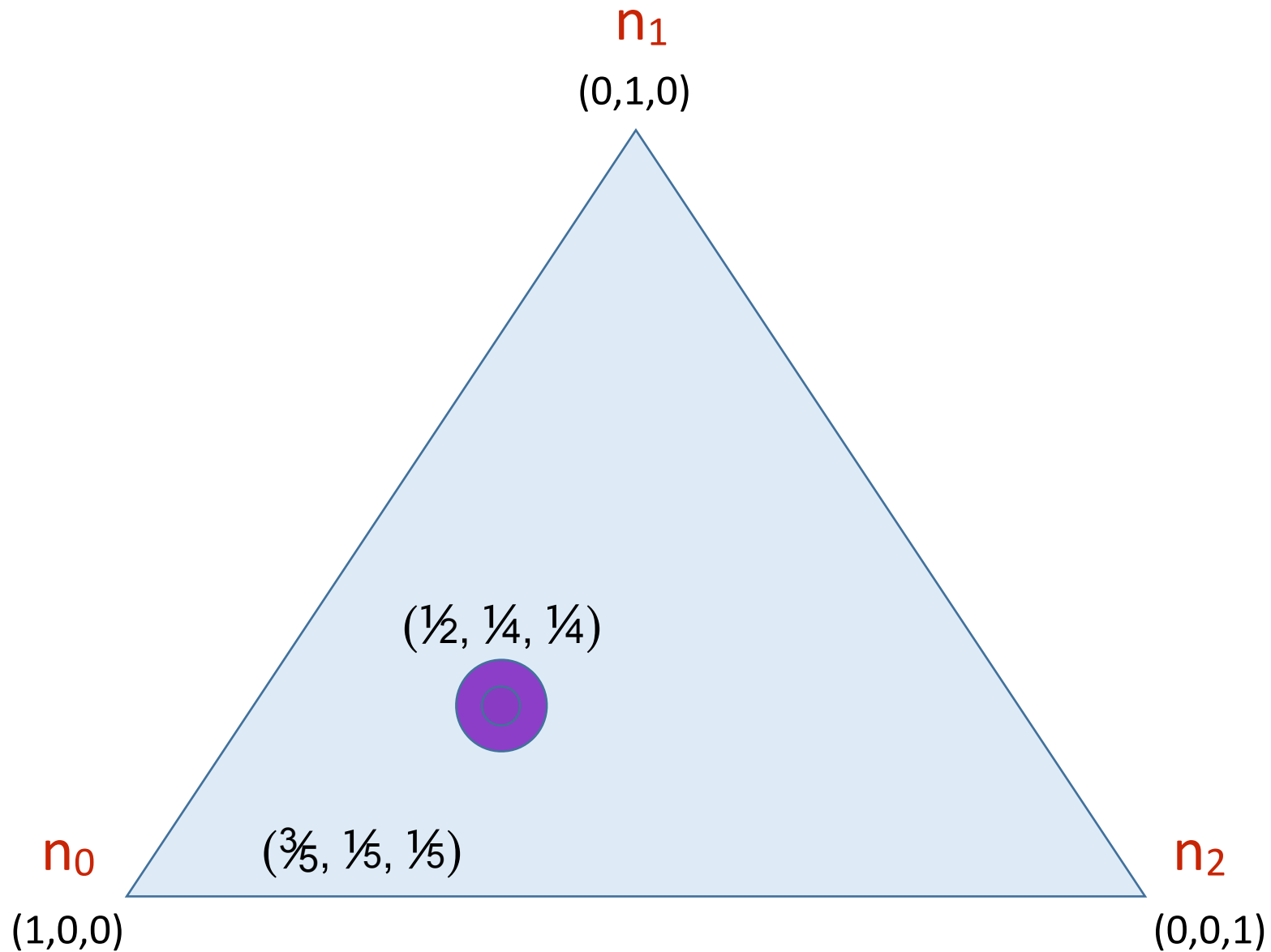
n_2

$\frac{1}{6}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$	$\frac{1}{18}$
$\frac{1}{18}$	$\frac{1}{18}$	$\frac{1}{12}$	$\frac{1}{18}$

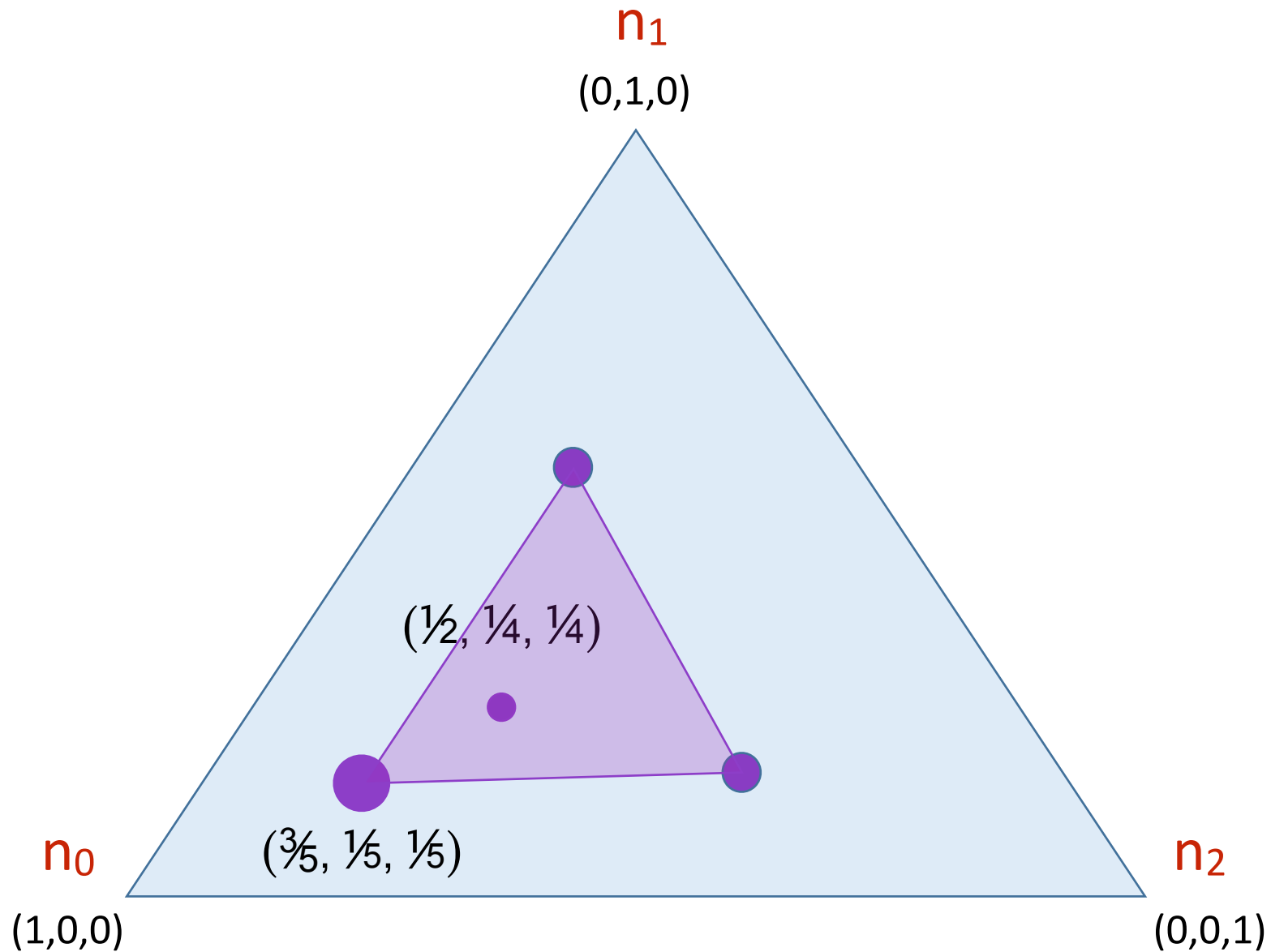
$p(n, o)$

joint prob

A graphical representation of the Bayesian update



A graphical representation of the Bayesian update



Information theory: useful concepts

- **Entropy $H(X)$ of a random variable X**

- A measure of the degree of uncertainty of the events
- It can be used to measure the vulnerability of the secret, i.e. how “easily” the adversary can discover the secret

- **Mutual information $I(S;O)$**

- Degree of correlation between the input S and the output O
- formally defined as difference between:
 - $H(S)$, the entropy of S *before* knowing, and
 - $H(S|O)$, the entropy of S *after* knowing O
- It can be used to measure the leakage:

$$\text{Leakage} = I(S;O) = H(S) - H(S|O)$$

- $H(S)$ depends only on the prior; $H(S|O)$ can be computed using the prior and the channel matrix

Entropy and Operational Interpretation

In the realm of security, there is no unique notion of entropy. A suitable notion of entropy should have an **operational interpretation** in terms of the kind of **adversary** we want to **model** , namely:

- the kind of attack, and
- how we measure its success

A general **model of adversary** [Köpf and Basin, CCS'07]:

- Assume an oracle that answers yes/no to questions of a certain form.
- The adversary is defined by the form of the questions, and the measure of success of the attack.
- In general we consider the best strategy for the attacker, with respect to a given measure of success.

Entropy

Example of adversary:

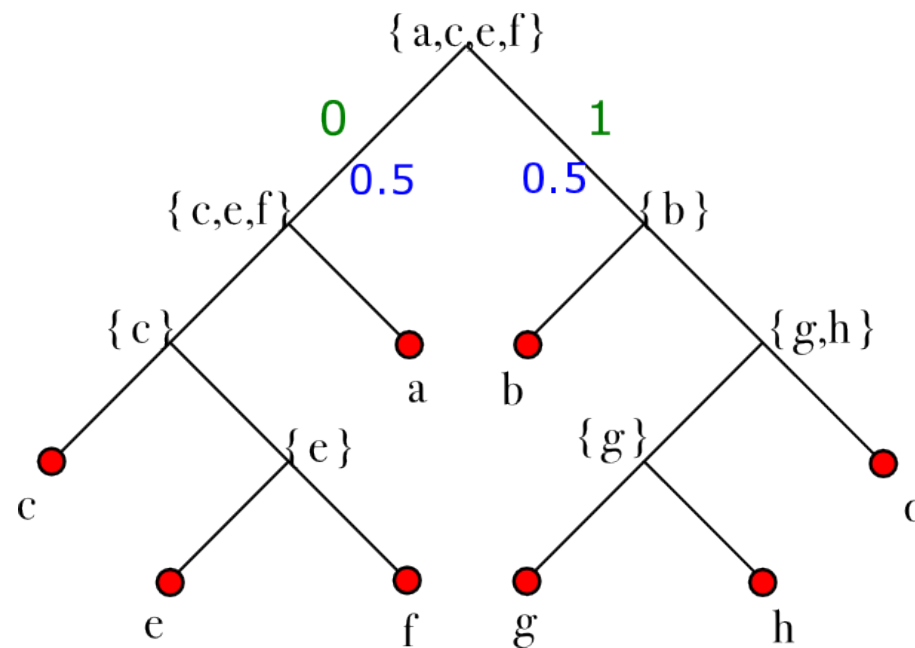
- The questions are of the form: “is $S \in P$?”
- The measure of success is: the expected number of questions needed to find the value of S in the attacker’s best strategy

It is possible to prove that the best strategy for the adversary is to split each time the search space in two subspaces with same probability masses.
This gives a perfectly balanced tree.

Entropy

Example: $S \in \{ a, b, c, d, e, f, g, h \}$

$$p(a) = p(b) = \frac{1}{4} \quad p(c) = p(d) = \frac{1}{8} \quad p(e) = p(f) = p(g) = p(h) = \frac{1}{16}$$



Entropy

Since in the best strategy the tree is balanced, the number of questions needed to determine the value s of the secret is: $-\log p(s)$
(log is in base 2)

Hence the **expected number** of questions is:

$$H(S) = - \sum_s p(s) \log p(s)$$

Uncertainty: **Shannon entropy**

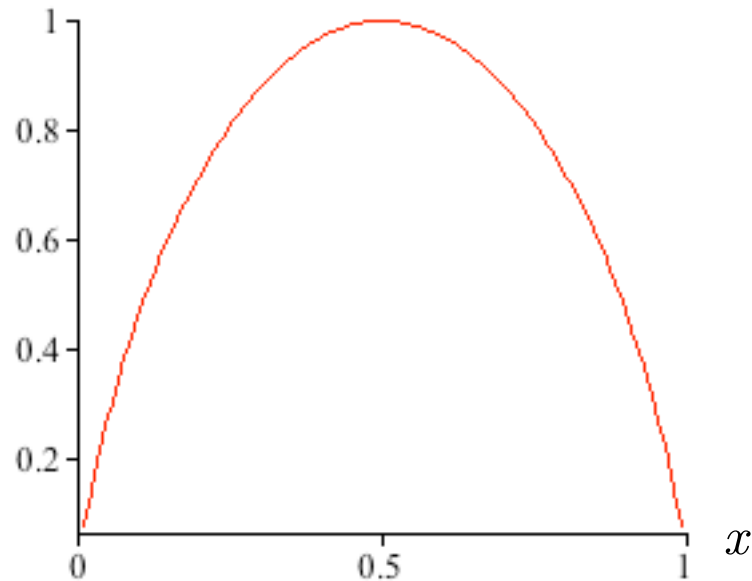
Shannon entropy: properties

The entropy is a concave function of the probability distribution

$$S = \{a, b\}$$

$$p(a) = x \quad p(b) = 1 - x$$

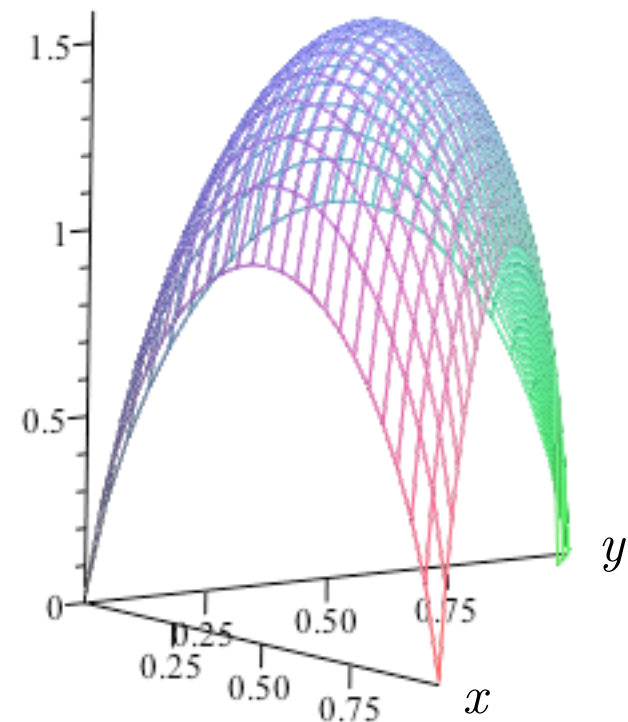
$H(S)$



$$S = \{a, b, c\}$$

$$p(a) = x \quad p(b) = y \quad p(c) = 1 - (x + y)$$

$H(S)$



Shannon conditional entropy

The conditional entropy is the expected value of the updated entropies:

$$\begin{aligned} H(S|O) &= \sum_o p(o) H(S|O = o) \\ &= - \sum_o p(o) \sum_s p(s|o) \log p(s|o) \end{aligned}$$

Shannon leakage

A priori

$$H(S) = - \sum_s p(s) \log p(s)$$

A posteriori

$$H(S | O) = - \sum_o p(o) \sum_s p(s|o) \log p(s|o)$$

Leakage = Mutual Information $I(S; O) = H(S) - H(S|O)$

- In general $H(S) \geq H(S|O)$
 - the entropy may increase after one single observation, but in the average it cannot increase
- $H(S) = H(S|O)$ if and only if S and O are independent
 - This is the case if and only if all rows of the channel matrix are the same
 - This case corresponds to strong anonymity in the sense of Chaum
- Shannon capacity $C = \max I(S;O)$ over all priors (worst-case leakage)

Entropy: Alternative notions

As we argued before, there is no unique notion of vulnerability. It depends on:

- the model of attack, and
- how we measure its success

Entropy: Alternative notions

We saw that if

- the questions are of the form: “is $S \in P$?”, and
- the measure of success is: the expected number of questions needed to find the value of S in the adversary’s best strategy

then the natural measure of protection is Shannon’s entropy

However, this model of attack does not seem so natural in security, and alternatives have been considered. In particular, the **limited-try attacks**

- The adversary has a limited number of attempts at its disposal
- The measure of success is the probability that he discovers the secret during these attempts (in his best strategy)

Obviously the best strategy for the adversary is to try first the values which have the highest probability

One try attacks: Rényi min-entropy

One-try attacks

- The questions are of the form: “is $S = s$?”
- The measure of success is: $-\log(\max_s p(s))$

Rényi min-entropy: $H_\infty(S) = -\log(\max_s p(s))$

Like in the case of Shannon entropy, $H_\infty(S)$ is highest when the distribution is uniform, and it is 0 when the distribution is a delta of Dirac (no uncertainty).

Conditional min-entropy

The expected value of the prob. of success (aka converse of the Bayes risk):

$$\begin{aligned}\Pr_{succ}(S|O) &= \sum_o p(o) \Pr_{succ}(S|O = o) \\ &= \sum_o p(o) \max_s p(s|o) \\ &= \sum_o \max_s (p(o|s) p(s))\end{aligned}$$

Now define $H_\infty(S|O) = -\log \Pr_{succ}(S|O)$ [Smith 2009]

Leakage in the min-entropy approach

A priori

$$H_{\infty}(S) = -\log \max_s p(s)$$

A posteriori

$$H_{\infty}(S|O) = -\log \sum_o \max_s (p(o|s) \cdot p(s))$$

Leakage = min-Mutual Inf.

$$I_{\infty}(S; O) = H_{\infty}(S) - H_{\infty}(S|O)$$

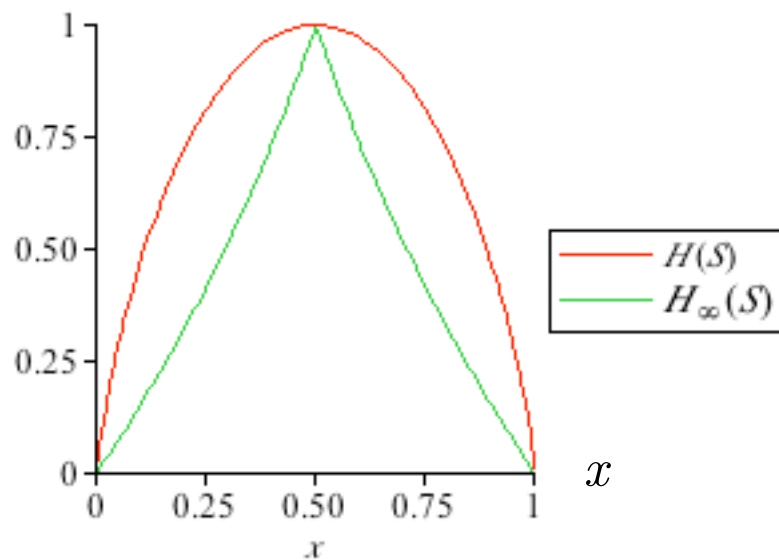
Properties of the min-entropy leakage

- In general $I_\infty(S;O) \geq 0$
- $I_\infty(S;O) = 0$ if all rows are the same (but not viceversa)
- Define min-capacity: $C_\infty = \max I_\infty(S;O)$ over all priors. We have:
 1. $C_\infty = 0$ if and only if all rows are the same
 2. $C_\infty = C$ in the deterministic case
 3. $C_\infty \geq C$ in general
 4. C_∞ is obtained on the uniform distribution (but, in general, there can be other distribution that give maximum leakage)
 5. **$C_\infty = \text{the log of the sum of the max of each column}$**

Rényi min-entropy vs. Shannon entropy

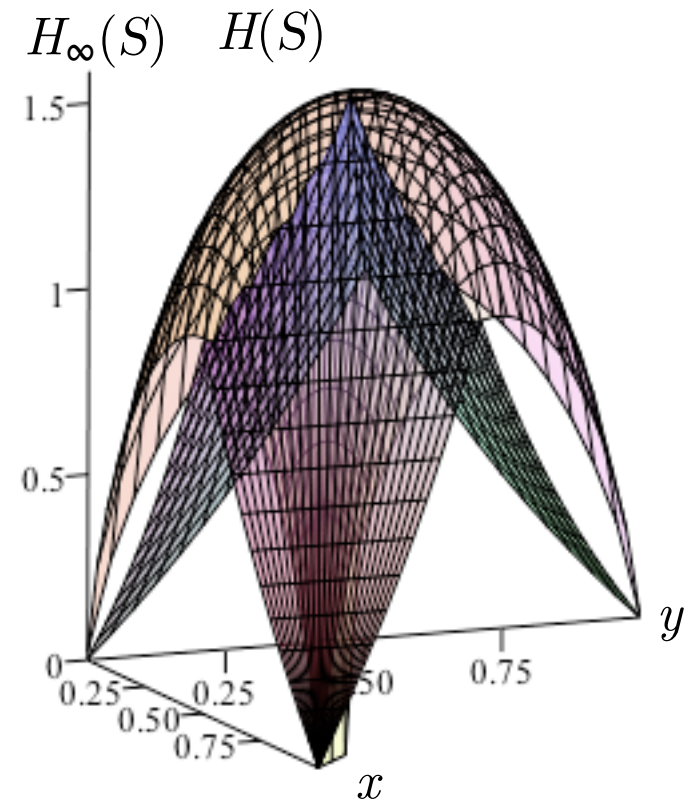
$$S = \{a, b\}$$

$$p(a) = x \quad p(b) = 1 - x$$



$$S = \{a, b, c\}$$

$$p(a) = x \quad p(b) = y \quad p(c) = 1 - (x + y)$$

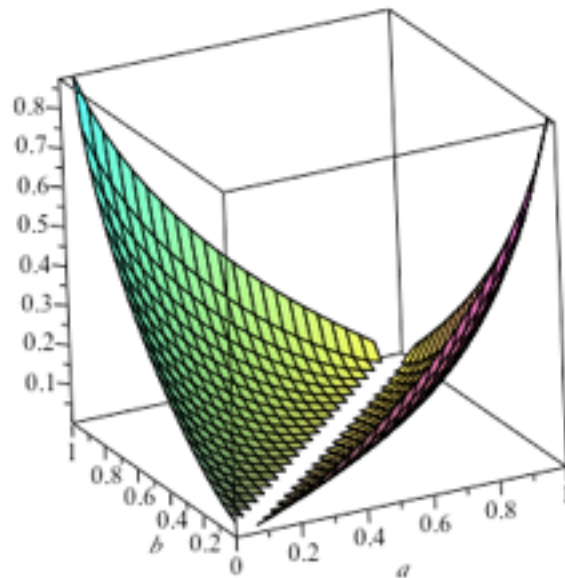


Rényi min entropy and conditional entropy are the log of piecewise linear functions

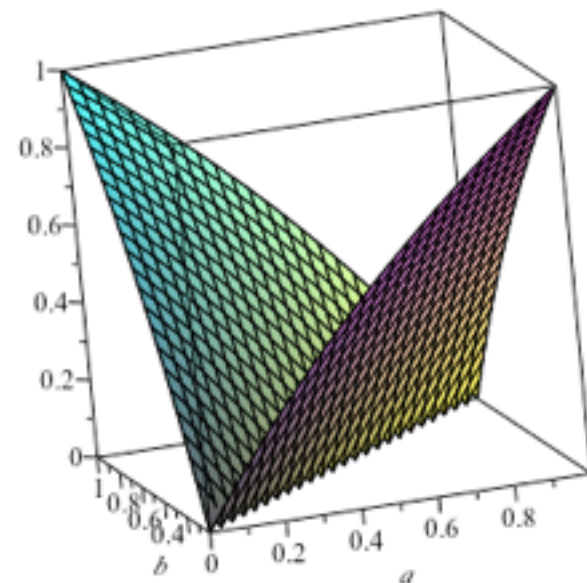
Shannon capacity vs. Rényi min-capacity

binary channel

a	$1-a$
b	$1-b$



Shannon capacity



Rényi min-capacity

In general, Rényi min capacity is an upper bound for Shannon capacity

Limitations of min-entropy leakage and further readings

- Min-entropy leakage implicitly assumes an operational scenario where adversary \mathcal{A} benefits only by guessing secret S **exactly**, and in **one try**.
- But many other scenarios are possible:
 - Maybe \mathcal{A} can benefit by guessing S **partially** or **approximately**.
 - Maybe \mathcal{A} is allowed to make **multiple** guesses.
 - Maybe \mathcal{A} is **penalized** for making a wrong guess.
- **g-leakage** captures all the above scenarios
- **g-leakage** can express all main notions of leakage from the literature: Shannon leakage, Min-entropy leakage, guessing entropy leakage, etc.

M. S. Alvim, K. Chatzikokolakis, C. Palamidessi, and G. Smith. Measuring information leakage using generalized gain functions. Proc. of CSF, IEEE

M. S. Alvim, K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith. Axioms for information leakage. Proc. of the CSF, IEEE, 2016.

Thank you !

Part II

Differential Privacy

Plan of the lecture

- A brief panoramic of the main deterministic approaches to privacy
- Differential Privacy (DP)
- The Bayesian interpretation of DP
- Compositionality and independence from prior
- The privacy budget
- Implementation of DP: Laplacian noise

The problem

- In general, the problem of privacy is to protect the disclosure of **sensitive information** of individuals when a collection of data about these individuals (*dataset*) is made **publicly available**
- The process of transforming the dataset in order to avoid such disclosure is called **sanitization**

First solution: anonymization

- This is the most obvious solution: remove the identity of individuals from the database, so that the sensitive information cannot be directly linked to the individual
- Example: assume that we have a medical database, where the sensitive information is disease that has been diagnosed
- For instance, Jorah Mormont may not want to reveal that he is affected by greyscale.

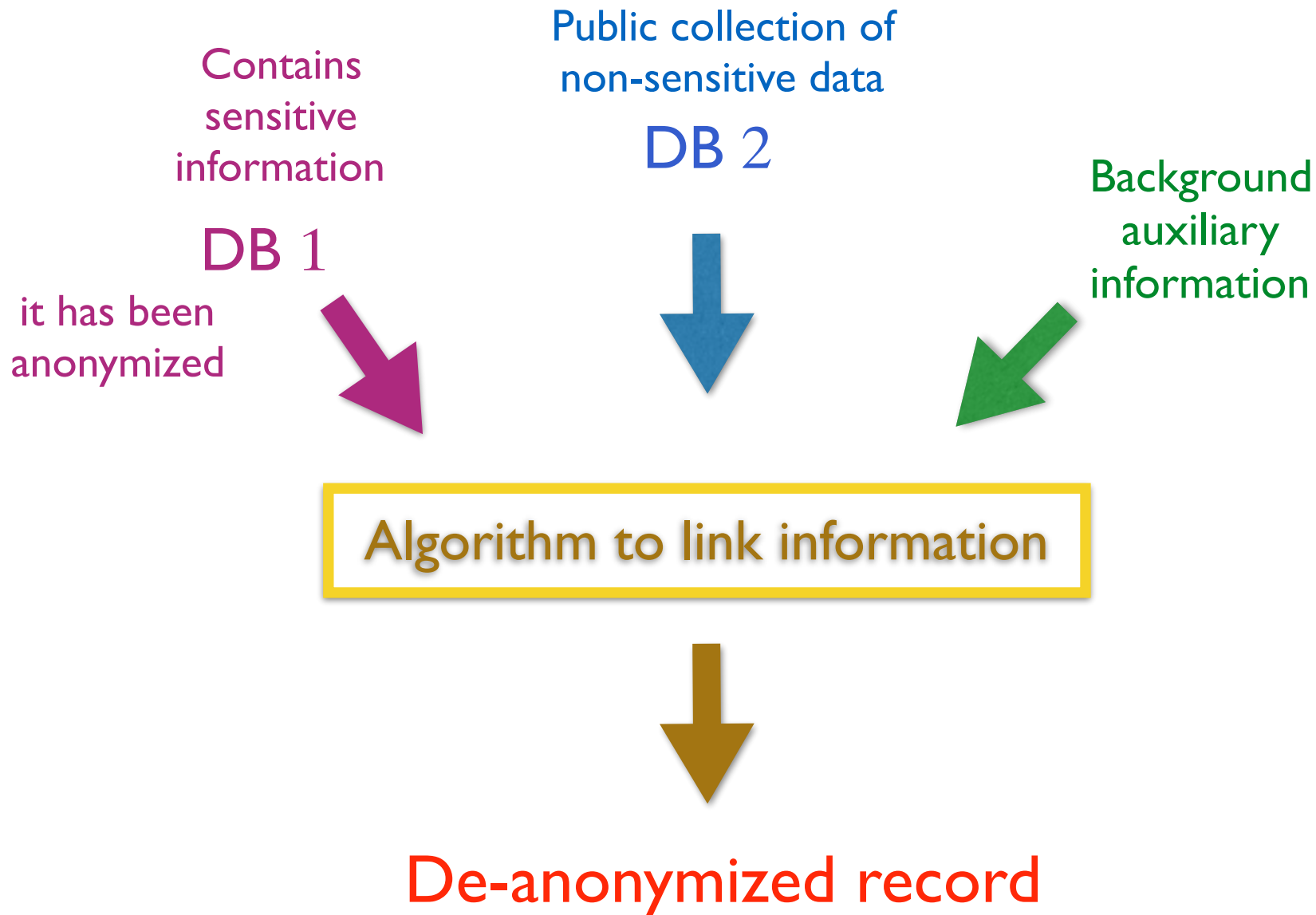
	Name	age	Disease
1	Jon Snow	30	cold
2	Jamie Lannister	39	amputated hand
3	Arya Stark	16	stomach ache
4	Bran Stark	14	crippled
5	Sandor Clegane	45	ignifobia
6	Jorah Mormont	48	gleyscale
7	Eddad Stark	32	headache
8	Ramsay Bolton	32	psychopath
9	Daenerys Targaryen	25	mania of grandeur

First solution: anonymization

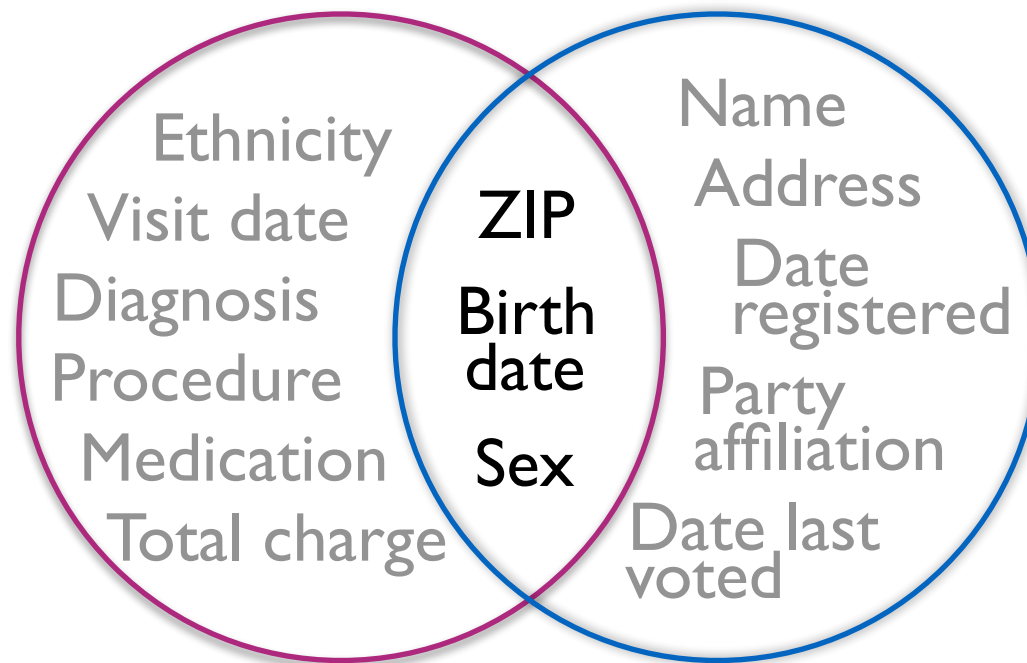
- Anonymization removes the column of the name, so that, for instance, the grayscale disease cannot be directly linked to Jorah Mormont
- Historically the first method, still used nowadays
- However, this solution has been (already several years ago) shown to be very weak and prone to de-anonymization attacks

	Name	age	Disease
1	-	30	cold
2	-	39	amputated hand
3	-	16	stomach ache
4	-	14	crippled
5	-	45	ignifobia
6	-	48	gleyscale
7	-	32	headache
8	-	32	psychopath
9	-	25	mania of grandeur

Sweeney's de-anonymization attack by linking



Sweeney's de-anonymization attack by linking [around year 2000]



DB 1: Medical data

DB 2: Voter list

87 % of US population is uniquely identifiable by 5-digit ZIP, gender, DOB

This attack has lead to the proposal of k-anonymity

K-anonymity [Samarati & Sweeney]

- Quasi-identifier: Set of attributes that can be linked with external data to uniquely identify individuals
- Make every record in the table indistinguishable from a least $k-1$ other records with respect to quasi-identifiers. This can be done by:
 - suppression of attributes, and/or
 - generalization of attributes, and/or
 - addition of dummy records
- Linking on quasi-identifiers yields at least k records for each possible value of the quasi-identifier

K-anonymity

Example: 4-anonymity w.r.t. the quasi-identifiers (nationality, ZIP, age)

- achieved by suppressing the nationality and generalizing ZIP and age

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Figure 1. Inpatient Microdata

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

Figure 2. 4-anonymous Inpatient Microdata

Problems with k-anonymity

- Obvious problem:
in the sanitized
dataset, all the
individual in a group
may the same value
for the sensitive
data, like in this
table
- Clearly, the people
in that group are
not protected from
the revelation of
their disease

Non-Sensitive					Sensitive
	Rase	Age	Sex	Zip Code	Disease
1	*	< 40	*	120**	Cancer
2	*	< 40	*	120**	Cancer
3	*	< 40	*	120**	Cancer
4	*	< 40	*	120**	Cancer
5	*	≥ 50	*	151**	Hemophilia
6	*	≥ 50	*	151**	Cancer
7	*	≥ 50	*	151**	Virus
8	*	≥ 50	*	151**	Virus
9	*	4*	*	120**	Hemophilia
10	*	4*	*	120**	Hemophilia
11	*	4*	*	120**	Virus
12	*	4*	*	120**	Virus

Table 2: 4-anonymous inpatient microdata.

ℓ -diversity [Kifer et al.]

- A solution to this problem was proposed under the name of ℓ -diversity.
- The idea is to form the groups in such a way that each group contains a variety of values for the sensitive data

Non-Sensitive					Sensitive
	Rase	Age	Sex	Zip Code	Disease
1	*	≤ 50	*	120**	Cancer
2	*	≤ 50	*	120**	Cancer
9	*	≤ 50	*	120**	Hemophilia
11	*	≤ 50	*	120**	Virus
5	*	> 50	*	151**	Hemophilia
6	*	> 50	*	151**	Cancer
7	*	> 50	*	151**	Virus
8	*	> 50	*	151**	Virus
3	*	≤ 50	*	120**	Cancer
4	*	≤ 50	*	120**	Cancer
10	*	≤ 50	*	120**	Hemophilia
12	*	≤ 50	*	120**	Virus

Table 5: 3-diverse table

Problems with previous methods

- High-dimensional and sparse databases.
 - Example: Netflix attack
- Composition attacks
- High complexity

These problems have lead to look for a radically different approach

→ Differential Privacy

Netflix attack

Robust De-anonymization of Large Sparse Datasets.
Narayanan and Shmatikov.

Showed the limitations of K-anonymity

De-anonymization of the **Netflix Prize** dataset (500,000 anonymous records of movie ratings), using **IMDb** as the source of background knowledge.

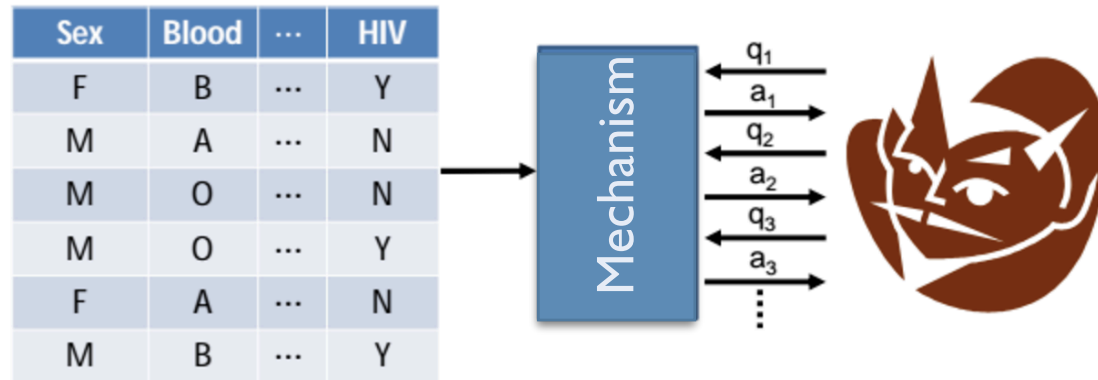
They demonstrated that an adversary who knows just a few preferences about an individual subscriber can identify his record in the dataset.





Differential Privacy

- Sanitization of Statistical Databases -

- SDBs are often used for research purposes. For example, a medical SDB can be used to study the correlation between certain diseases and other attributes like: age, sex, weight, etc.



- One can only retrieve aggregated information, not personal records
 - “What is the average weight of people affected by the disease ?” 
 - “Does Don have the disease ?” 

Deterministic methods have a problem are not robust wrt composition. Example

- A medical database D1 containing correlation between a certain disease and age.
- Query: “what is the minimal age of a person with the disease”

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

D1 is 2-anonymous with respect to the query. Namely, every possible answer partitions the records in groups of at least 2 elements

Alice	Bob
Carl	Don
Ellie	Frank

- A medical database D2 containing correlation between the disease and weight.
- Query: “what is the minimal weight of a person with the disease”

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Also D2 is 2-anonymous

Alice	Bob
Carl	Don
Ellie	Frank

k-anonymity is not compositional

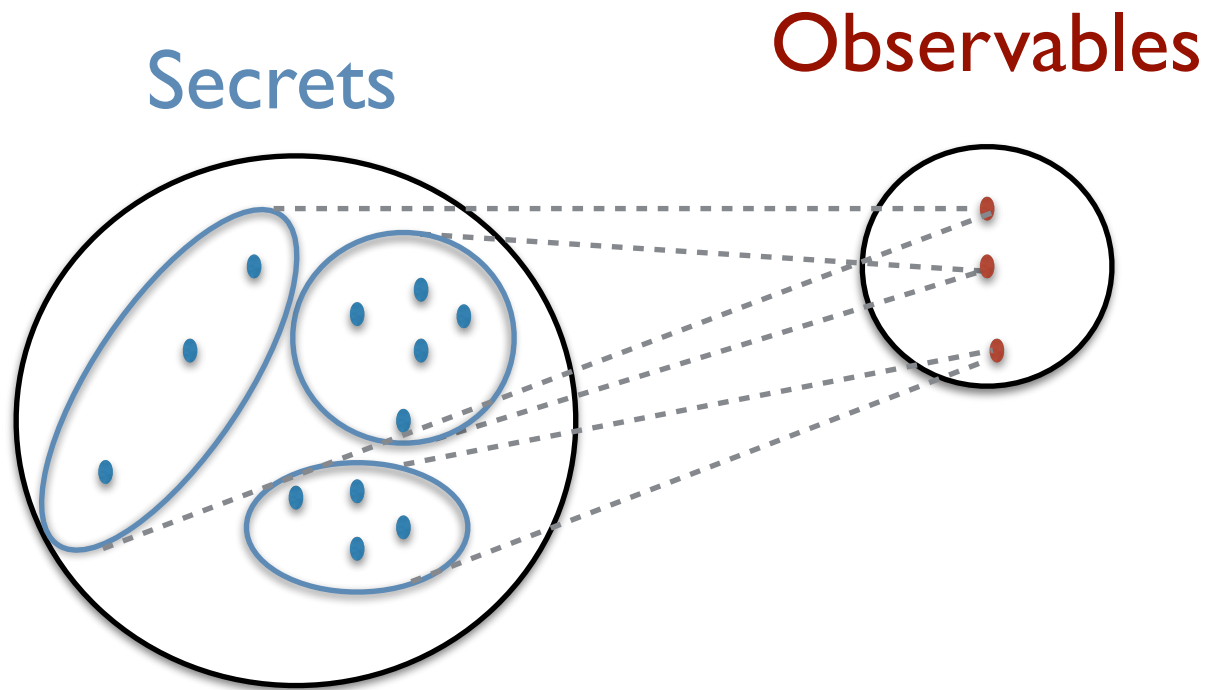
Combine with the two queries:
minimal weight and the minimal
age of a person with the disease
Answers: 40, 100. **Unique!**

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

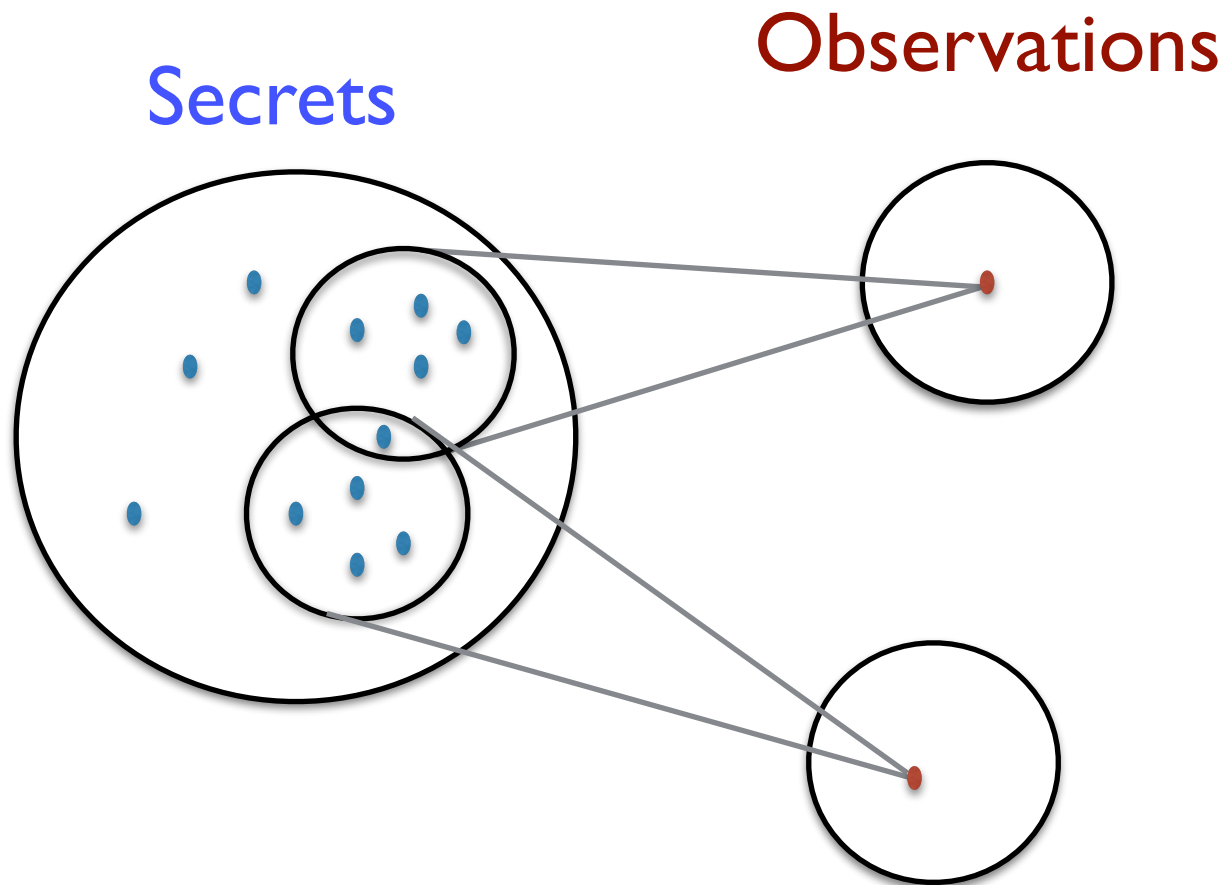
name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

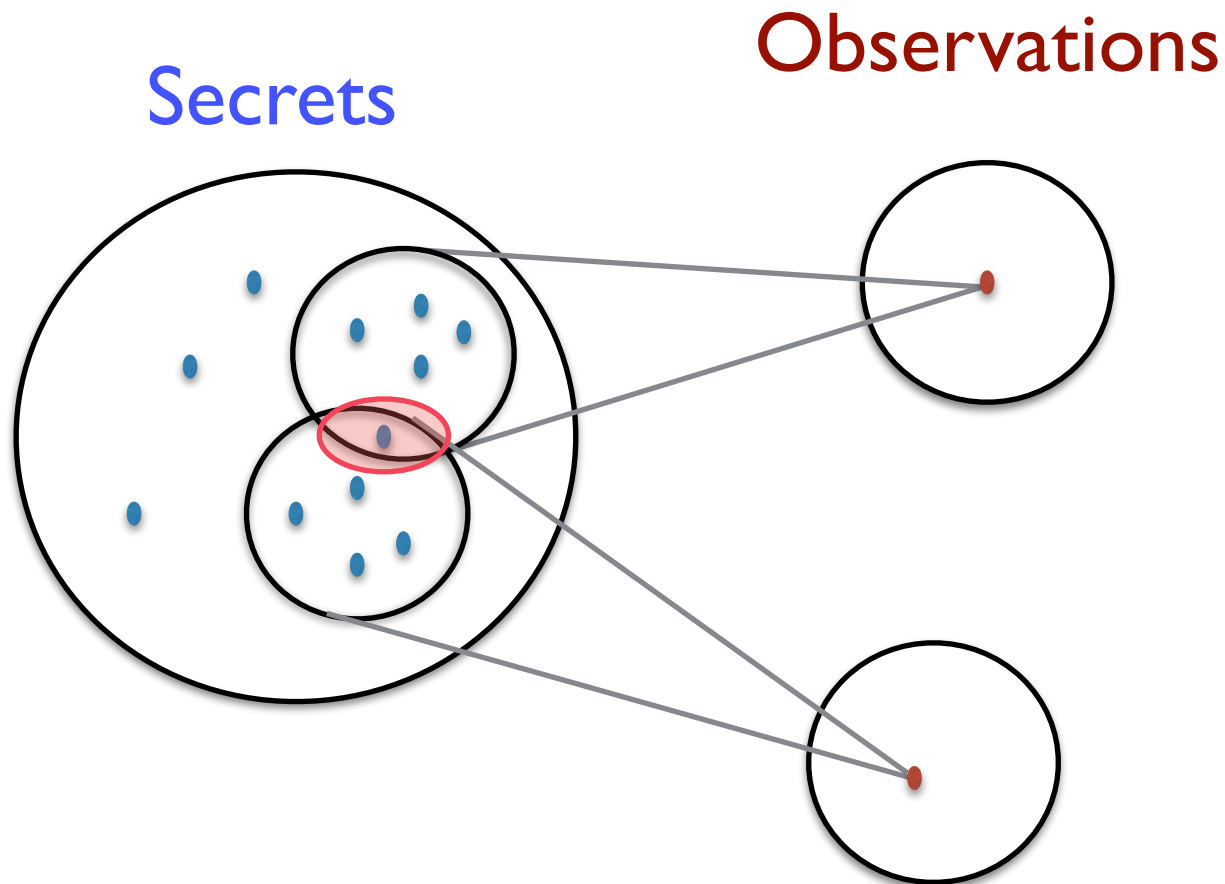
This is a general problem of **Deterministic approaches**:
They are based on the principle that one observable
corresponds to many possible values of the secret
(group anonymity)



Problem of the deterministic approaches: the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletons



Problem of the deterministic approaches: the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletons



Too bad!!! What can we do?

This is a job for...

Random man!



**“Always keep your foes confused.
If they are never certain who
you are or what you want,
they cannot know what
you are likely to do
next. Sometimes the best
way to baffle them is to
make moves that have
no purpose, or even seem
to work against you.”**

~ Petyr Baelish (Game of Thrones)

George R.R. Martin



Randomized approach for statistical databases

Introduce some probabilistic noise on the answer to obfuscate the link with any particular individual

Noisy answers

minimal age:

40 with probability $1/2$

30 with probability $1/4$

50 with probability $1/4$

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

minimal weight:

100 with prob. 4/7

90 with prob. 2/7

60 with prob. 1/7

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

Even if he combines the answers, the adversary cannot tell for sure whether a certain person has the disease

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy mechanisms

- The mechanisms reports an approximate answer, typically generated randomly on the basis of the true answer and of some probability distribution
- The probability distribution must be chosen carefully, in order to not destroy the utility of the answer
- A good mechanism should provide a good trade-off between **privacy** and **utility**. Note that, for the same level of privacy, different mechanisms may provide different levels of utility.

Differential Privacy

Definition A randomized mechanism \mathcal{K} is ϵ -differentially private if for all databases x, x' which are adjacent (i.e., differ for only one record), and for all $z \in \mathcal{Z}$, we have

$$\frac{p(K = z | X = x)}{p(K = z | X = x')} \leq e^\epsilon$$

By the Bayes theorem, this definition corresponds to say that the answer given by K does not change significantly the knowledge about an individual (prior and posterior are close)

Important properties:

- DP is robust with respect to composition of queries: the level of privacy ϵ decreases linearly with the number of queries
- The definition of DP is independent from the prior

Typical implementation of differential privacy: add Laplacian noise

- Randomized mechanism for a query $f: \mathcal{X} \rightarrow \mathcal{Y}$.
- **Add Laplacian noise.** If the exact answer is y , the reported answer is z , with a probability density function defined as:

$$dP_y(z) = c e^{-\frac{|z-y|}{\Delta f} \varepsilon}$$

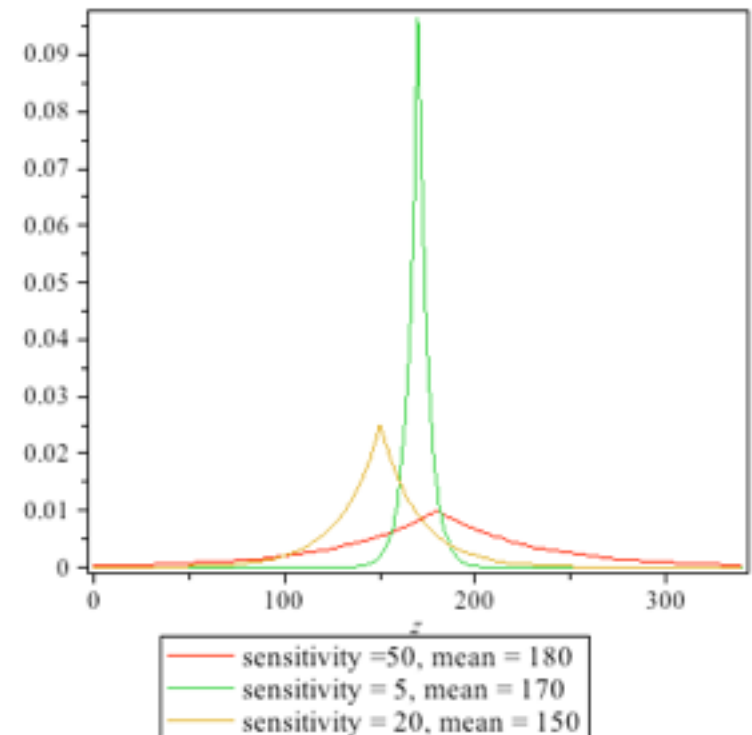
where Δf is the *sensitivity* of f :

$$\Delta f = \max_{x \sim x' \in \mathcal{X}} |f(x) - f(x')|$$

($x \sim x'$ means x and x' are adjacent, i.e., they differ only for one record)

and c is a normalization factor:

$$c = \frac{\varepsilon}{2 \Delta f}$$



Example of Laplacian Mechanism

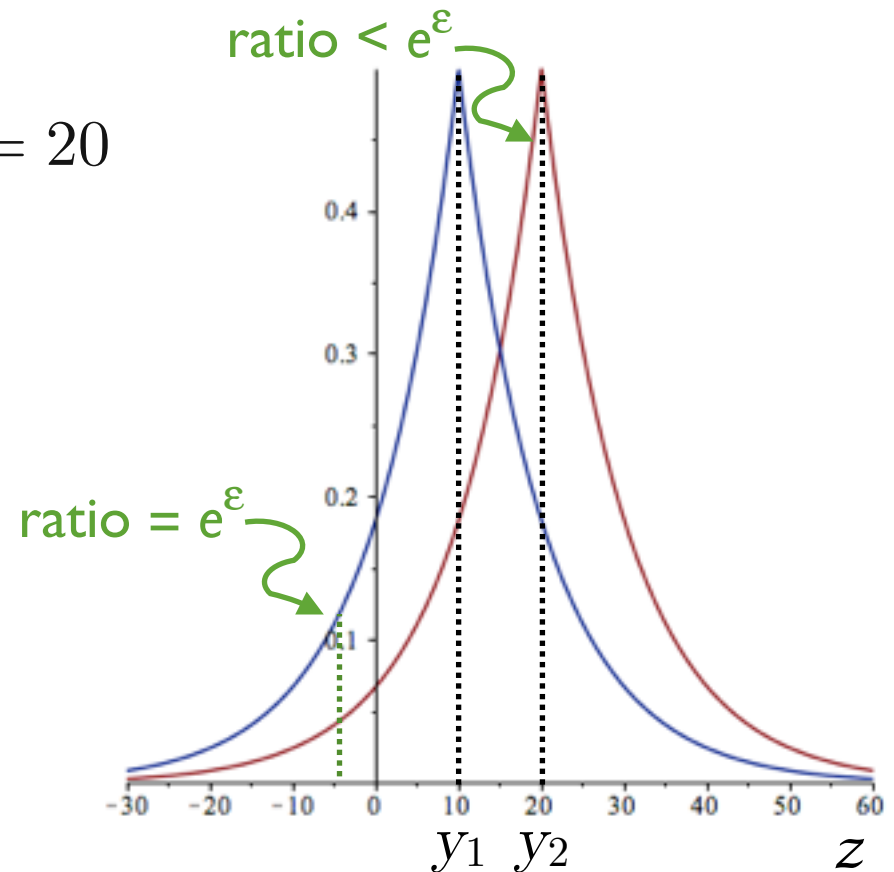
- $\varepsilon = 1$
- $\Delta_f = |f(x_1) - f(x_2)| = 10$
- $y_1 = f(x_1) = 10, y_2 = f(x_2) = 20$

Then:

- $dP_{y_1} = \frac{1}{2 \cdot 10} e^{\frac{|z-10|}{10}}$
- $dP_{y_2} = \frac{1}{2 \cdot 10} e^{\frac{|z-20|}{10}}$

The ratio between these distribution is

- $= e^\varepsilon$ outside the interval $[y_1, y_2]$
- $\leq e^\varepsilon$ inside the interval $[y_1, y_2]$



Utility

The reported answer, i.e. the answer given by the randomized function, should provide some useful information about the exact answer. This is formalized using the concept of *gain* function. In general, the gain function is anti-monotonic with the distance between the real value

Z = reported answer; Y = exact answer $gain: Z \times Y \rightarrow \text{Reals}$

Utility is defined as maximum expected gain:

$$\mathcal{U}(Y, Z) = \sum_{y, z} p(y, z) \text{gain}(y, \text{remap}(z))$$

In this formula, the remap is chosen so to maximize the expected gain.

The remap allows the user to use side information (i.e. a the priori probability) to maximize utility.

Optimal mechanisms

- Given a prior π , and a privacy level ϵ , an ϵ -differentially private mechanism K is called **optimal** if it provides the **best utility** among all those which provide ϵ -differential privacy
- Note that the privacy does not depend on the prior, but the utility (in general) does.
- In the finite case the optimal mechanism can be computed with linear optimization techniques, where the variables are the conditional probabilities $p(z \mid y)$ where y is the exact answer and z is the reported answer
- A mechanism is **universally optimal** if it is optimal for all priors π

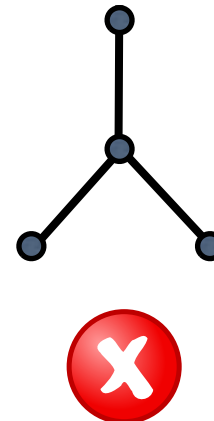
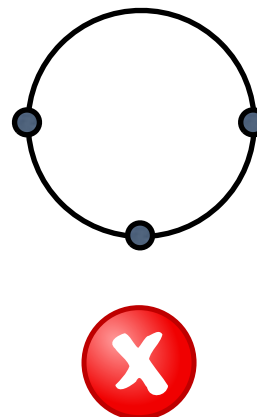
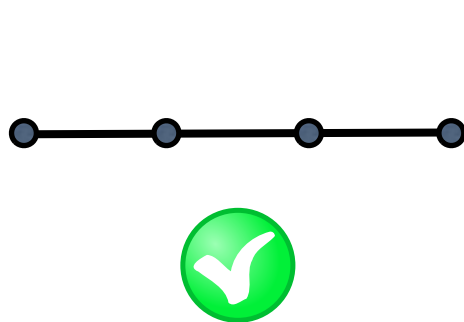
Privacy vs utility: two fundamental results

- I. [Ghosh et al., STOC 2009]
The geometric mechanism and the truncated geometric mechanism are **universally optimal** for counting queries and any (anti-monotonic) gain utility function

A counting query is a query of the form: How many individuals (tuples) in the database satisfy the property \mathcal{P} ?

Privacy vs utility: two fundamental results

2. [Brenner and Nissim, STOC 2010] The counting queries are the only kind of queries for which a universally optimal mechanism exists
- This means that for other kind of queries one the optimal mechanism is relative to a specific user.
 - The precise characterization is given in terms of the graph (\mathcal{V}, \sim) induced by (\mathcal{X}, \sim)



Thank you !

Part III

Location Privacy

1. Motivations
2. Generalization of Differential Privacy to arbitrary metric domains
3. Geo-indistinguishability
4. Optimal mechanisms for location privacy
5. Comparison

Location Privacy

- Use of Location Based Services
- Example: use an LBS to find points of interest (restaurants, shops, etc.)
- Revealing the exact location may be dangerous: profiling, inference of sensitive information, etc.
- Location Privacy by obfuscation: Revealing an approximate location is usually ok



Extending differential privacy to arbitrary metrics [Chatzikokolakis et al.]

Equivalent definition of DP:

A mechanism is ε -differentially private iff for every pair of databases x, x' and every answer z we have

$$\frac{p(z | x)}{p(z | x')} \leq e^{\varepsilon d_H(x, x')}$$

where d_H is the Hamming distance between x and x' , i.e., the number of records in which x and x' differ

Generalization: d -privacy

On a generic domain \mathcal{X} provided with a distance d :

$$\forall x, x' \in \mathcal{X}, \forall z \quad \frac{p(z | x)}{p(z | x')} \leq e^{\varepsilon d(x, x')}$$

Protection of the **accuracy** of the information

Properties of d -privacy :

$$\forall x, x' \in \mathcal{X}, \forall z \quad \frac{p(z | x)}{p(z | x')} \leq e^{\varepsilon d(x, x')}$$

- d -privacy is robust w.r.t. composition: the level of privacy decreases linearly with the number of observations
- d -privacy does not depend on the prior

Extended differential privacy on databases

- Surprisingly, in contrast to standard d.p., for certain metrics there are **universally optimal** mechanisms for various kinds of queries, including percentile, average, sum. (i.e., not only for counting queries)
- Two metrics that have this property are:
 - the manhattan metric (it protects the accuracy of the data)
 - the max metric (it protects all data)

Application to location privacy: geo-indistinguishability [Andres et al.]

d : the Euclidean distance

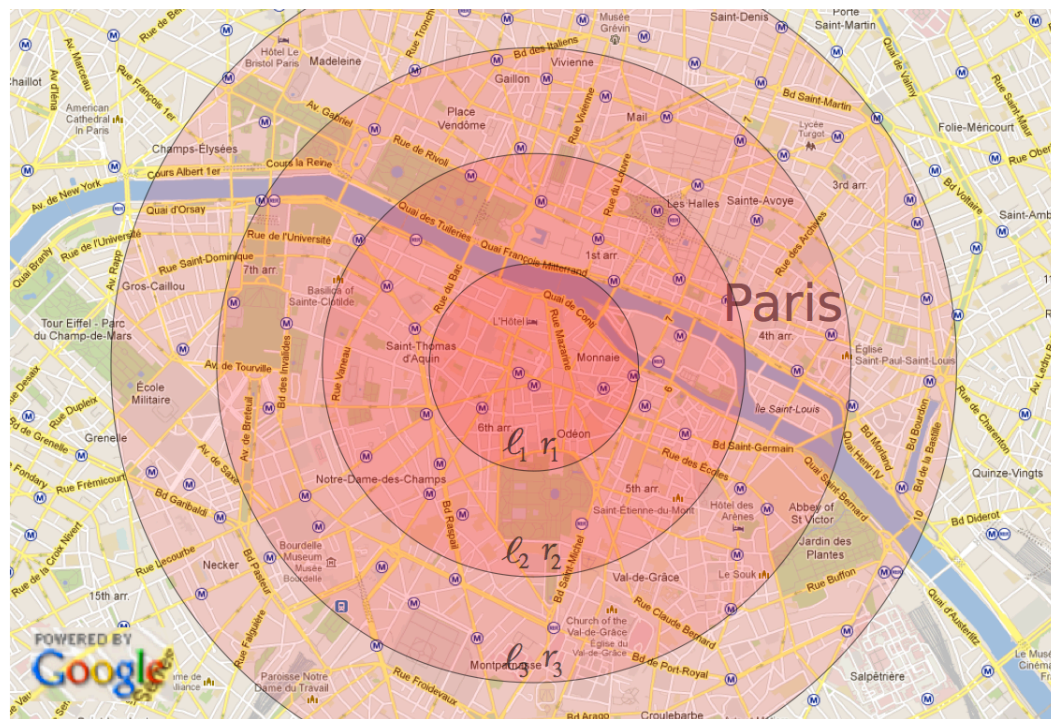
x : the exact location

z : the reported location

d – privacy

$$\frac{p(z|x)}{p(z|x')} \leq e^{\epsilon r}$$

where r is the distance
between x and x'



We call this property **geo-indistinguishability**

Note that, since it is a particular case of d -privacy, it is, like DP, independent from the prior, and the composition of observations (reported locations) decreases the level of privacy in a linear way.

Meaning of geo-indistinguishability

Using the Bayes theorem, we can give an alternative, and more intuitive, characterization of the geo-indistinguishability property:

According to the Bayes theorem, the conditional probability of z given x can be seen as a transformation from a prior π on x to a posterior P on x given z , :

$$P(x|z) = \frac{p(z|x) \pi(x)}{\sum_{x'} p(z|x') \pi(x')}$$

Hence the property of geo-indistinguishability can be rewritten as:

$$\forall \pi. \frac{P(x|z)}{P(x'|z)} \leq e^{\epsilon d(x,x')} \frac{\pi(x)}{\pi(x')}$$

Note that $\frac{P(x|z)}{P(x'|z)}$ depends on π , but, also in this characterization, we can see that the property of geo-indistinguishability is independent from π (since π is quantified universally)

Meaning of geo-indistinguishability

$$\forall \pi. \frac{P(x|z)}{P(x'|z)} \leq e^{\epsilon d(x,x')} \frac{\pi(x)}{\pi(x')}$$

The closer two points are,
the more they are indistinguishable

The level of distinguishability
also depends on the prior

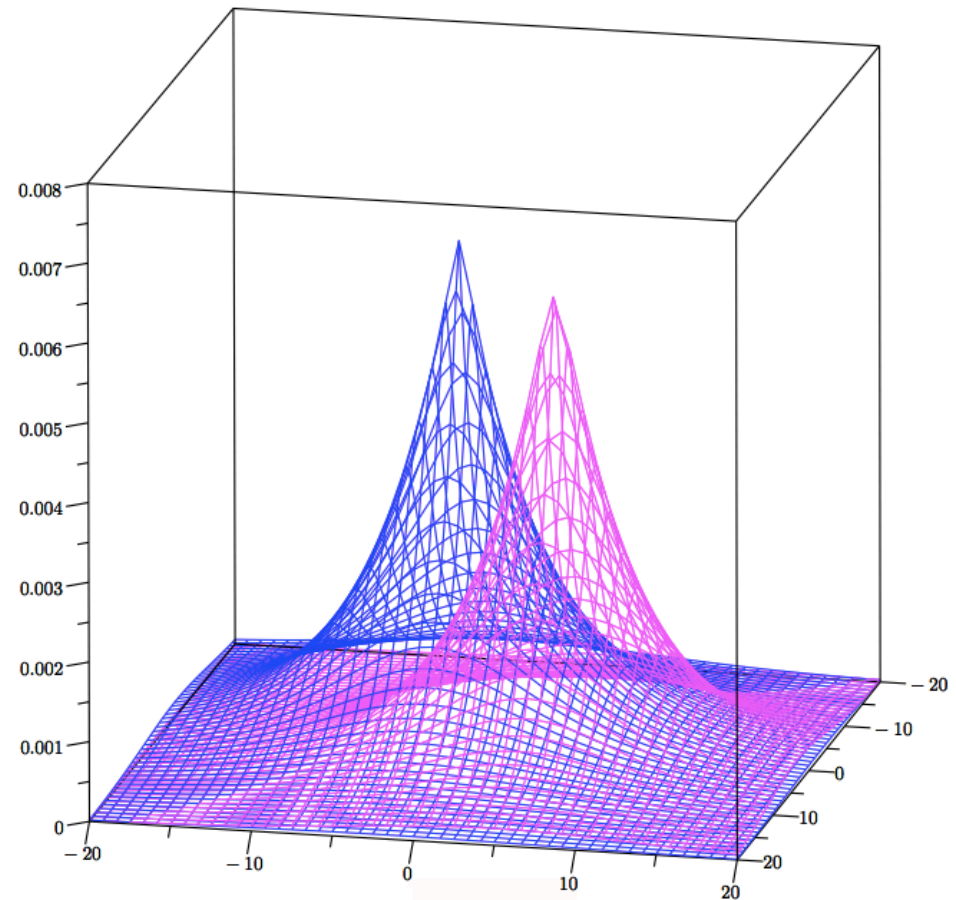
We want to be unable to tell whether the user is in rue Pigalle or at Notre Dame, but it is ok to disclose that he is in Paris and not in London

A d -private mechanism for LBS: Planar laplacian

Bivariate Laplacian

$$dp_x(z) = \frac{\epsilon^2}{2\pi} e^{\epsilon d(x,z)}$$

- We have an efficient method to draw points based on polar coordinates
- Afterwards we translate from polar coordinates to standard (latitude, longitude) coordinates.
- Some degradation of the privacy level in single precision, but negligible in double precision.



Privacy versus utility: evaluation

We have compared the trade off utility-privacy of our mechanism (Planar laplacian) with three other mechanisms in the literature:

- The Optimal Mechanism by Shokri, Theodorakopoulos, Troncoso, Hubaux, Le Boudec. [Shokri et al. CCS 2012]. Note that this mechanism is prior-dependent: it is specifically generated assuming a certain adversary (with a certain prior knowledge), using linear programming techniques. Our mechanism, in contrast, is prior-independent.
- Two prior-independent mechanisms:
 - Spatial cloacking: We partition the area of interest in zones, and instead of reporting the point, we report the zone in which the point is.
 - The mechanism of Shokri et al., generated assuming uniform prior.

Privacy versus utility: evaluation

- We have designed an “area of interest” containing $9 \times 9 = 81$ “locations”.
- For the cloaking mechanism, we have partitioned the area in 9 zones, indicated by the blue lines

1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45
46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81

Privacy versus utility: evaluation

- We configured the four mechanisms so to give the same utility, and we measured their privacy.
- **Utility:** expected distance between the true location and the reported one (utility loss) [Shroki et al., CCS 2012]
- **Privacy:** expected error of the attacker (using prior information) [Shroki et al., CCS 2012]. Note that we could not use geo-indistinguishability, because our mechanism is the only one that provide geo-indistinguishability
- Priors: concentrated over colored regions

1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45
46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81

(a)

1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45
46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81

(b)

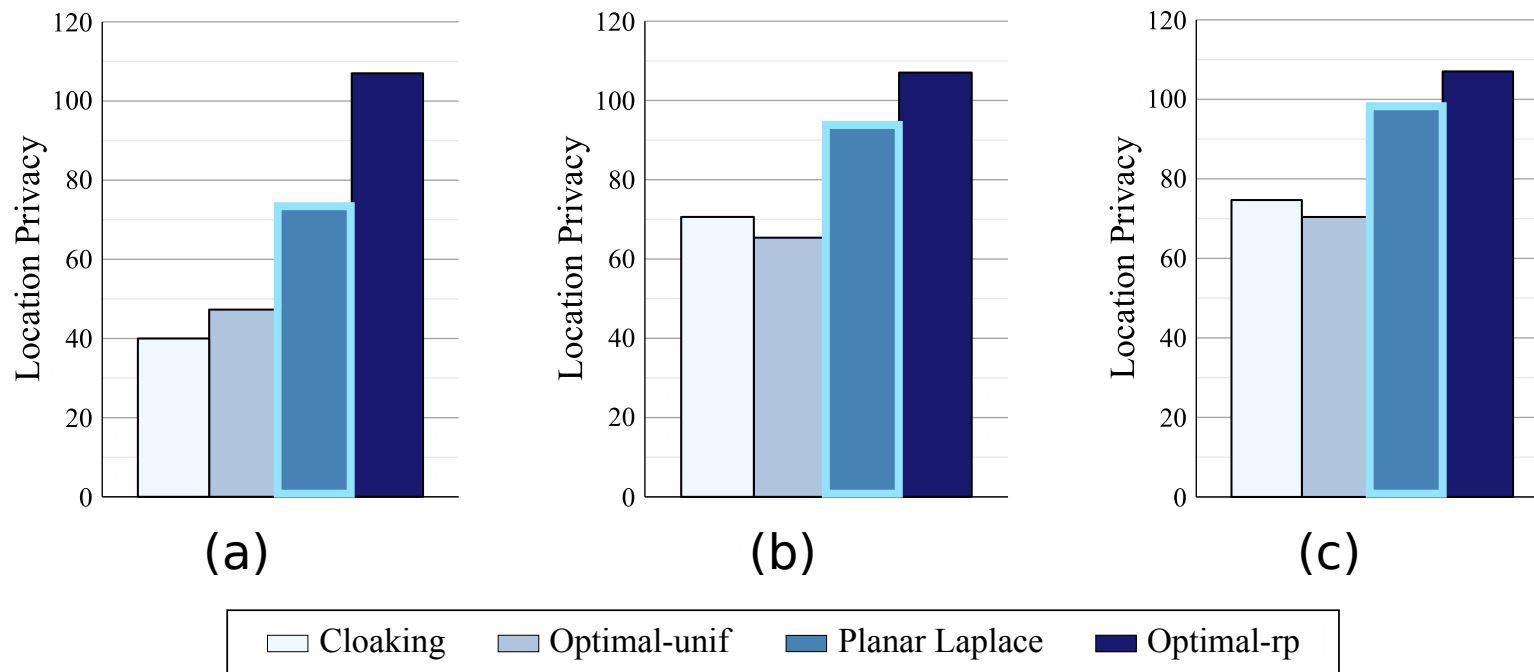
1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35	36
37	38	39	40	41	42	43	44	45
46	47	48	49	50	51	52	53	54
55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81

(c)

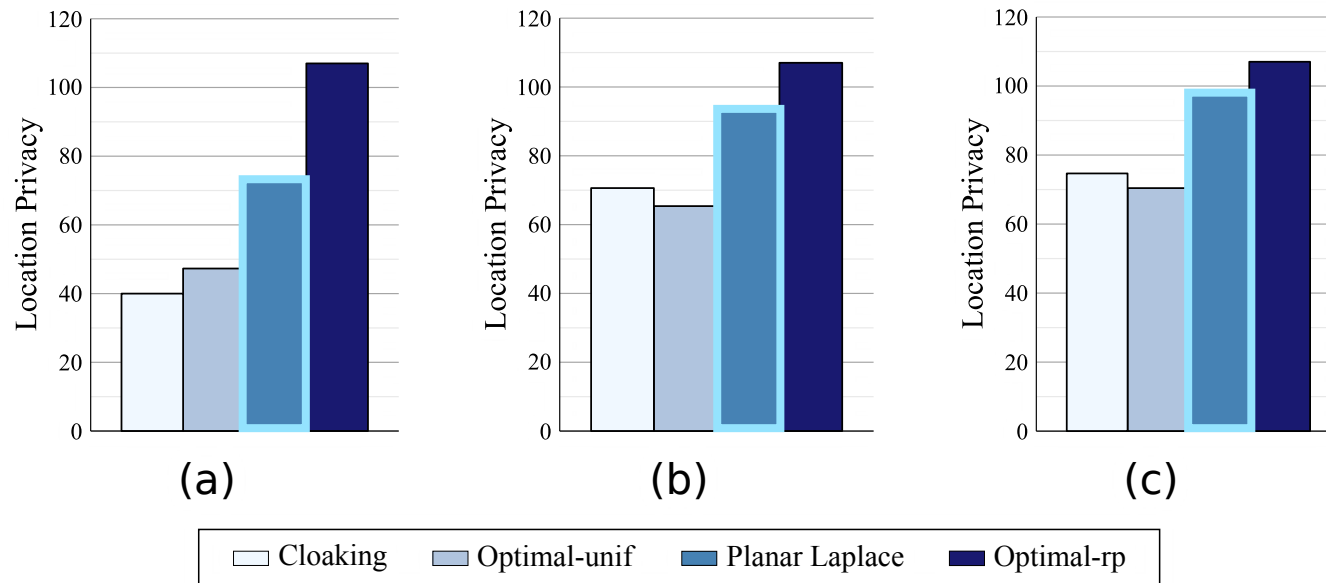
Privacy versus utility: evaluation

The four mechanisms:

- Cloaking,
- Optimal by [Shroki et al. CCS 2012] generated assuming uniform prior
- Ours (Planar Laplacian)
- Optimal by [Shroki et al. CCS 2012] generated assuming the given prior



Privacy versus utility: evaluation

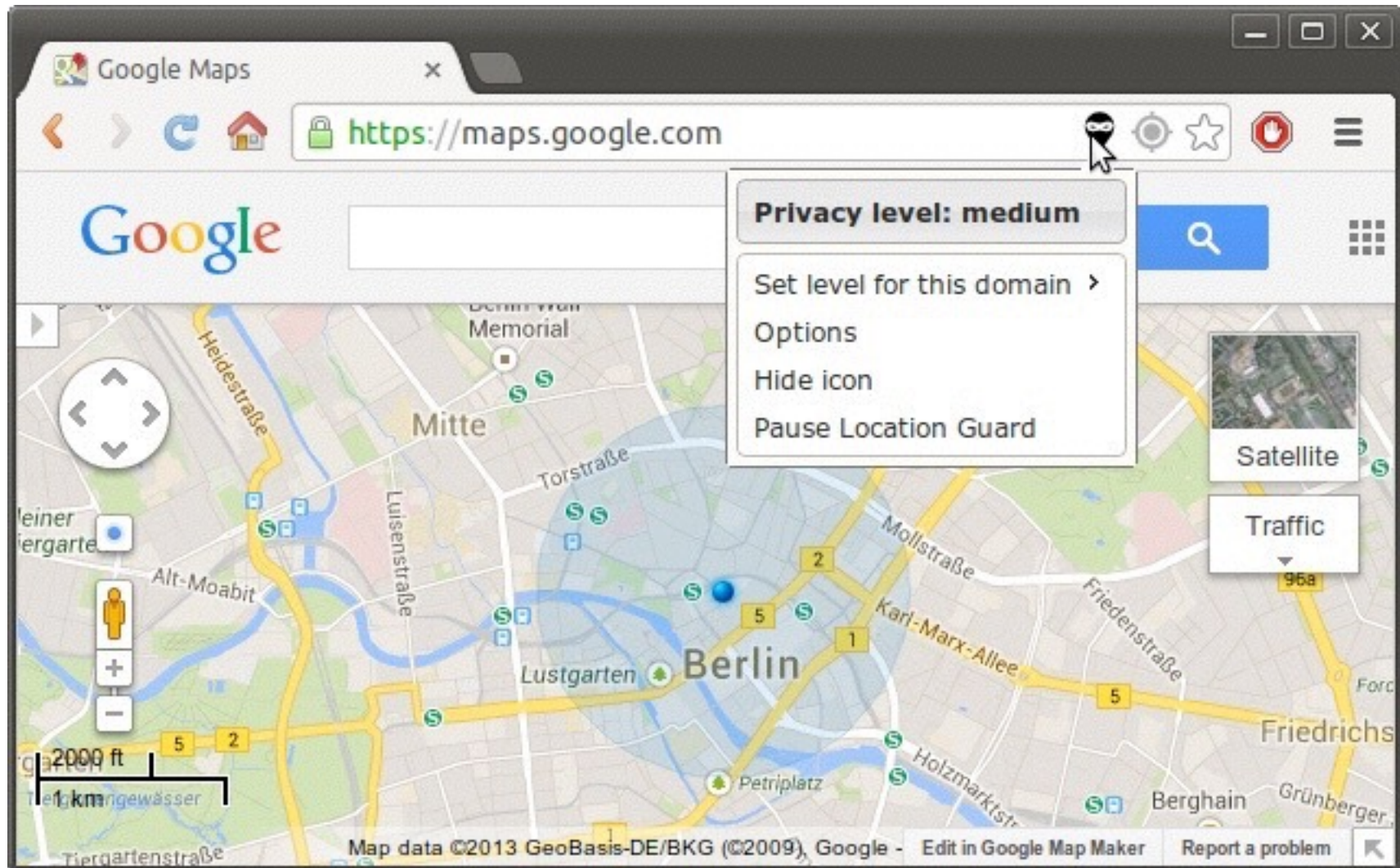


With respect to the privacy measures proposed by [Shokri et al, CCS 2012], our mechanism performs better than the other mechanisms proposed in the literature which are independent from the prior (and therefore from the adversary)

Tool: “Location Guard”

<http://www.lix.polytechnique.fr/~kostas/software.html>

Extension for Firefox, Chrome, and Opera. It has been released about one year ago, and nowadays it has about 60,000 active users.



Thank you !