

CS489/689

Privacy, Cryptography, Network and Data Security

Inference Attacks

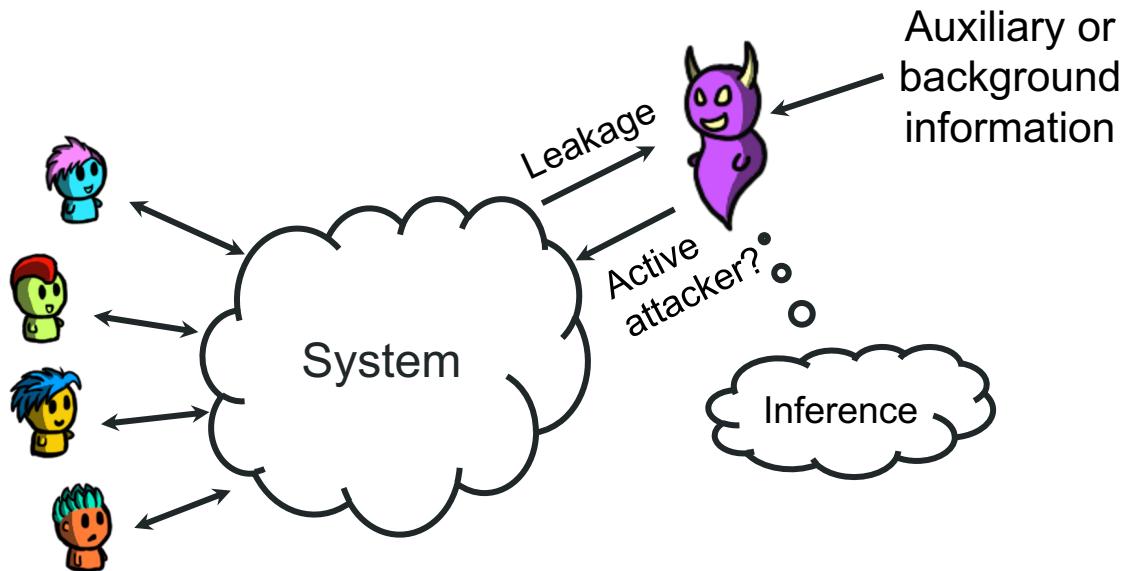
Today: More Adversarial Thinking

Specifically:

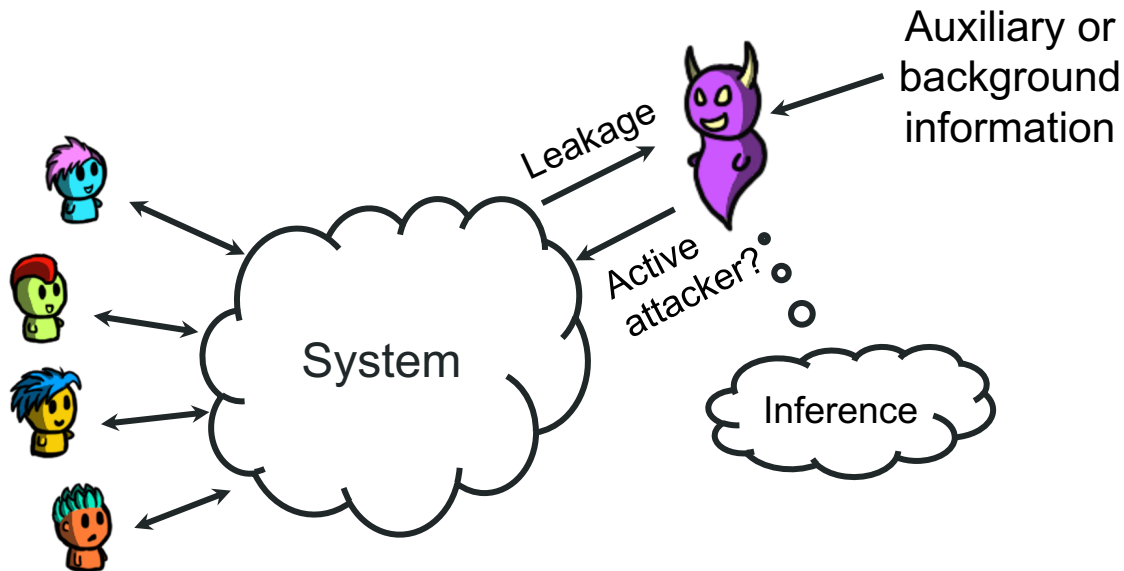
- Inference attacks
- Privacy implications of such attacks
- Examples

Inference Attacks?

What are inference attacks?

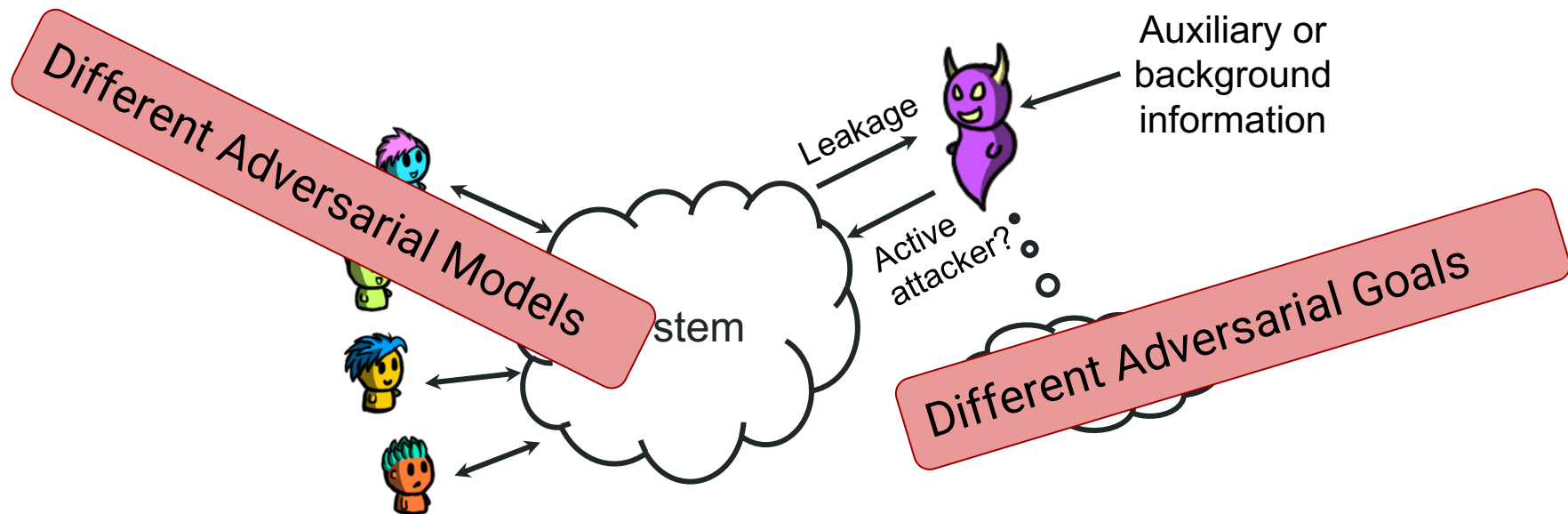


What are inference attacks?



Goal: Learn something (non-trivial) and privacy sensitive from the system

What are inference attacks?



Goal: Learn something (non-trivial) and privacy sensitive from the system

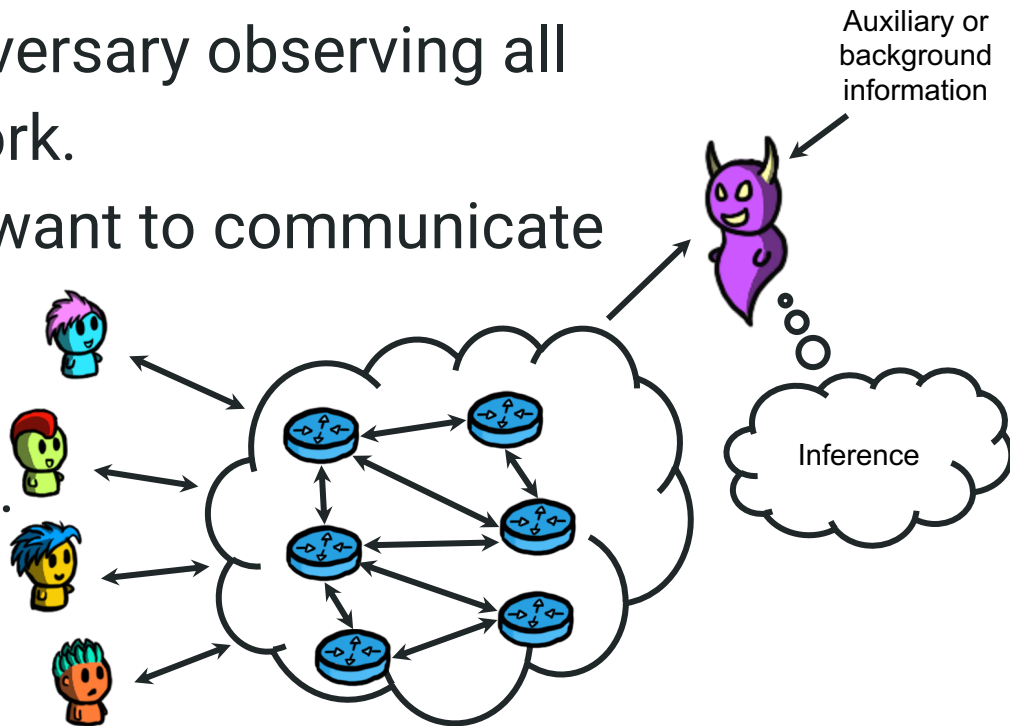
Context for Inference Attacks: The Model

- Attacks generally rely on information “leakage”
- The leakage can be intentional:
 - Sending usage statistics to a service provider (Microsoft, Apple, ...)
 - Reporting our location to Google Maps
 - Publishing census data
- Some leakage is unintentional:
 - E.g., side-channels: you saw these earlier!

Attacks can combine all leaked information with auxiliary information to infer non-trivial sensitive data!

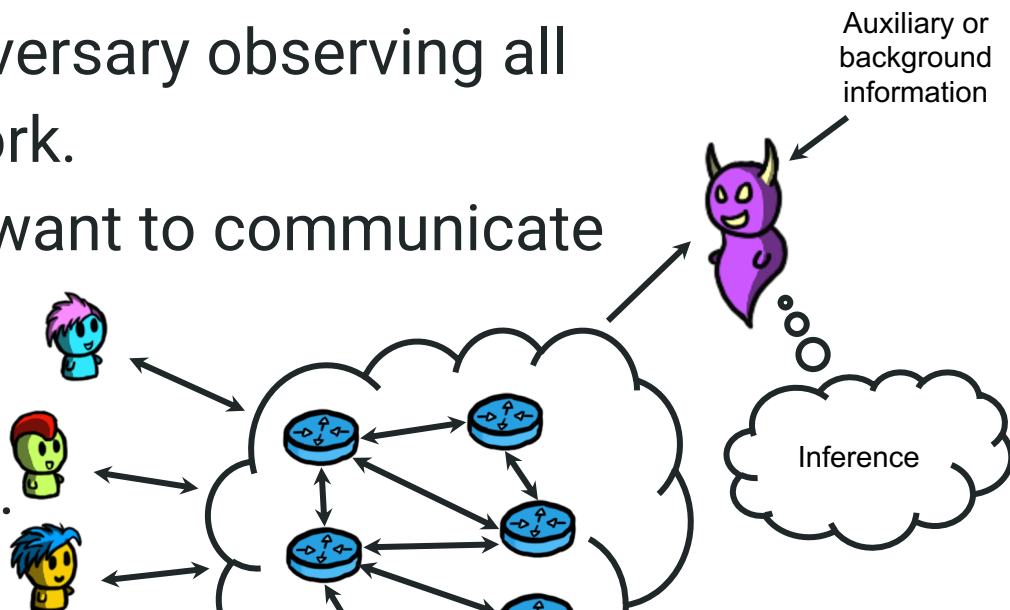
Example 1: Communication Systems

- **Adversary:** A passive adversary observing all flows of traffic in a network.
- **Functionality:** The users want to communicate with each other (they don't intend to leak anything to an adversary).



Example 1: Communication Systems

- **Adversary:** A passive adversary observing all flows of traffic in a network.
- **Functionality:** The users want to communicate with each other (they don't intend to leak anything to an adversary).



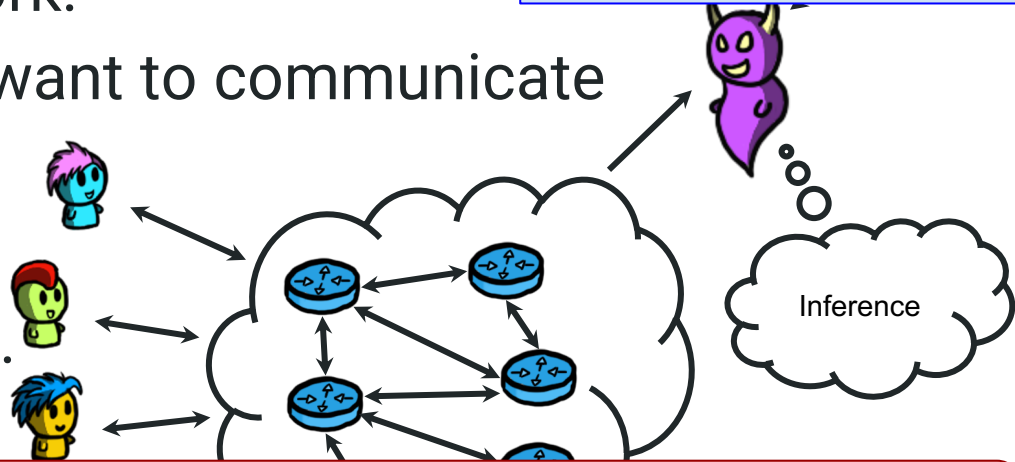
Q: What non-trivial privacy-sensitive information could the adversary infer?

Example 1: Communication Systems

- **Adversary:** A passive adversary observing all flows of traffic in a network.
- **Functionality:** The users want to communicate with each other (they don't intend to leak anything to an adversary).

Leakage:

- Packet payload
- Packet headers
- Timing information



Q: What non-trivial privacy-sensitive information could the adversary infer?

Example 1: Communication Systems

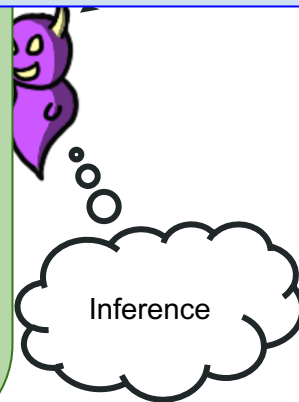
- **Adversary**
flows
- **Functionality**
with
(they
anyth

A:

- What the users are talking about
- Who is talking with whom
- The social graph of the users
- How often two users communicate
- How often a user participates in a system
- Whether or not a user communicates at all
- ...

Leakage:

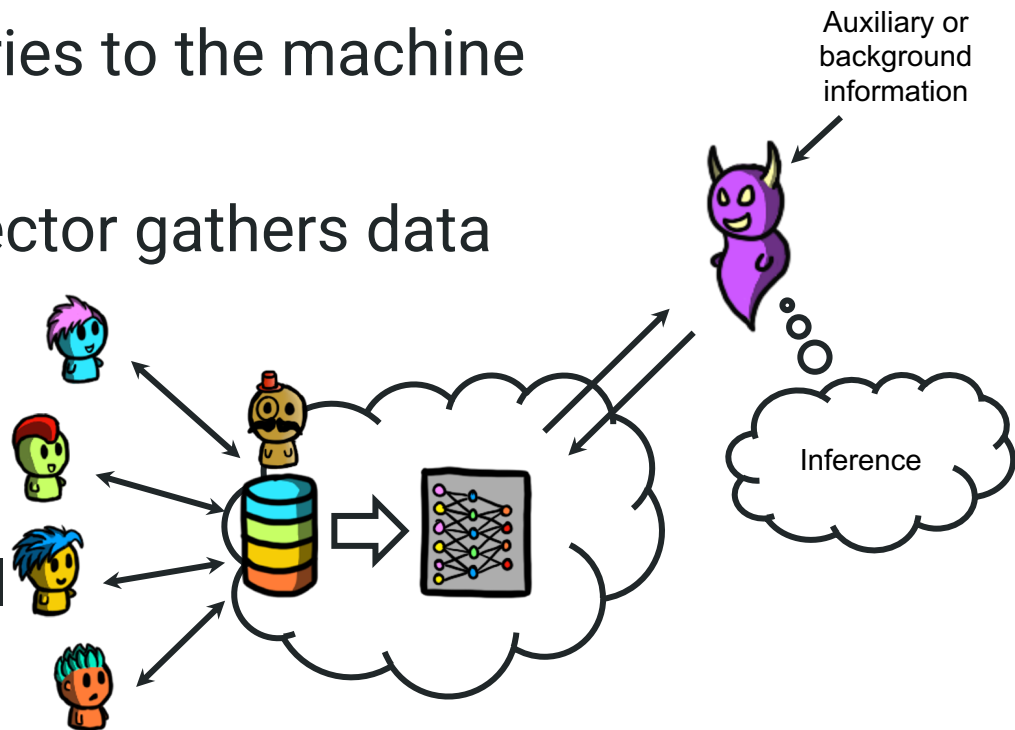
- Packet payload
- Packet headers
- Timing information



Q: What non-trivial privacy-sensitive information could the adversary infer?

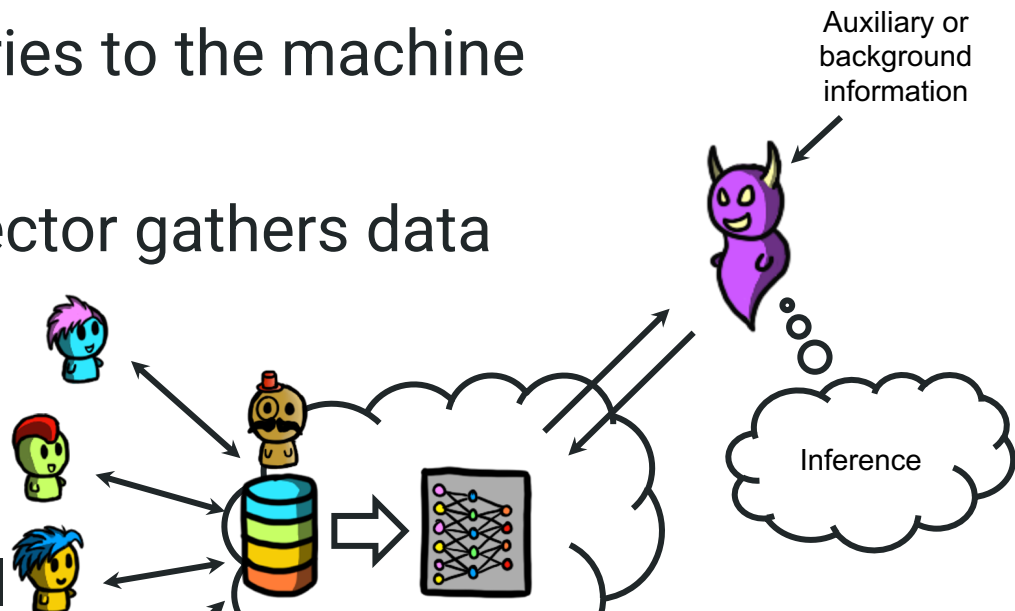
Example 2: Machine Learning

- **Adversary:** can issue queries to the machine learning model.
- **Functionality:** A data collector gathers data from users and trains a machine learning model with it (they don't intend to leak anything non-trivial to the adversary).



Example 2: Machine Learning

- **Adversary:** can issue queries to the machine learning model.
- **Functionality:** A data collector gathers data from users and trains a machine learning model with it (they don't intend to leak anything non-trivial)



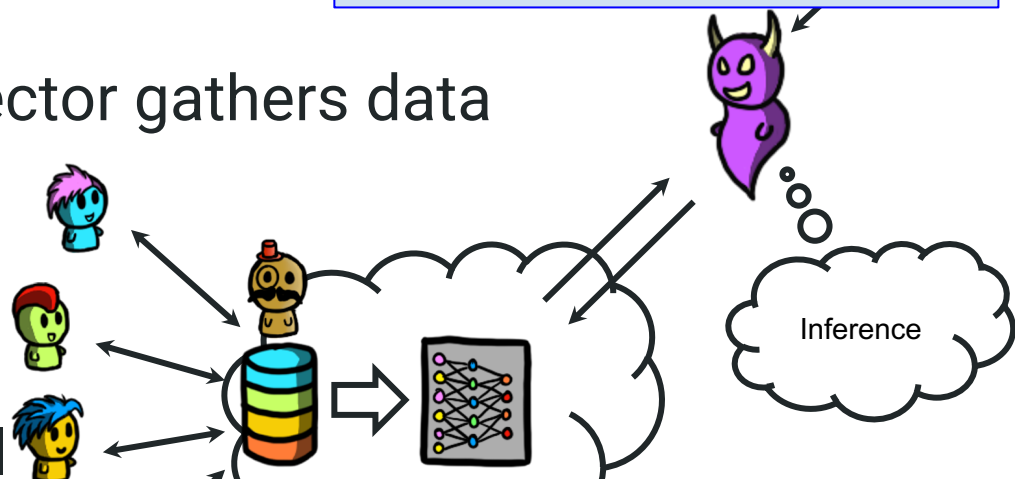
Q: What non-trivial privacy-sensitive information could the adversary infer?

Example 2: Machine Learning

- **Adversary:** can issue queries to the machine learning model.
- **Functionality:** A data collector gathers data from users and trains a machine learning model with it (they don't intend to leak anything non-trivial)

Leakage:

- Inferences from the ML model



Q: What non-trivial privacy-sensitive information could the adversary infer?

Example 2: Machine Learning

- **Adversary**

learn

- **Function**

from

mach

with i

to leak

A:

- Each user's data (the whole training dataset)
- Whether or not a particular data sample was in the training set
- A general property of the training population
- Given partial data about a user, learn other attributes about the user
- ...

Leakage:

- Inferences from the ML model



Q: What non-trivial privacy-sensitive information could the adversary infer?

Why study inference attacks?

Adversarial Thinking

- Think like an adversary to understand the ***vulnerabilities*** of a system and develop ***protection techniques***.
- When designing inference attacks, we also apply **Kerckhoff's principle** (or Shannon's maxim), adapted to privacy

Adversarial Thinking

- Think like an adversary to understand the ***vulnerabilities*** of a system and develop ***protection techniques***.
- When designing inference attacks, we also apply **Kerckhoff's principle** (or Shannon's maxim), adapted to privacy

Assume the adversary knows how the system works

- there are no hidden parameters other than the users' data
- the adversary can even know some rough distribution that the users' data follows)

Designing a System Aware of Inference Attacks

For any system that relies on users' data, there are two goals:

- **Utility:** Design a system that provides benefits to its users and the service provider
- **Privacy:** Design a system that provides protection against inference attacks

Q: What are “utility” and “privacy”? How do we “measure” them?

Designing a System Aware of Inference Attacks

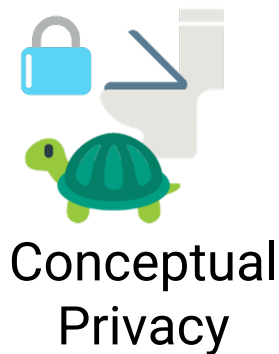
For any system that relies on users' data, there are two goals:

- **Utility:** Design a system that provides benefits to its users and the service provider
- **Privacy:** Design a system that provides protection against inference attacks

Q: What are “utility” and “privacy”? How do we “measure” them?

It's complicated...

Recall, What is privacy?



What is privacy?

- Useful definition: informational self-determination
“The right of the individual to decide what information about himself should be communicated to others and under what circumstances” (Westin, 1970)
- Privacy is having control over:
 - Who we share our data with
 - Who they can share it with
 - For what purpose they use it
 - Etc.

Quantifying Privacy?

- Protecting the sensitive information e.g., not just data, also meta-data, relationships, timing, whether a user participated in a system, etc.
- Quantifying privacy is very hard

There is **no cure-all metric** for privacy, measuring privacy can be computationally intractable, etc.

Quantifying Privacy: Theoretical Notions

- **Syntactic** notions of privacy: these are computed on the leaked or released data. They are data dependent
 - K-anonymity, l-diversity, t-closeness, etc

Quantifying Privacy: Theoretical Notions

- **Syntactic** notions of privacy: these are computed on the leaked or released data. They are data dependent
 - K-anonymity, l-diversity, t-closeness, etc
- **Semantic** notions of privacy: these are computed on the data release mechanism itself, and they hold regardless of the data (data independent)
 - Mostly Differential Privacy

Quantifying Privacy: Empirical Notions

- The performance of an **inference attack** e.g., the attacker error, accuracy, true positive rate, false positive rate, etc
- Can provide an **upper bound** on privacy

Quantifying Privacy: Empirical Notions

- The performance of an **inference attack** e.g., the attacker error, accuracy, true positive rate, false positive rate, etc
- Can provide an **upper bound** on privacy

Q: Why an upper bound?

Quantifying Privacy: Empirical Notions

- The performance of an **inference attack** e.g., the attacker error, accuracy, true positive rate, false positive rate, etc
- Can provide an **upper bound** on privacy

Q: Why an upper bound?

A: Can't get more privacy if this attack succeeds

Utility and Privacy

Utility

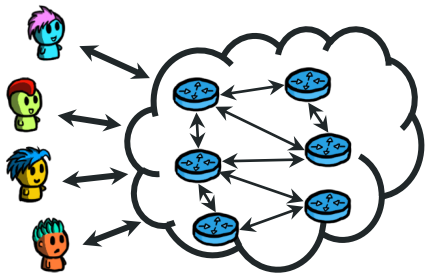
Definition: the benefit that users (and the provider) get from using the system.

Utility

Definition: the benefit that users (and the provider) get from using the system.

Communications system:

- For users: being able to communicate

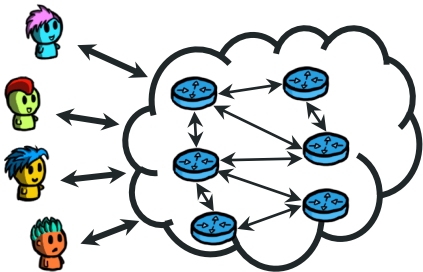


Utility

Definition: the benefit that users (and the provider) get from using the system.

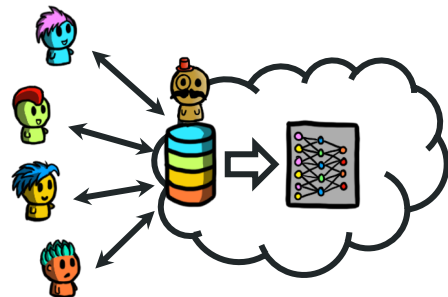
Communications system:

- For users: being able to communicate



Machine learning:

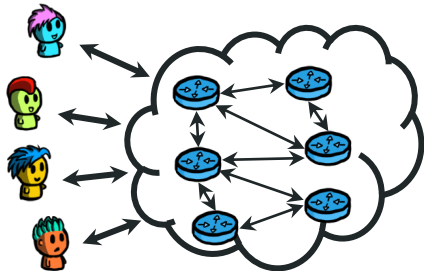
- For participants: maybe they get compensation?
- For data owner: it can sell access to the model for revenue
- Analysts: they pay to get benefits from the model's outputs
- General public: maybe the model outputs are good for society?



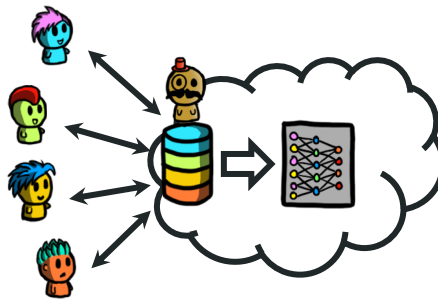
Quantifying Utility

Q: How do we *quantify* utility?

Communications system:



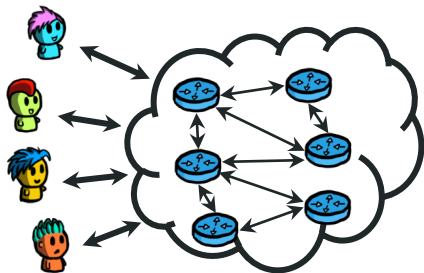
Machine learning:



Quantifying Utility

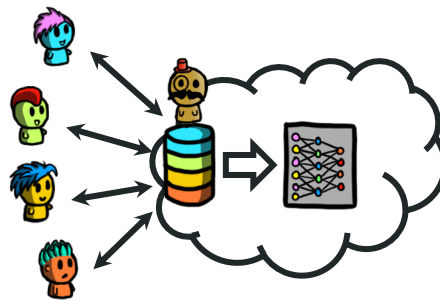
Q: How do we *quantify* utility?

Communications system:



- Low packets dropped
- High bandwidth/throughput
- Low latency/delay...

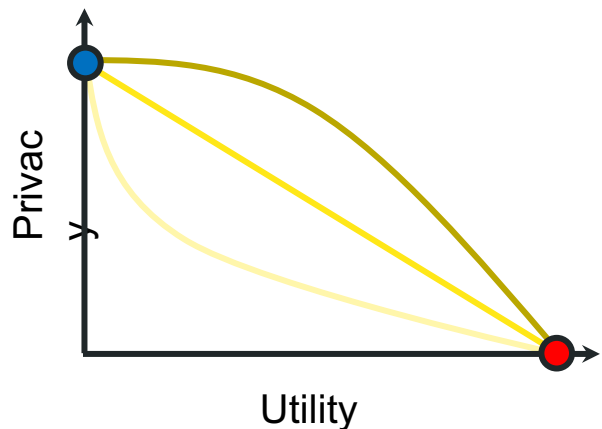
Machine learning:



- Useful model (high test accuracy)
- Unbiased model (low disparity among subpopulations)
- Low computational requirements to build the model
- Fast training algorithm...

The Privacy-Utility trade-off

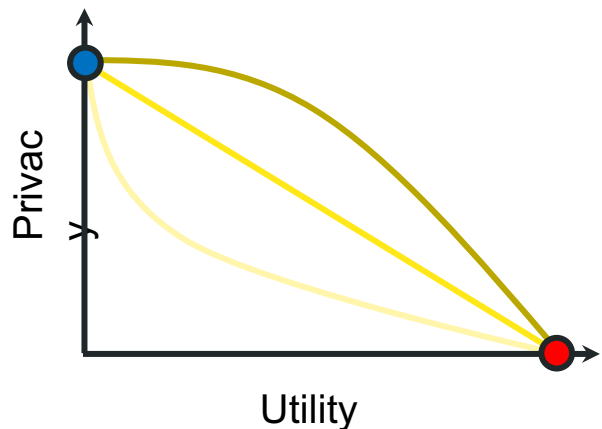
- Given any metric for privacy and for utility, they are usually at odds:



- **Q:** How do you design a system that provides **maximum utility**?
- **Q:** How do you design a system that provides **maximum privacy**?
- Designing a system that provides a good privacy-utility trade-off is hard!

The Privacy-Utility trade-off

- Given any metric for privacy and for utility, they are usually at odds:



- How do you design a system that provides **maximum utility**?
 - You design it without privacy in mind
- How do you design a system that provides **maximum privacy**?
 - You don't design it
- Designing a system that provides a good privacy-utility trade-off is hard!

Inference Attacks: Goals and Techniques

- As we saw before, the attacker can have different **goals**:
 - Infer data
 - Infer a property of the data
 - Infer the presence (membership) of some data
 - Infer the behavior of a user
 - Infer some attributes of a data sample
 - Infer dependencies among the data
 - ...

Inference Attacks: Goals and Techniques

- As we saw before, the attacker can have different **goals**:
 - Infer data
 - Infer a property of the data
 - Infer the presence (membership) of some data
 - Infer the behavior of a user
 - Infer some attributes of a data sample
 - Infer dependencies among the data
 - ...
- There are different **techniques** to perform an inference attack:
 - Statistical tools (estimation theory, detection theory, maximum likelihood, Bayesian inference...)
 - Combinatorics
 - Heuristics
 - Machine learning
 - ...

Inference Attack Examples

Inference attacks: examples

- For the rest of the lecture, we will see examples of inference attacks with different **goals** and **techniques**.
- You need to understand these attacks, their goal, the leakage they exploit and the techniques they use.
 - At the end of the lecture, you should be able to run these attacks in other examples.
 - Also, given a new system, with some leakage specification and an attack goal, you should be able to come up with reasonable privacy/utility metrics and an inference attack.

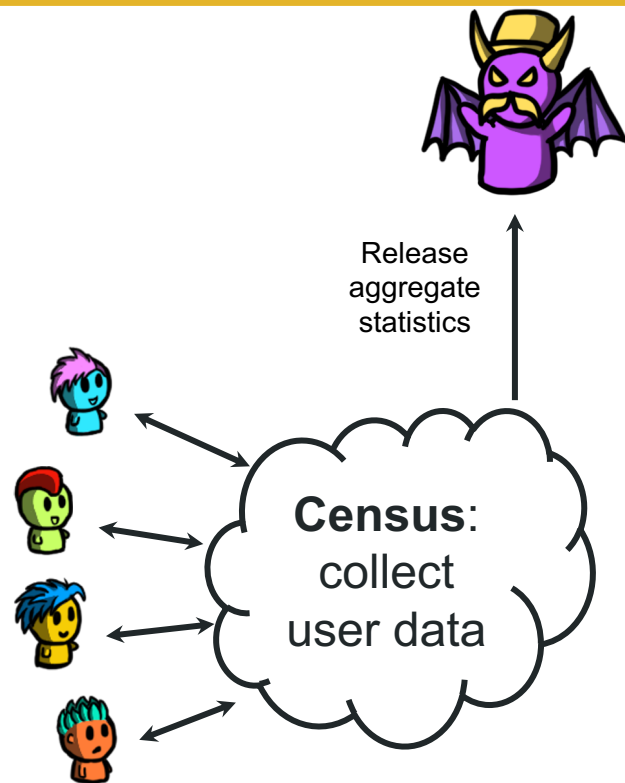
We will see:

1. Census reconstruction attacks
2. SQL inference attacks (tracker attacks)
3. Database reconstruction attacks
4. Statistical inference attacks
 - Maximum Likelihood
 - Maximum A-Posteriori
5. De-anonymization attacks
6. Side-channel attacks
7. ML Inference attacks
8. Linking attacks

1. Census Reconstruction Attacks

1. Census Reconstruction Attacks

- A census involves collecting lots of privacy-sensitive data.
- Some useful aggregate statistics are released.
- The adversary tries to infer (reconstruct) some individuals' data.



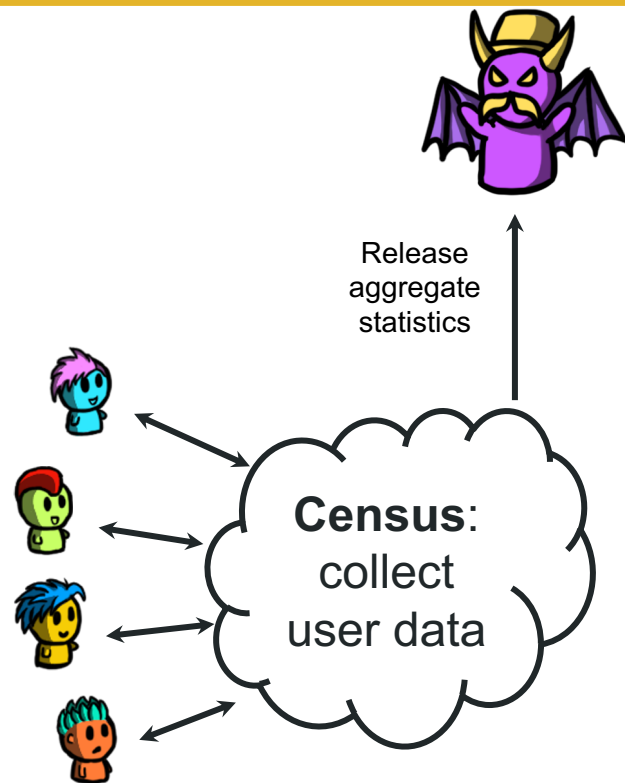
1. Census Reconstruction Attacks

- A census involves collecting lots of privacy-sensitive data.
- Some useful aggregate statistics are released.
- The adversary tries to infer (reconstruct) some individuals' data.
- Example:

Background data: adversary knows a participant that self-identifies as white is 35 years old.

Released aggregates:

	COUNT	AGE MEAN
Total population	4	24
White	2	26
Asian	2	22



1. Census Reconstruction Attacks

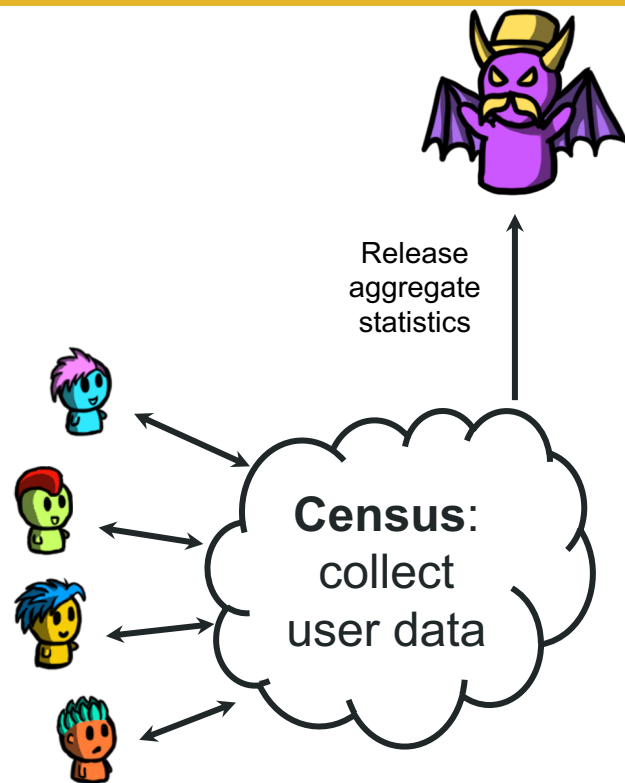
- A census involves collecting lots of privacy-sensitive data.
- Some useful aggregate statistics are released.
- The adversary tries to infer (reconstruct) some individuals' data.
- Example:

Background data: adversary knows a participant that self-identifies as white is 35 years old.

Released aggregates:

	COUNT	AGE MEAN
Total population	4	24
White	2	26
Asian	2	22

Q: Can you guess the age and self-identified race of every participant?



1. Census Reconstruction Attacks

- A census involves collecting lots of privacy-sensitive data.
- Some useful aggregate statistics are released.
- The adversary tries to infer (reconstruct) some individuals' data.
- Example:

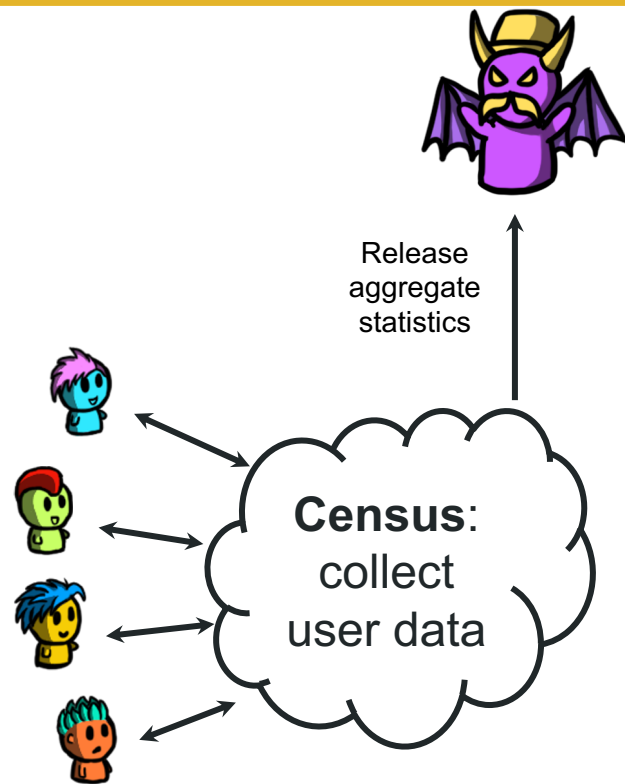
Background data: adversary knows a participant that self-identifies as white is 35 years old.

Released aggregates:

	COUNT	AGE MEAN
Total population	4	24
White	2	26
Asian	2	22

Q: Can you guess the age and self-identified race of every participant?

A: W1=17, W2=35, A1=21, A2=23



1. Census reconstruction attacks

- Another example, no background information:

Q: Can you guess the self-identified race, age, and marital status?

	COUNT	AGE MEAN	AGE MEDIAN
Total population	4	37.5	35.5
White	2	42.5	42.5
Asian	2	32.5	32.5
Single	1	25	25
Married	3	41.66	31



1. Census reconstruction attacks



- Another example, no background information:

Q: Can you guess the self-identified race, age, and marital status?

	COUNT	AGE MEAN	AGE MEDIAN
Total population	4	37.5	35.5
White	2	42.5	42.5
Asian	2	32.5	32.5
Single	1	25	25
Married	3	41.66	31

A: If you **assume the single person is Asian**, $A_1=25$, then $A_2=40$.

One white has to be $W=31$ (because that's the median of married), and the other white is $W=54$. These values meet the total population age median.

1. Census reconstruction attacks



- Another example, no background information:

Q: Can you guess the self-identified race, age, and marital status?

	COUNT	AGE MEAN	AGE MEDIAN
Total population	4	37.5	35.5
White	2	42.5	42.5
Asian	2	32.5	32.5
Single	1	25	25
Married	3	41.66	31

A: If you assume the single person is Asian, $A_1=25$, then $A_2=40$.

One white has to be $W=31$ (because that's the median of married), and the other white is $W=54$.

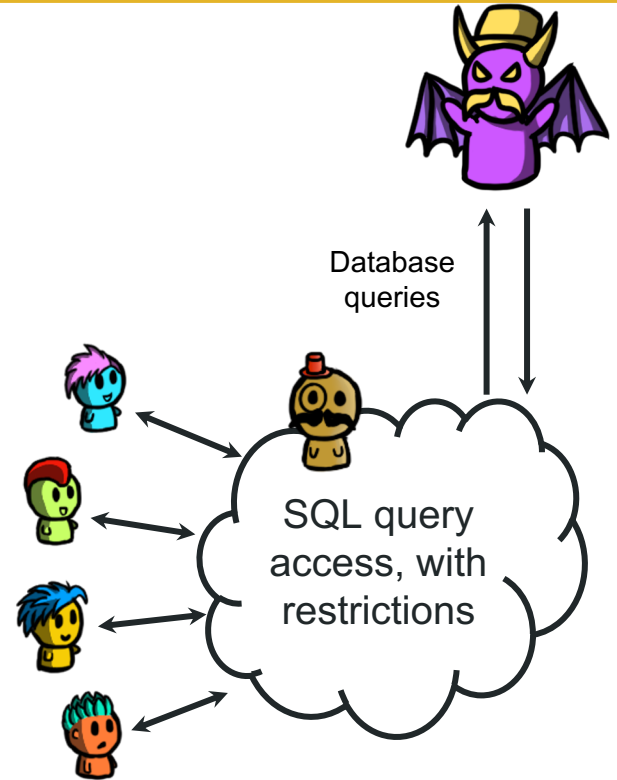
These values meet the total population age median.

If you **do the same assuming the single is White**, you get $W_1=25$, $W_2=54$, $A_1=31$, $A_2=34$, which does not meet the age median result, so it can't be true.

2. SQL Query Attacks

2. SQL query attacks

- A data collector creates a relational database (table) with data from different clients.
- An adversary can issue SQL queries to gather data from the table.
- The database management system allows queries with the following syntax:
`SELECT SUM(ATTRIBUTE) FROM (TABLE) WHERE (CONDITION)`
- However, any queries that match less than X entries or more than N-X entries are discarded.



2. SQL query attacks: example

- The table Employees has four attributes:
 - Names are unique
 - Ages are between 18 and 65
 - Position is either 'full time' or 'part time'
 - Salaries are between 50k and 500k
- You know Carol is in the dataset, and that around 50% of the people in the dataset are 'full time'.
- There are N records in the dataset; any query that matches less than $\frac{N}{10}$ or more than $\frac{9N}{10}$ entries *is discarded*.
- Can you recover Carol's salary? How many queries do you need?

Name	Age	Position	Salary
Alice	40	full time	120k
...
Carol
...

SELECT SUM(ATTRIBUTE) FROM (TABLE) WHERE (CONDITION)

2. SQL query attacks: solution

- There are N records in the dataset; any query that matches less than $\frac{N}{10}$ or more than $\frac{9N}{10}$ entries *is discarded*.

Name	Age	Position	Salary
Alice	40	full time	120k
...
Carol
...

Solution:

Q1=SELECT SUM(Salary) FROM Employees WHERE (Position='full time' OR Name=Carol)

Q2=SELECT SUM(Salary) FROM Employees WHERE (Position='full time' AND Name!=Carol)

Salary=Q1-Q2

If Carol is part time:

		Q1	Q2	Q1-Q2
Full time		■	■	
Part time				
	Carol	■		■

If Carol is full time:

		Q1	Q2	Q1-Q2
Full time		■	■	
	Carol	■		■
Part time				

Q1-Q2 always gets Carol's salary!

2. SQL query attacks:

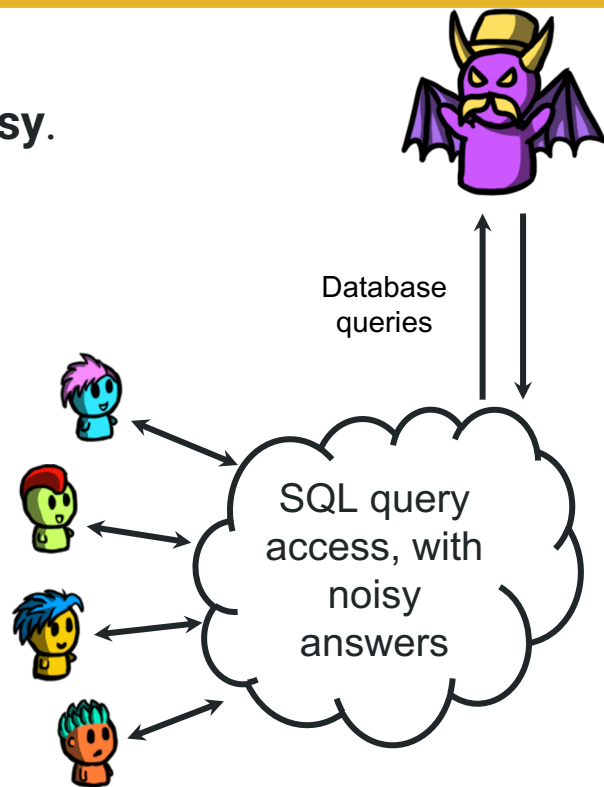
- The lesson is: even if the result of a query is harmless (too general), the combination of two or more queries can be very dangerous (very specific).
- Placing restrictions on individual queries, while still reporting exactly values, does not work.
- When coming up with SQL query attacks in this setting:
 - Look for an attribute that you can use to make sure you always bypass the restriction so that the query goes through.
 - After you design the queries, check that they get the desired value regardless of the values of other attributes in the dataset (e.g., whether Carol was full or part time in the previous example)

3. Database Reconstruction Attacks: Dinur-Nissim

3. Database Reconstruction Attacks: Dinur-Nissim

- Now we are going to see an example where the adversary can issue queries but the answers are **noisy**.
- We consider the case where the adversary knows everything in the database except for one binary attribute, e.g.,

Name	Age	Position	Salary
Alice	40	?	120k
Bob	40	?	80k
Carol	32	?	150k
Dave	21	?	80k



3. Database Reconstruction Attacks: Dinur-Nissim

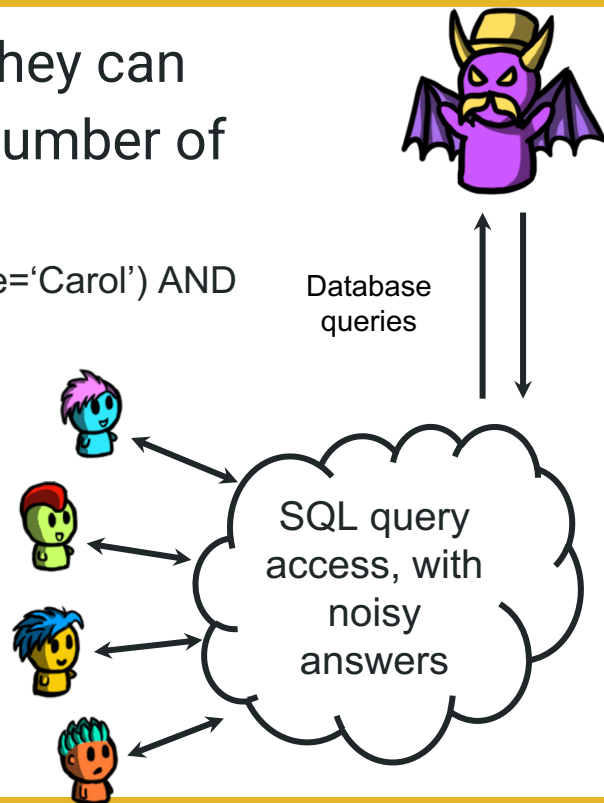
- Since the adversary knows the primary key, they can craft a condition that matches any specific number of rows, e.g.,

```
SELECT COUNT(*) FROM Employees WHERE (Name='Alice' OR Name='Carol') AND Position='full time'
```

Name	Age	Position	Salary
Alice	40	?	120k
Bob	40	?	80k
Carol	32	?	150k
Dave	21	?	80k

True output:

- 0 if non are full time
 - 1 if one is full time
 - 2 if both are full time
- (But the system will only report noisy outputs)



3. Dinur-Nissim attack: example

- Example: the DBMS adds noise uniformly chosen between -1, 0, 1.
- The server queries for the sum of rows that have 'full time' AND a certain combination of names, and does this for every possible combination of names. These are the

Name	Position
Alice	?
Bob	?
Carol	?

Rows	Binary representation	Output
Alice	001	2
Bob	010	1
Bob, Alice	011	0
Carol	100	1
Carol, Alice	101	2
Carol, Bob	110	2
Carol, Bob, Alice	111	1

Q: Can you tell who is full time and who is part time?

3. Dinur-Nissim attack: example

- Example: the DBMS adds noise uniformly chosen between -1, 0, 1.
- The server queries for the sum of rows that have ‘full time’ AND a certain combination of names, and does this for every possible combination of names. These are the

Name	Position
Alice	?
Bob	?
Carol	?

Rows	Binary representation	Output
Alice	001	2
Bob	010	1
Bob, Alice	011	0
Carol	100	1
Carol, Alice	101	2
Carol, Bob	110	2
Carol, Bob, Alice	111	1

Q: Can you tell who is full time and who is part time?

Attack idea: write all possible “candidate” databases and, for each query, see if there are any candidates that are not possible. The goal is to reach a small set of possible candidates.

3. Dinur-Nissim: solution

Name	Position
Alice	?
Bob	?
Carol	?

This means:
Bob and Carol
are 'full time',
Alice is 'part
time'

Binary rep	Output
001	2
010	1
011	0
100	1
101	2
110	2
111	1

Candidates:	000	001	010	011	100	101	110	111
Query 001=2	No		No		No		No	
Query 010=1								
Query 011=0				No				No
Query 100=1								
Query 101=2								
Query 110=2		No						
Query 111=1								

Noise: choose randomly in the set [+1, +0, -1]

Attack idea: write all possible “candidate” databases and, for each query, see if there are any candidates that are not possible. The goal is to reach a small set of possible candidates.

Solution: the only plausible candidate is 101, which means Alice and Carol are full time, and Bob is part time!

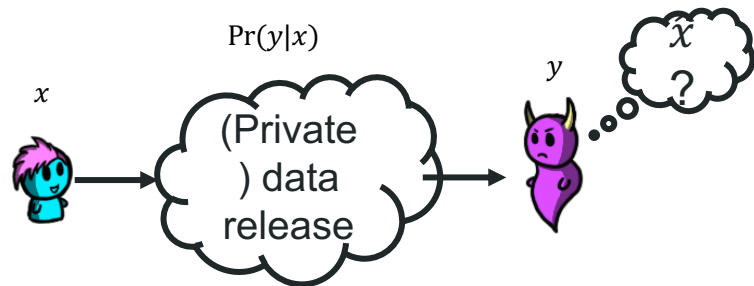
4. Statistical Inference: Probability recap

4. Statistical Inference: Probability recap

- The following attacks require some basic knowledge of probability and statistics. Let's do a recap.
- For simplicity, we assume *discrete* random variables here.
- x is Alice's private information, y is the leakage; usually \hat{x} is the adversary's estimate of x .
- $\Pr(x)$: the *prior* probability distribution of Alice's secret value
- $\Pr(y|x)$: the *mechanism* that models the leakage given Alice's secret information
 - In Bayesian inference, $\Pr(y|x)$ is also called the *likelihood* (of x having generated y)
- $\Pr(x|y)$: the *posterior* probability distribution (the probability that x took a certain value given the observed leakage y)
- **Bayes' theorem** connects these concepts:

$$\Pr(x|y) = \frac{\Pr(y|x) \cdot \Pr(x)}{\Pr(y)}$$

- **Law of total probability:** $\Pr(y) = \sum_x \Pr(x) \Pr(y|x)$



4. Statistical Inference: Probability recap

- Recall the expected value of a random variable:

$$E\{x\} = \sum_x x \cdot \Pr(x)$$

- When the adversary sees y , they can compute the conditional expectation of x (leveraging the leakage y):

$$E\{x|Y = y\} = \sum_x x \cdot \Pr(x|y)$$

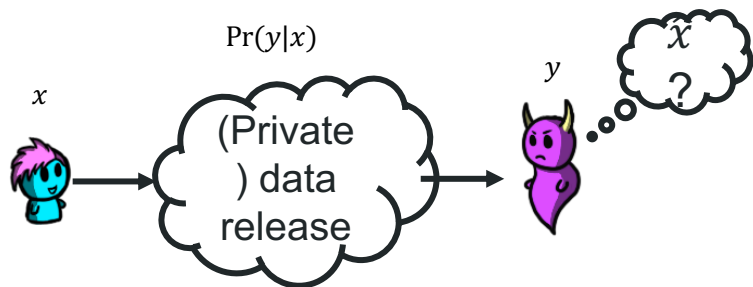
- Given y , $\Pr(x)$, and $\Pr(y|x)$, how do we run an attack (i.e., find x)?
 - There are many options!

4. Statistical Inference: Maximum Likelihood

- The **Maximum Likelihood** (ML) approach simply looks for the x that is *most likely* to have generated y , i.e.,

$$\hat{x} = \operatorname{argmax}_x \Pr(y|x)$$

Q: what is the downside of this?



4. Statistical Inference: Maximum Likelihood

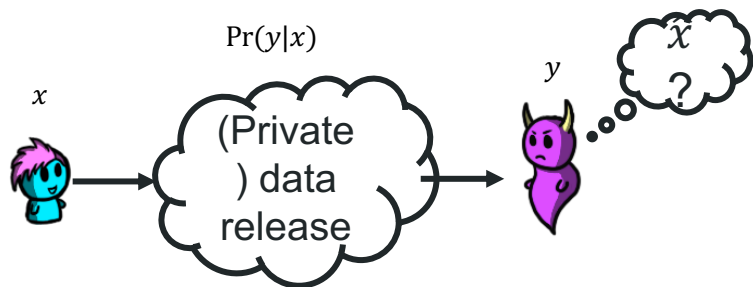
- The **Maximum Likelihood** (ML) approach simply looks for the x that is *most likely* to have generated y , i.e.,

$$\hat{x} = \operatorname{argmax}_x \Pr(y|x)$$

Q: what is the downside of this?

A: Maybe that x had a very low prior probability...

- However, if the adversary does not know the prior, this is reasonable.
- No need to compute the posterior!



4. Statistical Inference: Maximum A-Posteriori

- The **Maximum A-Posteriori** (MAP) approach chooses the x that maximizes the posterior probability:

$$\hat{x} = \operatorname{argmax}_x \Pr(x|y)$$

- **Q:** Expand the posterior and simplify the expression:

$$\hat{x} = \operatorname{argmax}_x \Pr(x|y) = \operatorname{argmax}_x \Pr(x) \cdot \Pr(y|x)$$

- This is like ML, but taking into account the posterior.
- Note that we do not need to compute $\Pr(y)$!
- **Q:** when are MAP and ML equivalent?
- When the prior is uniform! (every secret value x is just as likely)

4. Statistical Inference: other attacks

- MAP and ML choose an x that maximizes a probability. Sometimes the attacker just wants to get an x that is “as close as possible” to the real x .
- Let $d(x, \hat{x})$ be a distance measuring how different x and \hat{x} are.

Q: What is the estimation of x (i.e., \hat{x}), that *minimizes* the *average distance* to x ?

4. Statistical Inference: other attacks

- MAP and ML choose an x that maximizes a probability. Sometimes the attacker just wants to get an x that is “as close as possible” to the real x .
- Let $d(x, \hat{x})$ be a distance measuring how different x and \hat{x} are.

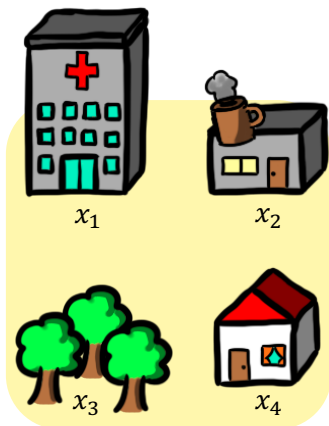
Q: What is the estimation of x (i.e., \hat{x}), that *minimizes the average (expected) distance* to x ?

A:

$$\hat{x} = \operatorname{argmin}_{x'} E\{d(x, x')\} = \operatorname{argmin}_{x'} \sum_x \sum_y \Pr(x) \cdot \Pr(y|x) \cdot d(x, x')$$

Statistical Inference example: location privacy

- Alice wants to query for a location-based service, without revealing her real location x to the service provider. She runs a randomized mechanism $\Pr(y|x)$ and reports an obfuscated location y .
- Consider all locations are in a discrete set of only 4 possible locations: a hospital (x_1), a café (x_2), a forest (x_3), and a house (x_4).



Pol	Coordinates	$\Pr(x)$
x_1	(0,1)	0.2
x_2	(1,1)	0.4
x_3	(0,0)	0.1
x_4	(1,0)	0.3

$\Pr(y x)$	$y = x_1$	$y = x_2$	$y = x_3$	$y = x_4$
$x = x_1$	0.5	0.2	0.2	0.1
$x = x_2$	0.2	0.5	0.1	0.2
$x = x_3$	0.2	0.1	0.5	0.2
$x = x_4$	0.1	0.2	0.2	0.5

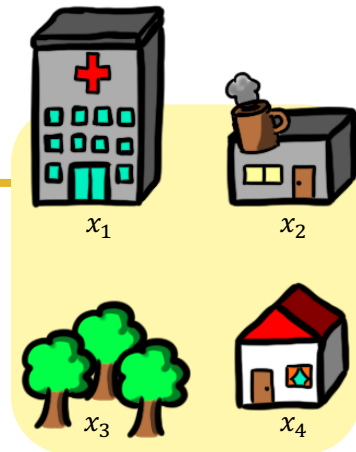
Alice reports that she is in the forest ($y = x_3$).

Q: What are the ML and MAP estimates of x ?

Location privacy: solutions

Pol	Coordinates	Pr(x)
x_1	(0,1)	0.2
x_2	(1,1)	0.4
x_3	(0,0)	0.1
x_4	(1,0)	0.3

Pr(y x)	$y = x_1$	$y = x_2$	$y = x_3$	$y = x_4$
$x = x_1$	0.5	0.2	0.2	0.1
$x = x_2$	0.2	0.5	0.1	0.2
$x = x_3$	0.2	0.1	0.5	0.2
$x = x_4$	0.1	0.2	0.2	0.5

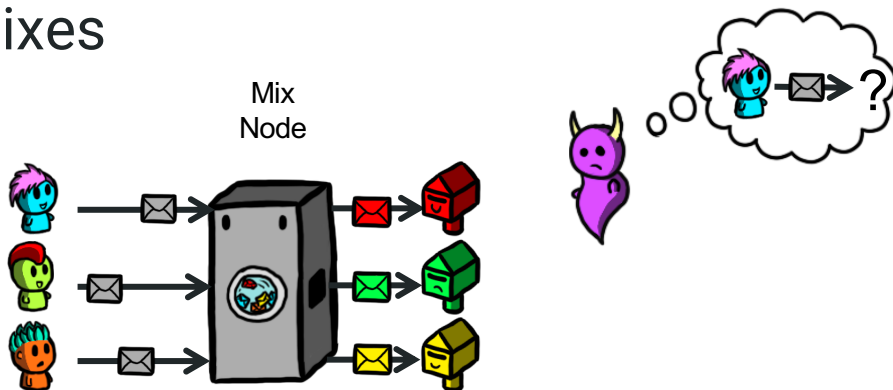


- ML: $\hat{x} = \operatorname{argmax}_x \Pr(y|x) = x_3$
- MAP: $\hat{x} = \operatorname{argmax}_x \Pr(x) \cdot \Pr(y|x) = x_4$
- Minimum Euclidean: this one requires more thinking... which you will have to do in the assignment

5. De-Anonymization Attacks

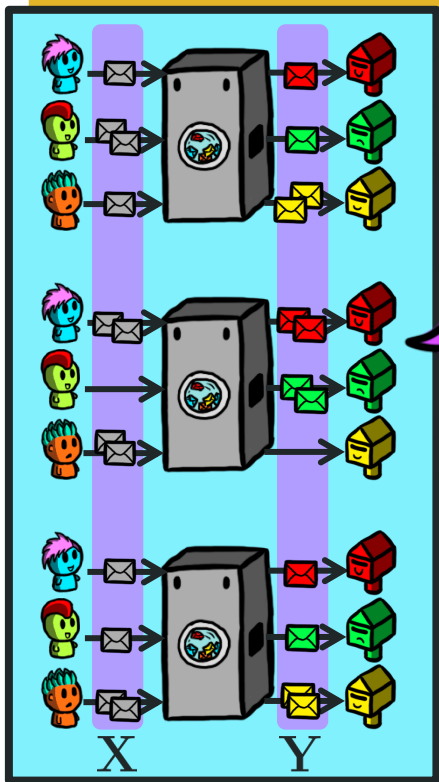
5. De-Anonymization Attacks

- Recall: mixes



- Attack:** given the observations, learn something private.
- There are many possible *goals* and *techniques*. We will see one *statistical disclosure attack*.

5. De-Anonymization Attacks: profiling



Sending profiles?

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Pr(sends to)

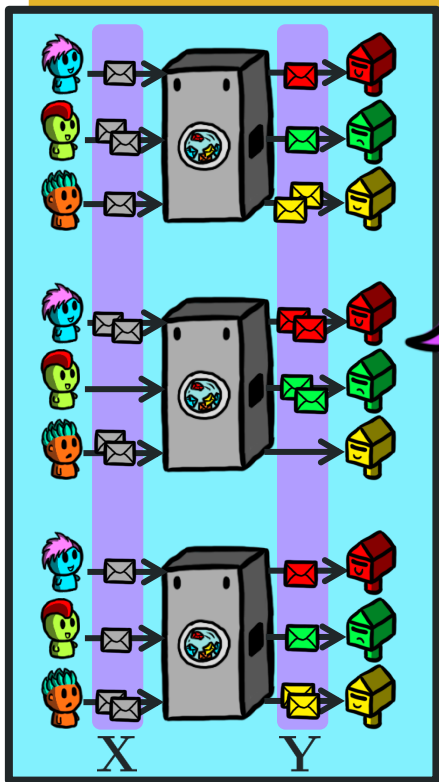
Adversary's observations:

$$X = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix}$$

- The attack's goal is to learn the sending profiles P . This is the privacy-sensitive variable.
- The observations are X and Y .

Q: Given these observations and the target privacy-sensitive variable, what would be the MAP attack?

5. De-Anonymization Attacks: profiling



Sending profiles?

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Pr(sends to)

Adversary's observations:

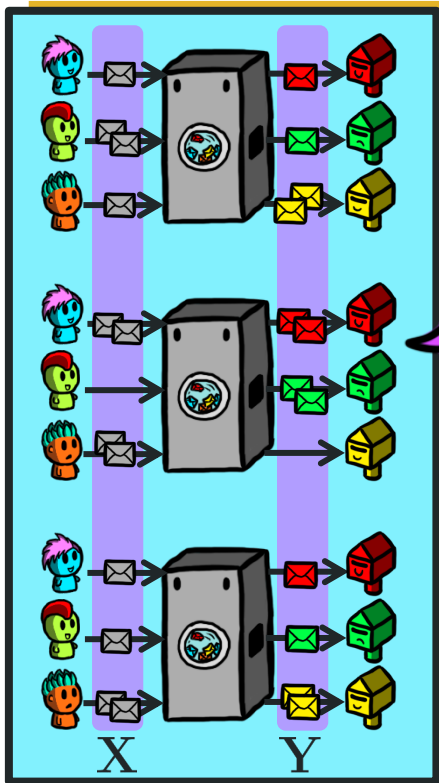
$$X = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix}$$

- The attack's goal is to learn the sending profiles P . This is the privacy-sensitive variable.
- The observations are X and Y .

Q: Given these observations and the target privacy-sensitive variable, what would be the MAP attack?

$$\mathbf{A:} \hat{P} = \operatorname{argmax}_P \Pr(P|X, Y)$$

5. De-Anonymization Attacks: profiling



Sending profiles?

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Pr(sends to)

Adversary's observations:

$$X = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix}$$

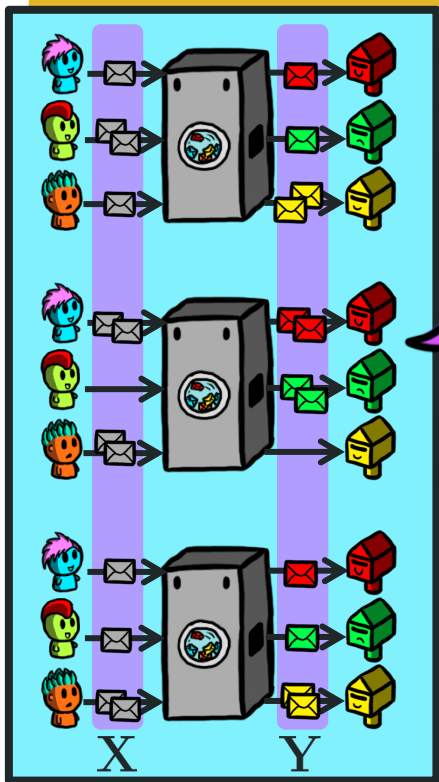
- The attack's goal is to learn the sending profiles P . This is the privacy-sensitive variable.
- The observations are X and Y .

Q: Given these observations and the target privacy-sensitive variable, what would be the MAP attack?

$$\mathbf{A:} \hat{P} = \operatorname{argmax}_P \Pr(P|X, Y)$$

Q: Do you see any problems with this attack?

5. De-Anonymization Attacks: profiling



Sending profiles?

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Pr(sends to)

Adversary's observations:

$$X = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix}$$

- The attack's goal is to learn the sending profiles P . This is the privacy-sensitive variable.
- The observations are X and Y .

Q: Given these observations and the target privacy-sensitive variable, what would be the MAP attack?

$$\mathbf{A:} \hat{P} = \operatorname{argmax}_P \Pr(P|X, Y)$$

Q: Do you see any problems with this attack?

A:

- How do you get a prior $\Pr(P)$?
- More importantly, how do you even model the probability $\Pr(X, Y|P)$?
- We can assume X and P are independent, but we'd still have to model $\Pr(Y|X, P)$, which is untractable!

5. De-Anonymization attacks: LSDA (I)

$$\mathbf{P} = \begin{matrix} \begin{matrix} \text{Red} & \text{Green} & \text{Yellow} \\ \text{Head} & \text{Head} & \text{Head} \end{matrix} \\ \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix} \begin{matrix} \text{Pink} \\ \text{Green} \\ \text{Orange} \\ \text{Head} \\ \text{Head} \\ \text{Head} \end{matrix} \end{matrix}$$

Adversary's observations:

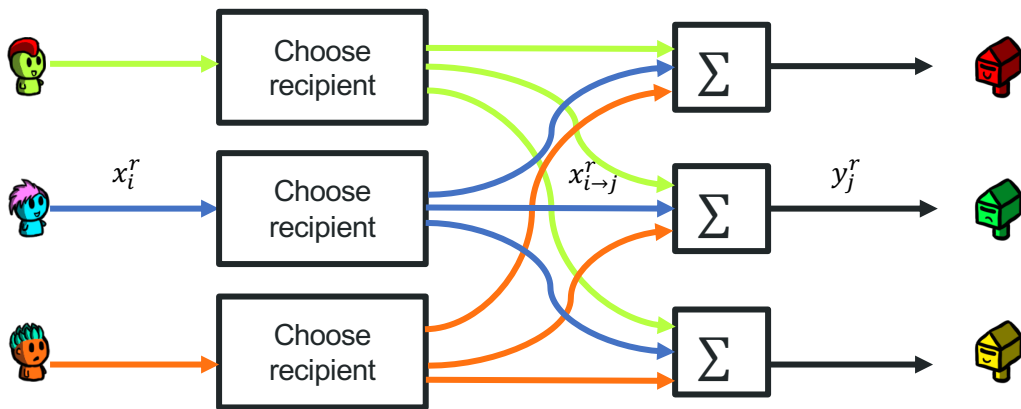
$$\mathbf{X} = \begin{matrix} \begin{matrix} \text{Pink} & \text{Green} & \text{Orange} \\ \text{Head} & \text{Head} & \text{Head} \end{matrix} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \end{matrix} \quad \mathbf{Y} = \begin{matrix} \begin{matrix} \text{Red} & \text{Green} & \text{Yellow} \\ \text{Head} & \text{Head} & \text{Head} \end{matrix} \\ \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots \end{bmatrix} \end{matrix}$$

- Solution: approximate approach, do an average-case analysis to develop an attack.
- We will see just one attack: the Least-Squares Disclosure Attack (LSDA) [1]
- The idea of this attack is to compute the *expected value of the outputs* Y , given X and P , i.e.,
$$E\{Y|X, P\}$$
- Then, it finds the P that minimizes the mean squared error between the observed Y and $E\{Y|X, P\}$, i. e.,
$$\hat{P} = \operatorname{argmin}_P \|Y - E\{Y|X, P\}\|_2^2$$
- Let's see how to compute this expected value...



[1] F. Pérez-González, and Carmela Troncoso. "Understanding statistical disclosure: A least squares approach." PETS 2012

Deriving LSDA

- Let's consider a threshold or timed mix (no delay between rounds).
- Sender i sends x_i^r messages in communication round r .
- Out of those x_i^r messages, $x_{i \rightarrow j}^r$ is the number that are for receiver j .
- Receiver j receives y_j^r messages in communication round r .



$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

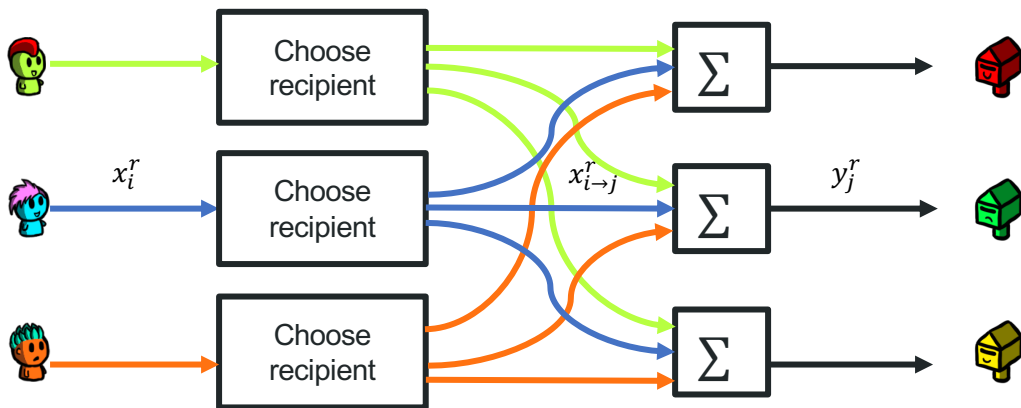
Pr( sends to )

The matrix P is a 3x3 matrix of probabilities. The element $p_{1,2}$ is highlighted with a red box. A red arrow points from the text above to the $p_{1,2}$ element.

Q: If the senders choose the recipient of each of their messages *independently* following the sending profiles in P , what is the probability distribution of $x_{i \rightarrow j}^r$ given x_i^r and P ?

Deriving LSDA

- Let's consider a threshold or timed mix (no delay between rounds).
- Sender i sends x_i^r messages in communication round r .
- Out of those x_i^r messages, $x_{i \rightarrow j}^r$ is the number that are for receiver j .
- Receiver j receives y_j^r messages in communication round r .



$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

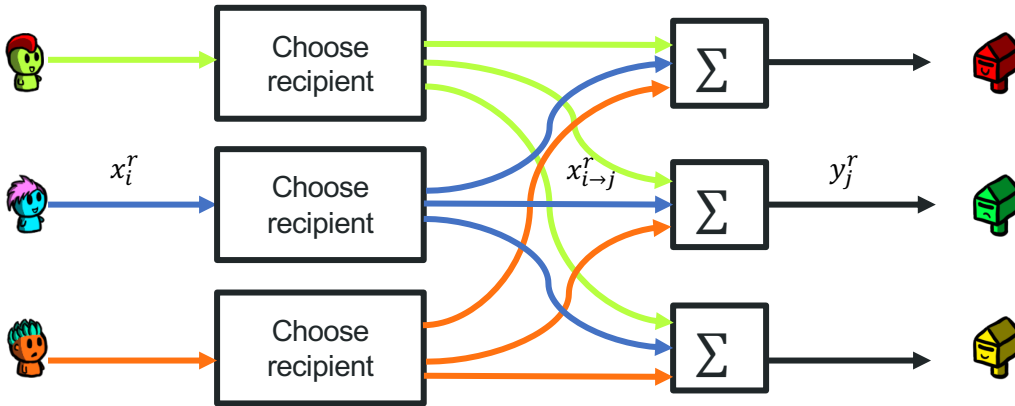
$\Pr(\text{👤 sends to 📡})$

Q: If the senders choose the recipient of each of their messages *independently* following the sending profiles in P , what is the probability distribution of $x_{i \rightarrow j}^r$ given x_i^r and P ?

A: $x_{i \rightarrow j}^r \sim \text{Bino}(x_i^r, p_{i,j})$

Deriving LSDA

- Let's consider a threshold or timed mix (no delay between rounds).
- Sender i sends x_i^r messages in communication round r .
- Out of those x_i^r messages, $x_{i \rightarrow j}^r$ is the number that are for receiver j .
- Receiver j receives y_j^r messages in communication round r .

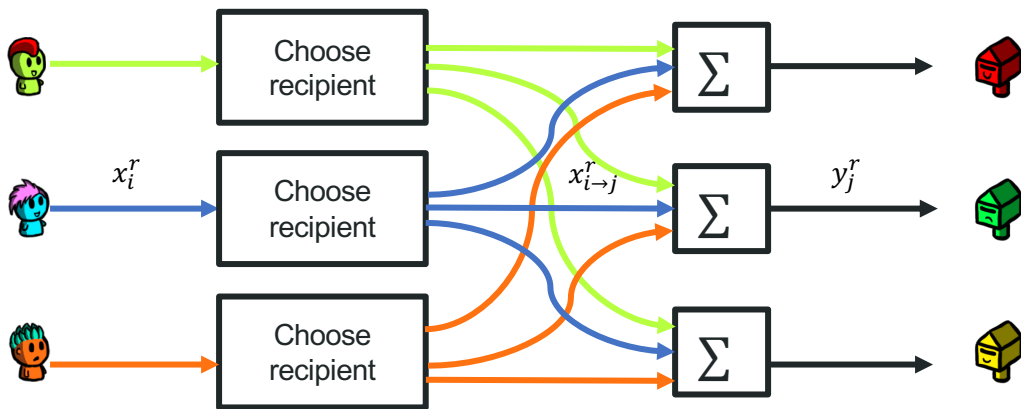


$$\mathbf{P} = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Q: Since y_j^r is the messages received by j in round r (from all the senders!!). What is each distribution?

Deriving LSDA

- Let's consider a threshold or timed mix (no delay between rounds).
- Sender i sends x_i^r messages in communication round r .
- Out of those x_i^r messages, $x_{i \rightarrow j}^r$ is the number that are for receiver j .
- Receiver j receives y_j^r messages in communication round r .



$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$

Pr(👤 sends to 🏠)

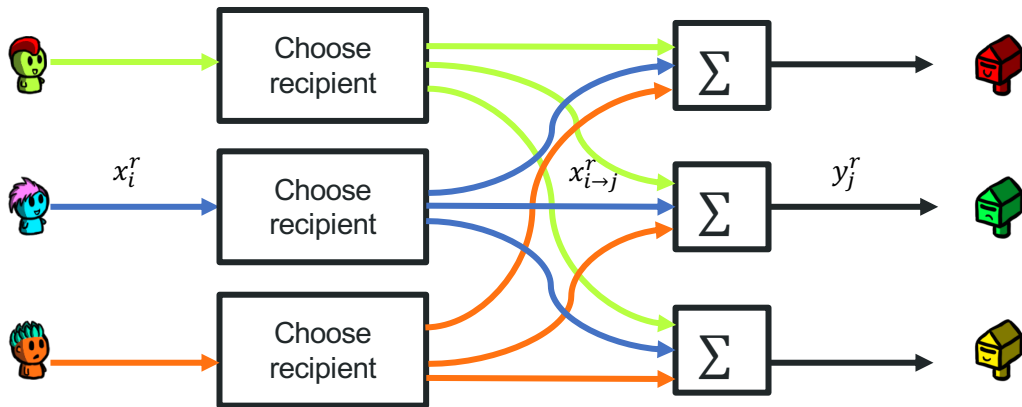
Q: Since y_j^r is the messages received by j in round r (from all the senders!!). What is each distribution?

A: It is a sum of Binomials:

$$y_j^r = \sum_{i=1}^n x_{i \rightarrow j}^r$$

Deriving LSDA

Q: Given these expressions, what is $E\{y_j^r | X, P\}$?



$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$$
 Pr(Blue sends to Green)

$$X = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots & & \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots & & \end{bmatrix}$$

$$x_{i \rightarrow j}^r \sim \text{Bino}(x_i^r, p_{i,j})$$

$$y_j^r = \sum_{i=1}^n x_{i \rightarrow j}^r$$

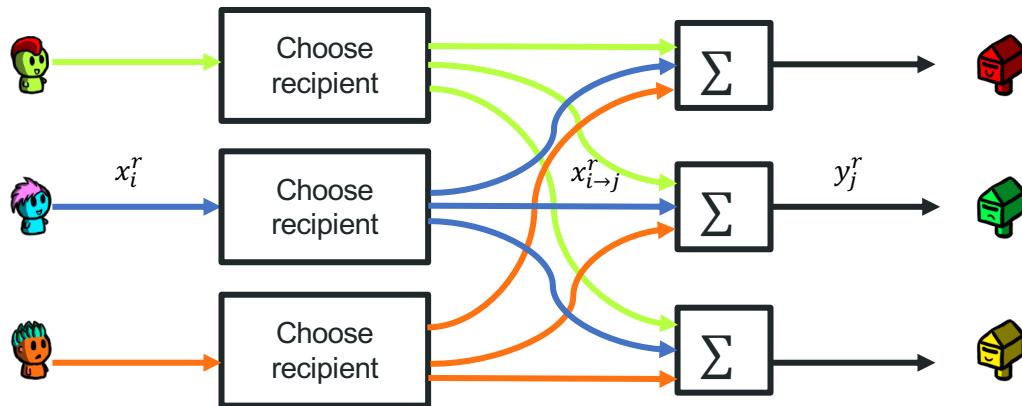
Deriving LSDA

Q: Given these expressions, what is $E\{y_j^r | X, P\}$?

A:

$$E\{y_j^r | X, P\} = \sum_{i=1}^n E\{x_{i \rightarrow j}^r | X, P\} = \sum_{i=1}^n x_i^r \cdot p_{i,j}$$

Note that, for the whole matrix Y , this can just be written as $E\{Y | X, P\} = X \cdot P$



$P =$

			$\Pr(\text{blue person sends to } \text{green cube})$
$p_{1,1}$	$p_{1,2}$	$p_{1,3}$	
$p_{2,1}$	$p_{2,2}$	$p_{2,3}$	
$p_{3,1}$	$p_{3,2}$	$p_{3,3}$	

A red box highlights $p_{1,2}$ and a red arrow points to it from the text above.

$X =$

1	2	1
2	0	2
1	1	2
⋮		

$Y =$

1	1	2
2	2	0
1	1	2
⋮		

$$x_{i \rightarrow j}^r \sim \text{Bino}(x_i^r, p_{i,j})$$

$$y_j^r = \sum_{i=1}^n x_{i \rightarrow j}^r$$

Deriving LSDA

$$\mathbf{P} = \begin{matrix} \begin{matrix} \text{red cube} & \text{green cube} & \text{yellow cube} \end{matrix} \\ \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix} \begin{matrix} \text{blue head} \\ \text{green head} \\ \text{orange head} \end{matrix} \end{matrix} \quad \mathbf{X} = \begin{matrix} \begin{matrix} \text{blue head} & \text{green head} & \text{orange head} \end{matrix} \\ \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & 2 \\ 1 & 1 & 2 \\ \vdots & & \end{bmatrix} \end{matrix} \quad \mathbf{Y} = \begin{matrix} \begin{matrix} \text{red cube} & \text{green cube} & \text{yellow cube} \end{matrix} \\ \begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & 0 \\ 1 & 1 & 2 \\ \vdots & & \end{bmatrix} \end{matrix}$$

- We have $E\{Y|X, P\} = X \cdot P$, and the observation of Y .
- Therefore, LSDA tries to solve:

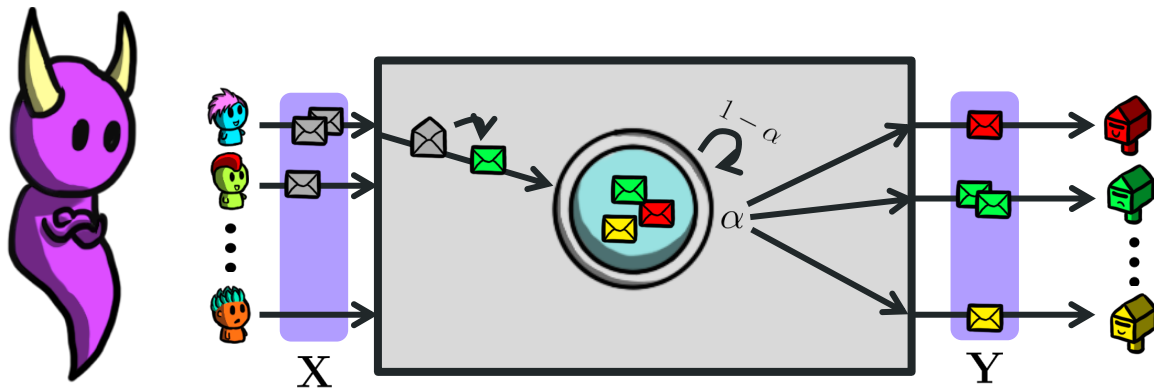
$$\hat{P} = \operatorname{argmin}_{P \in \mathcal{P}} \|Y - X \cdot P\|_2^2$$

- Here, \mathcal{P} is the set of all matrices P whose elements are between 0 and 1 and whose rows add up to 1.
- This is a constrained least-squares optimization problem. If we remove the constraints, we have a closed-form expression for the solution:

$$\hat{P} = (X^T X)^{-1} X^T Y$$

5. LSDA in other mixes

- What you saw is the vanilla (unconstrained) LSDA in a simple mix that does not delay messages between rounds.
- The attack can also be extended to more complicated mixes. Usually, to derive the attack, one just needs to update the expression of $E\{Y|X, P\}$ taking into account the operation of the mix.
- For example, given a mix that delays each message for d rounds with probability $\Pr(D = d)$, the expression of $E\{Y|X, P\}$ becomes:



$E\{Y|X, P\} = D \cdot X \cdot P$
where D is a square matrix whose (r, s) th entry is:

$$\Pr(D = r - s)$$

End of Day 14, to be continued

6. Side-Channel Attacks

6. Side-channel attacks

- So far, we have considered systems where the designer knows exactly what is leaked to the adversary. Let's think about each of them:
 - Census reconstruction
 - SQL inference attacks
 - Dinur-Nissim dataset reconstruction
 - Statistical inference (e.g., in location privacy)
 - De-anonymization in mixes
- Side-channel attacks: the adversary gets some additional leakage not expected by the system designer.
- Recall side-channel attacks from lecture 10.

6. Recall, Side-channel attacks

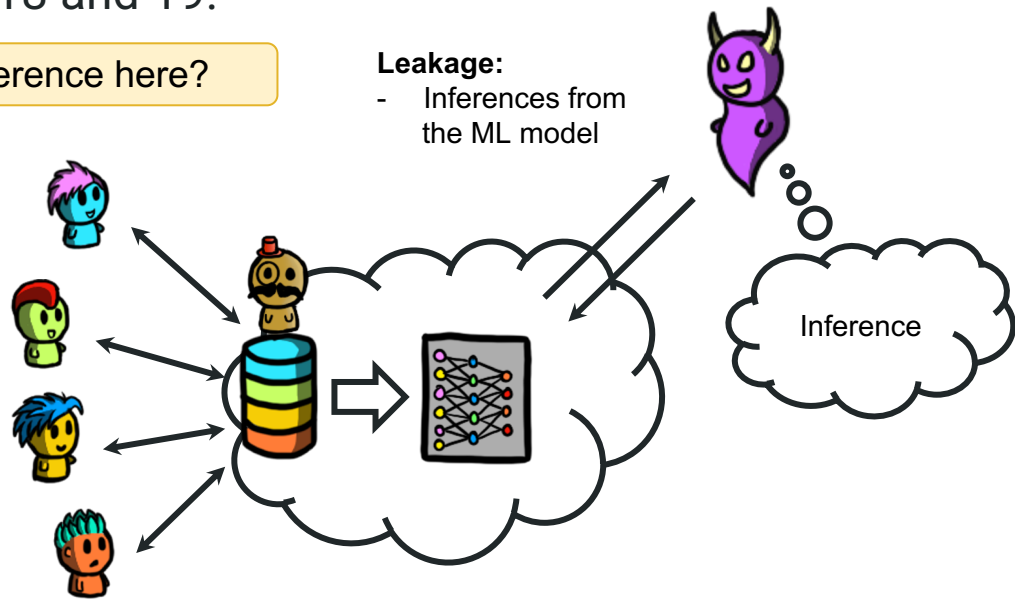
- Timing side-channels
 - The runtime of code depends on sensitive data
- Storage side-channels
 - Storage is not properly cleaned and then shared
 - Cache side-channels, Cold boot attack
- Hardware side-channels
 - Power, electro-magnetic waves, sound
- Fault injection
 - Leakage from error behavior

7. Inference Attacks in Machine Learning

7. Inference attacks in Machine Learning

- There are many possible inference attacks in ML.
- For now, we will just think about the adversary **goals** and possible **techniques**; we'll see more details in lectures 18 and 19.

Q: We saw this before: what could be an inference here?



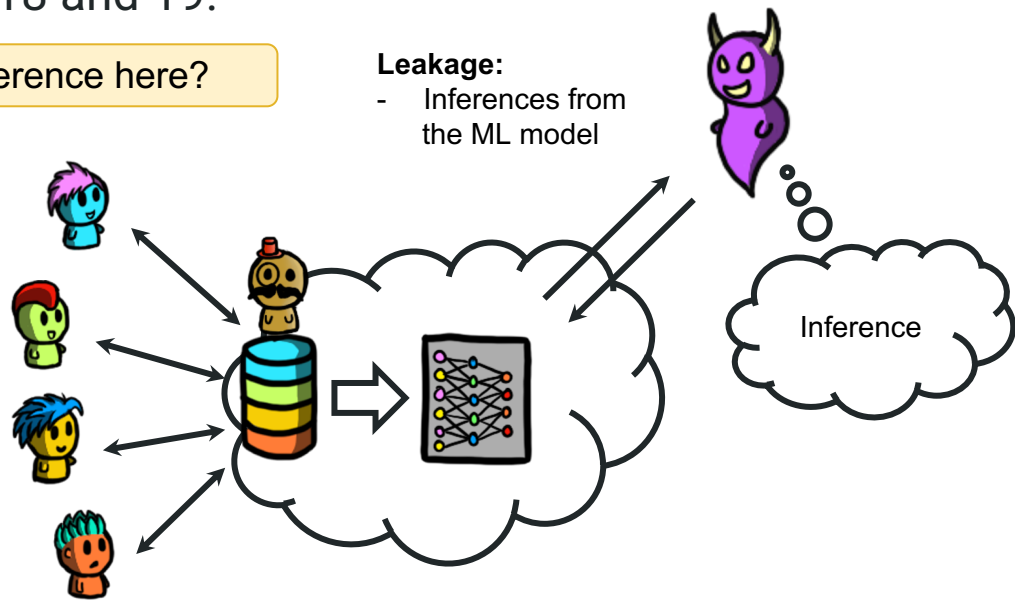
7. Inference attacks in Machine Learning

- There are many possible inference attacks in ML.
- For now, we will just think about the adversary **goals** and possible **techniques**; we'll see more details in lectures 18 and 19.

Q: We saw this before: what could be an inference here?

A:

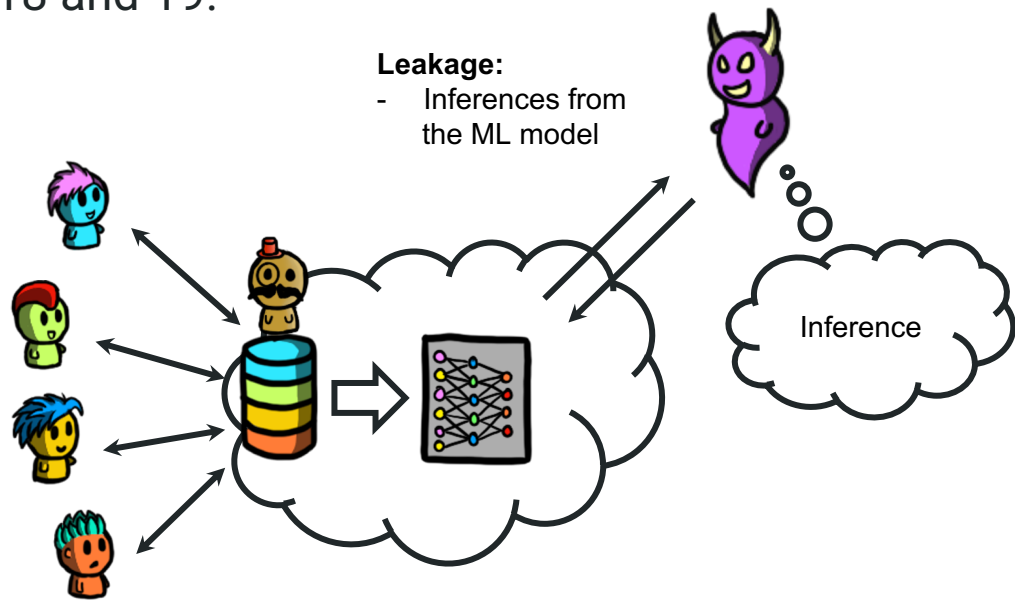
- Membership inference
- Attribute inference (parts of a data sample)
- Property inference (property of the whole training set)
- Reconstruction attack (infer a whole training set)
- ...



7. Inference attacks in Machine Learning

- There are many possible inference attacks in ML.
- For now, we will just think about the adversary **goals** and possible **techniques**; we'll see more details in lectures 18 and 19.

Q: If you were the adversary, which *techniques* would you use to run an attack in this scenario?

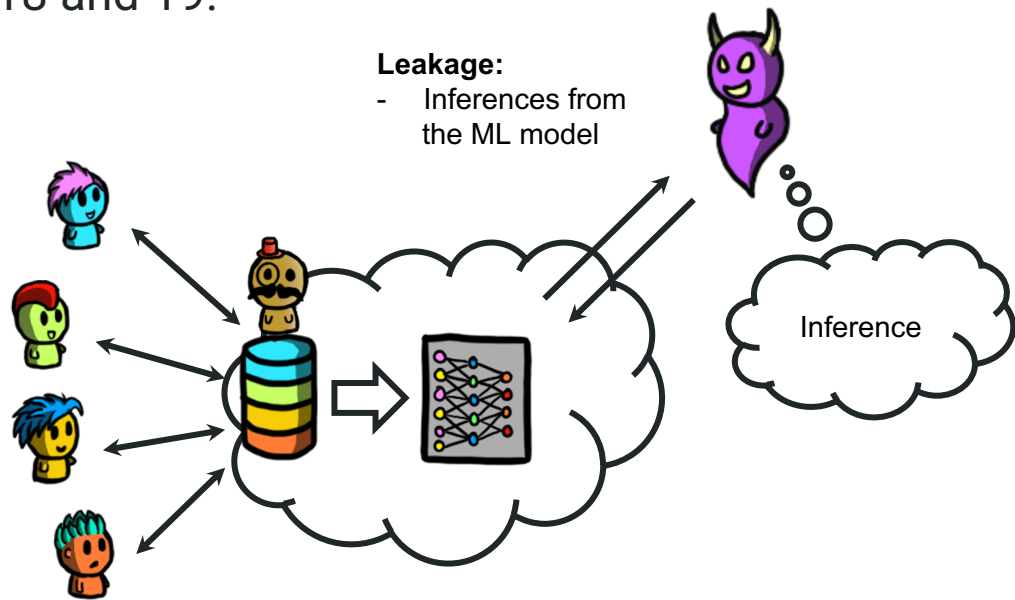


7. Inference attacks in Machine Learning

- There are many possible inference attacks in ML.
- For now, we will just think about the adversary **goals** and possible **techniques**; we'll see more details in lectures 18 and 19.

Q: If you were the adversary, which *techniques* would you use to run an attack in this scenario?

A: the idea is to use the fact that the model is more “confident” on samples it has trained on. We can use the confidence score, we can use thresholding techniques or train an ML model as an attack, etc.

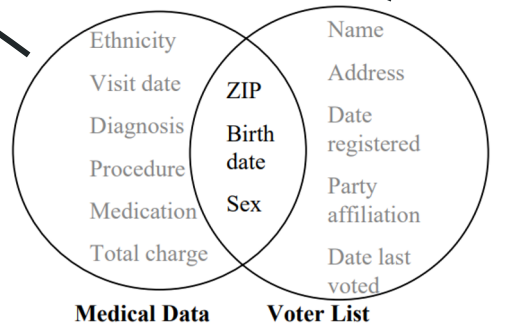


8. Linking Attacks

8. Linking attacks

- As the name suggests, linking attacks find connections between two different sources of leakage that, alone, seem harmless.
- Famous example, from [1]:

The Group Insurance Commission (GIC) in Massachusetts, sold data from 135,000 state employees to industry and researchers. They believed it was anonymous, so it was fine.



For \$20, you can purchase the voter registration list for Cambridge, Massachusetts

Fun fact: 87% (216 million of 248 million) of the population in the United States had reported characteristics that likely made them **unique** based only on {5-digit ZIP, gender, date of birth}

Figure 1 Linking to re-identify data

[1] Sweeney, Latanya. "k-anonymity: A model for protecting privacy." *International journal of uncertainty, fuzziness and knowledge-based systems* 10.05 (2002): 557-570.

Conclusion

We have seen:

1. **Census reconstruction attacks** - Small groups are bad. With enough constraints we can solve for the data.
2. **SQL inference attacks (tracker attacks)** - With enough queries, we can bypass basic defences.
3. **Database reconstruction attacks** — Again, too many queries can reduce noise. Also, noise must be chosen carefully.
4. **Statistical inference attacks** - We can attack randomized mechanisms using knowledge of the distribution.
 - Maximum Likelihood
 - Maximum A-Posteriori
5. **De-anonymization attacks** - Even when the distribution is hard to model, we can launch approximate attacks.
6. **Side-channel attacks** - Assumptions / Threat model matters.
7. **ML Inference attacks** - Inference attacks can scale to much more complex aggregations and mechanisms.
8. **Linking attacks** - As with multiple queries, multiple datasets are also a problem.

Conclusion

- Inference attacks are one way of quantifying the leakage of a mechanism empirically
 - Need to be cautious as:
 - What if a better attack is developed later
 - What if the assumptions of the attacks do not represent real world threats
- Next we look at defenses
 - More theoretical way to measure privacy
 - Usually a lower bound on privacy