An brief tour of Differential Privacy

Your guide: Avrim Blum

A preliminary story

• A classic cool result from theoretical crypto:
  - Say you want to figure out the average grade on a test of people in the room, without revealing anything about your own grade other than what is inherent in the answer.
  - Turns out you can actually do this. In fact, any function at all. "secure multiparty computation".
  - It's really cool. Want to try?
  - Anyone have to go to the bathroom?
  - What happens if we do it again?
  - Or what about someone who came in late?

Differential Privacy [Dwork et al.]

• "Lets you go to the bathroom in peace"
  - What we want is a protocol that has a probability distribution over outputs such that if person \( i \) changed their input from \( x_i \) to any other allowed \( x_i' \), the relative probabilities of any output do not change by much.
  - So, for instance, can pretend your input was any other allowed value you want.
  - Can view as model of "plausible deniability".
  - Even if no bad intent, who knows what prior info people have?

Differential Privacy: Definition

It's a property of a protocol \( A \) which you run on some dataset \( X \) producing some output \( A(X) \).

• \( A \) is \( \epsilon \)-differentially private if for any two neighbor datasets \( X, X' \) (differ in just one element \( x_i \to x_i' \)),

\[
\Pr(A(X) = v) \leq e^\epsilon \Pr(A(X') = v)
\]

for all outcomes \( v \),

\[
e^{-\epsilon} \leq \Pr(A(X) = v)/\Pr(A(X') = v) \leq e^\epsilon\]

≈ 1-\( \epsilon \) probability over randomness in \( A \)

≈ 1+\( \epsilon \) probability over randomness in \( A \)

View as model of plausible deniability

(pretend after the fact that my input was really \( x_i' \))

for all outcomes \( v \),

\[
e^{-\epsilon} \leq \Pr(A(X) = v)/\Pr(A(X') = v) \leq e^\epsilon\]

≈ 1-\( \epsilon \) probability over randomness in \( A \)

≈ 1+\( \epsilon \) probability over randomness in \( A \)
Differential Privacy: Definition

It's a property of a protocol \( A \) which you run on some dataset \( X \) producing some output \( A(X) \).

- \( A \) is \( \epsilon \)-differentially private if for any two neighbor datasets \( X, X' \) (differ in just one element \( x_i \rightarrow x'_i \)),

\[
\frac{\Pr(A(X)=v)}{\Pr(A(X')=v)} \leq e^\epsilon
\]

for all outcomes \( v \),

\[
\approx 1-\epsilon \quad \text{probability over randomness in } A
\]

OK, great. How can we achieve it? What kind of \( \epsilon \) can we get with reasonable utility?

Silly algorithm: \( A(X)=0 \) no matter what. Or \( A(X)=\text{unif}[0,b] \)

Differential Privacy via output perturbation

Say have \( n \) inputs in range \([0,b]\). Want to release average while preserving privacy.

- Natural idea: take output and perturb with noise.
- Better: Laplace (or geometric) distrib \( p(x) \propto e^{-|x|/\lambda} \)

\[
e^{(x-b/n)/\lambda} / e^{-x/\lambda} = e^{b/n}\lambda
\]

Set \( \lambda = b/(n\epsilon) \)

"Laplace mechanism"

So, add noise roughly \( 1/\epsilon \) \times (effect any individual can have on outcome) gives desired ratio \( e^\epsilon = (1+\epsilon) \).

If want answer within \( \pm \alpha b \), need \( n \geq 1/(\epsilon\alpha) \).

Utility/privacy/database-size tradeoff
Laplace mechanism more generally

\[ f \xrightarrow{} f(X) + \text{noise} \]

1. E.g., \( f = \) standard deviation of income
2. E.g., \( f = \) result of some fancy computation.

**Global Sensitivity of** \( f \):

\[ GS_f = \max_{\text{neighbors } X, X'} |f(X) - f(X')| \]

- Just add noise \( \text{Lap}(GS_f/\epsilon) \).

What can we do with this?

\[ f \xrightarrow{} f(X) + \text{noise} \]

1. Interface to ask questions
2. Run learning algorithms by breaking down interaction into series of queries.
3. *But*, each answer leaks some privacy:
   - If \( k \) questions and want total privacy loss of \( \epsilon \), we'd better answer each with \( \epsilon/k \).
   - Need to use improved mechanism to do better.

Remainder of presentation

- Local sensitivity / Smooth sensitivity [Nissim-Raskhodnikova-Smith '07]
- Objective perturbation [Chaudhuri-Monteleoni-Sarwate '08]
- Sample and Aggregate [NRS '07]
- Exponential Mechanism [McSherry-Talwar '07]
- What can you say about publishing a sanitized database? [B-Ligett-Roth '08]

Local Sensitivity

\[ f \xrightarrow{} f(X) + \text{noise} \]

1. Consider \( f = \) median income
   - If \( f \) is not very sensitive on the actual input \( X \), does that mean we don't need to add much noise?
   - Be careful: what if sensitivity itself is sensitive?

Smooth Sensitivity

\[ f \xrightarrow{} f(X) + \text{noise} \]

- [NRS07] prove can instead use (roughly) the following smooth bound instead:
  \[ \text{Max}_Y [ LS_f(Y) e^{-\epsilon d(X,Y)} ] \]
  - E.g., what does this say in the case of the median?
Smooth Sensitivity

- In principle, could apply sensitivity idea to any learning algorithm (say) that you’d like to run on your data.
- But might be hard to figure out what it is.

Sample-and-aggregate (also [NRS07])

- Say you have some learning algorithm and hard to tell how sensitive it would be to changing a single input.
- Some way to run it privately anyway?

Sample-and-aggregate (also [NRS07])

- Get outputs
- Then combine these outputs.
- Changing an input can only change one of outputs.
- So, just have to use privacy-preserving combination procedure.

Objective perturbation [CMS08]

- Idea: add noise to the objective function used by the learning algorithm.
- Natural for algorithms like SVMs that have regularization term.
- [CMS] show how to do this, if use a smooth loss function.
- Also show nice experimental results.

Exponential Mechanism [MT07]

- What about running some generic optimization algorithm? Want to find some that optimizes some.
- Idea: score each possible output based on how close to optimum.
- Run Laplace over scores: i.e., produce random output with prob exponential in -score.
- Get privacy based on GS(score). May not be efficient. Will see interesting use in a sec...

What about outputting sanitized databases?

- So far, just question-answering. Each answer leaks some privacy - at some point, have to shut down.
- What about outputting a sanitized database that people could then examine as they wish?
- And is related to the original database...
What about outputting sanitized databases?

- Could ask a few questions (using previous mechs) and then engineer a database that roughly agrees on these answers.
- But really, we want a database that matches on questions we haven’t asked yet.
- Do you need to leak privacy in proportion to number of questions asked?

Actually, no you don’t…

- Fix a class \( C \) of quantities to preserve. E.g., fraction of entries with \( x[i_1]=1, x[i_2]=0 \ldots x[i_k]=1 \).
- Want \( \epsilon \)-privacy and preserve all \( q \in C \) up to \( \pm \alpha \).
- E.g., in this case, we want to preserve all 3\( d \) conjunctive queries.

Fix a class \( C \) of quantities to preserve. E.g., fraction of entries with \( x[i_1]=1, x[i_2]=0 \ldots x[i_k]=1 \).
- Want \( \epsilon \)-privacy and preserve all \( q \in C \) up to \( \pm \alpha \).

[BLR] show: in principle, can do with database of size only \( n = O(d \log |C|) \)

- Put explicit distribution on them, using exponential mechanism of [McSherry-Talwar]
- Solve to get \( n \approx \frac{VCdim(C) \cdot d}{\epsilon \alpha^3} \)

Actually, no you don’t…

- Fix a class \( C \) of quantities to preserve. E.g., fraction of entries with \( x[i_1]=1, x[i_2]=0 \ldots x[i_k]=1 \).
- Want \( \epsilon \)-privacy and preserve all \( q \in C \) up to \( \pm \alpha \).

[BLR] show: in principle, can do with database of size only \( n = O(d \log |C|) \)

- Put explicit distribution on them, using exponential mechanism of [McSherry-Talwar]
- Solve to get \( n \approx \frac{VCdim(C) \cdot d}{\epsilon \alpha^3} \)

Idea:
- Learning theory bounds say that there exist small databases that apx preserve all quantities in \( C \). In particular, \( m = O(VCdim(C)/\alpha^3) \) is sufficient.
- Put explicit distribution on them, using exponential mechanism of [McSherry-Talwar]
- For what \( n \) does this whp output \( S \) of low penalty?

Alg very inefficient since putting explicit distrib on all small databases.
- Improvements due to [RR10] [HR10]. Time poly in \( 2^d \) (size of universe) and online.
- Still, seems very hard to get fully efficient algorithm.
- Note: even \( 2^{d/2} \) would be interesting…
Differential Privacy summary & discussion

Positives:
- Clear semantic definition. Any event (anything an adversary might do to you) has nearly same prob if you join or don't join, lie or tell the truth.
- Nice composability properties.
- Variety of mechanisms developed for question answering in this framework.
- *Some* work on sanitized database release.

Negatives / open issues
- It's a pessimistic/paranoid quantity, so may be more restrictive than needed.
- "\( \varepsilon \)" is not zero. Privacy losses add up with most mechanisms (but see, e.g., [RR10],[HR10])
- Doesn't address group information.
- Notion of "neighboring database" might need to be different in network settings.
- ...