Tight and Flexible Accounting of Differential Privacy

Yu-Xiang Wang

COMPUTER SCIENCE
UC SANTA BARBARA
Computing. ReInvented.

Privacy challenges in the AI Era

Personal health records

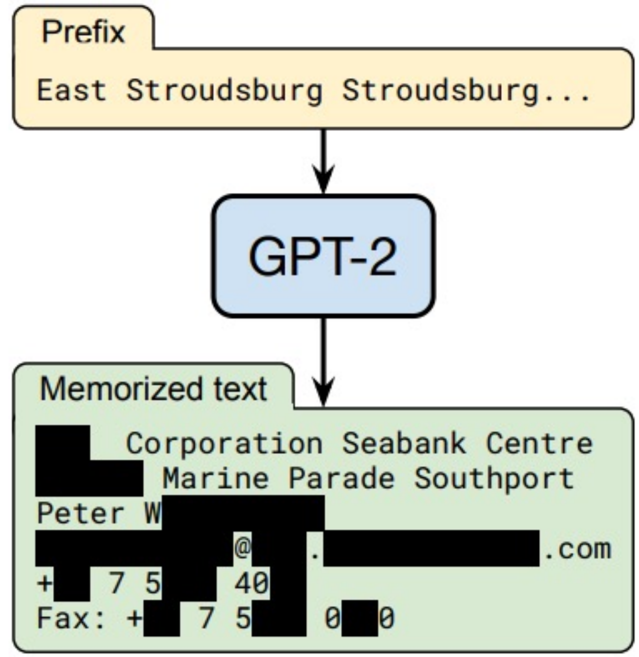


Figure 1: **Our extraction attack.** Given query access to a neural network language model, we extract an individual person's name, email address, phone number, fax number, and physical address. The example in this figure shows information that is all accurate so we redact it to protect privacy.

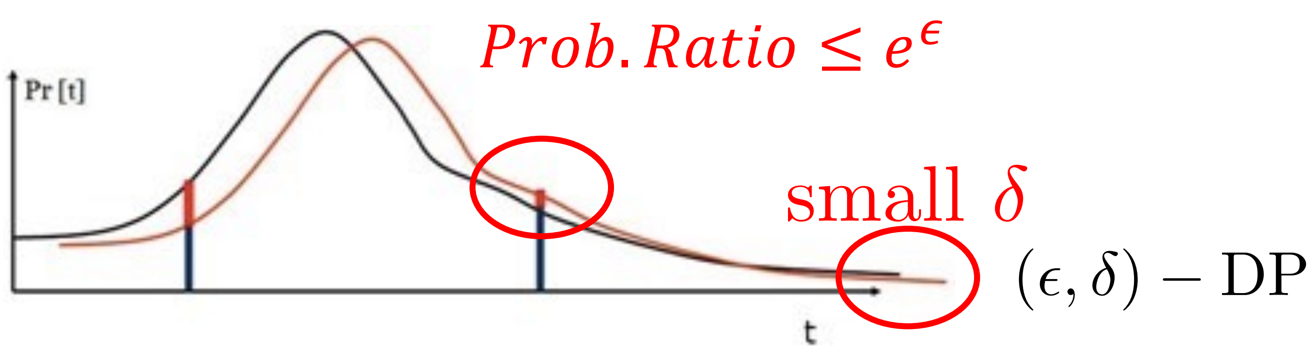
(Carlini et al., 2020)

Differential Privacy provably addresses these challenges.

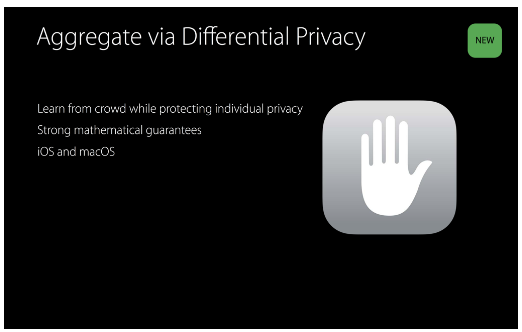
- GDPR / CCPA, Risk of identifying users, extracting their data
- **Differential privacy (DMNS 2006)** is a formal definition of privacy with many good properties.

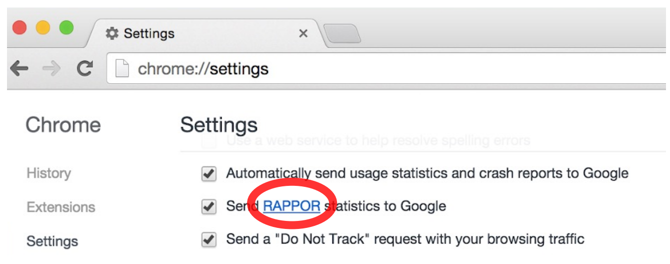
  *legal compliance of DP is still being debated

  - The two worlds *with or without* “Alice” are indistinguishable.



Differential privacy is transforming into a practical technology!





Key challenges from the 2018 "DP-Deployed" Meeting...

- **Utility loss:** Utility remains the primary issue for small to medium-sized data, or high capacity models.
- **Privacy accounting:** There is no standard in selecting, reporting, interpreting privacy parameter ϵ. It is hard to precisely quantify the actual privacy loss due to the slacks in the mathematical analysis.
- **Scalability issue:** The design and analysis of DP mechanisms is delicate and error-prone even for experts — there will never be enough PhDs with DP training to meet the growing demand.
- **Implementation:** There are few high-quality codes for DP, with the exception of PINQ (McSherry, 2009), Ektelo(Zhang et al., 2018) and tf.privacy (Google et al., 2018), each serving a particular niche.

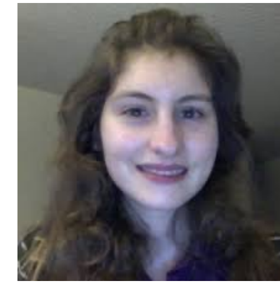
The meeting calls for:

- DP algorithms that are not just "rate-optimal", but also simple / practical.
- DP tools with **exactly optimal privacy accounting** that allows flexible design of complex algorithms using basic building blocks.

"Constant matters in differential privacy!"

My group's research enables more practical DP

Yuqing Zhu



Rachel Redberg

Challenges in Practicing DP

- **Lack of utility on small/medium data**
- **Mathematical overhead of advanced DP mechanisms**
- **Shortage of experts vs. Growing demand**

***Main research goal*: Less privacy loss, more utility!**

RT1: Exact Optimal DP Accounting (Tighten up the worst case)

RT2: Per-Instance & data-adaptive DP (More utility via "nice" input data)

RT3: DP with Auxiliary Public Info (Even more utility via hybrid models)

autodp: the "autograd" for differential privacy
Enable state-of-the-art DP computation for non-experts

This talk is primarily about the following paper:
Zhu, Dong and W. (2021) Optimal Accounting of Differential Privacy via Characteristic Function.

Outline of the talk

- Mechanism-specific privacy accounting
  - Application to Differentially Private Deep Learning
- Limitation of RDP and existing theory of PLD
- Main results:
  - Dominating pairs
  - Composition and amplification by sampling
  - Characteristic function representation
- Autodp -- a flexible tool for privacy accounting

Composition theorem of DP

Composed mechanism **Individual mechanisms**

$$(M_1, M_2, \ldots, M_k)(x) = (M_1(x), M_2(x), \ldots, M_k(x))$$

- Classical Composition Theorem
  - Individual mechanisms satisfy DP with parameters
$$(\epsilon_1, \delta_1), (\epsilon_2, \delta_2), \ldots, (\epsilon_k, \delta_k)$$
  - Then composed mechanisms satisfy $(\epsilon_g, k\delta + \delta')$-DP with $\epsilon_g = \sqrt{2k \ln(1/\delta')} \cdot \epsilon + k \cdot \epsilon \cdot (e^{\epsilon} - 1)$

Why is this not good enough?

Mechanism-specific analysis of DP mechanisms and their composition

Composed mechanism **Individual mechanisms**

$$(M_1, M_2, \ldots, M_k)(x) = (M_1(x), M_2(x), \ldots, M_k(x))$$

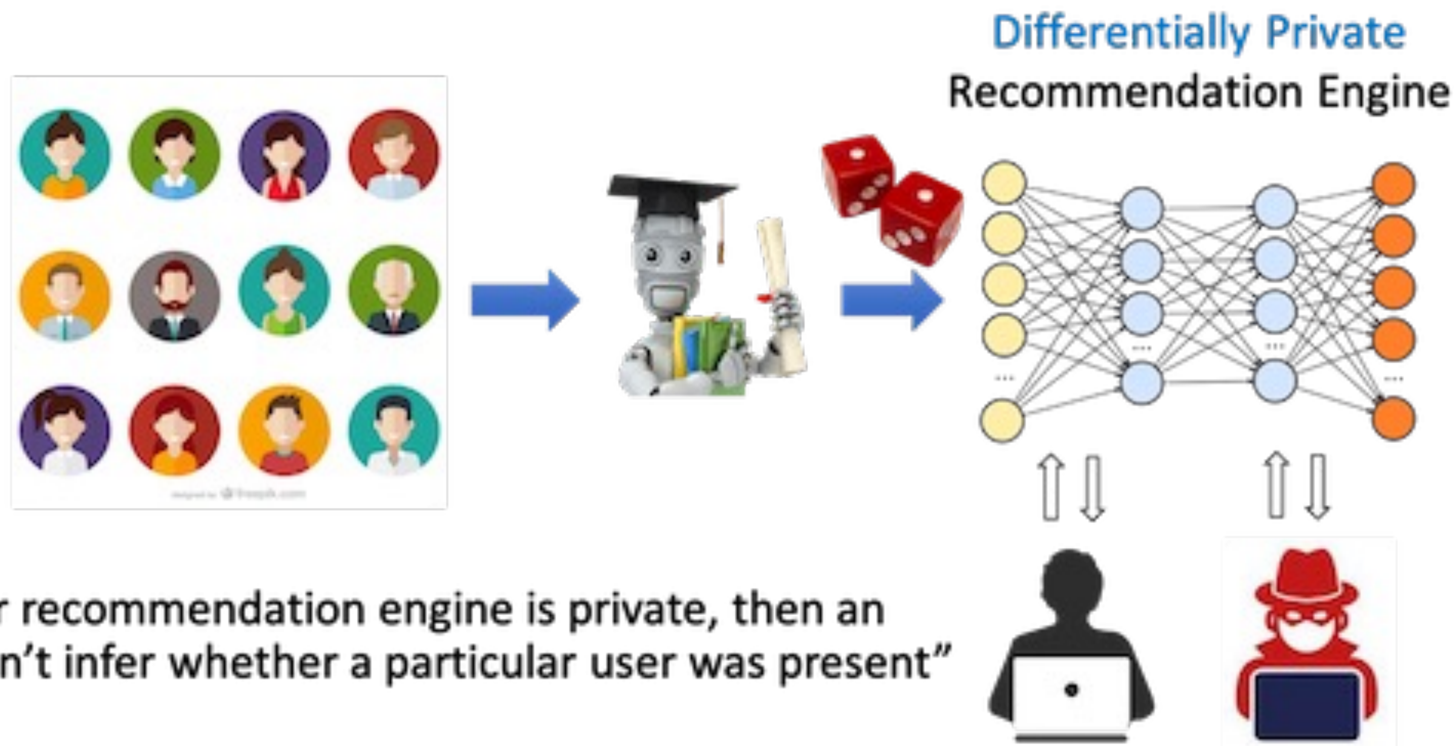
- Instead of composing DP guarantees, why not **composing specific mechanisms**?
  - We can describe each mechanism by a function.

	Functional view	Pros	Cons
Renyi DP [Mironov, 2017]	$D_\alpha(P\|Q) \leq \epsilon(\alpha), \forall \alpha \geq 1$	Natural composition	lossy conversion to (ϵ, δ)-DP.
Privacy profile [Balle and Wang, 2018]	$\mathbb{E}_q[(\frac{p}{q} - e^\epsilon)_+] \leq \delta(e^\epsilon), \forall \epsilon \geq 0$	Interpretable.	messy composition.
f-DP[Dong et al., 2021]	Trade-off function f	Interpretable, CLT	messy composition.
PLD [Sommer et al., 2019, Koskela et al., 2020]	Probability density of $\log(p/q)$	Natural composition via FFT	Limited applicability.

Table 1: Modern functional views of DP guarantees and their pros and cons.

- This is the key idea underlying modern DP accounting.

Example: Differentially Private Machine Learning



"If your recommendation engine is private, then an adversary can't infer whether a particular user was present"

Example: Deep Learning with Differential Privacy and NoisySGD

$$\theta_{t+1} \leftarrow \theta_t - \eta_t \left(\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \nabla f_i(\theta_t) + Z_t \right)$$

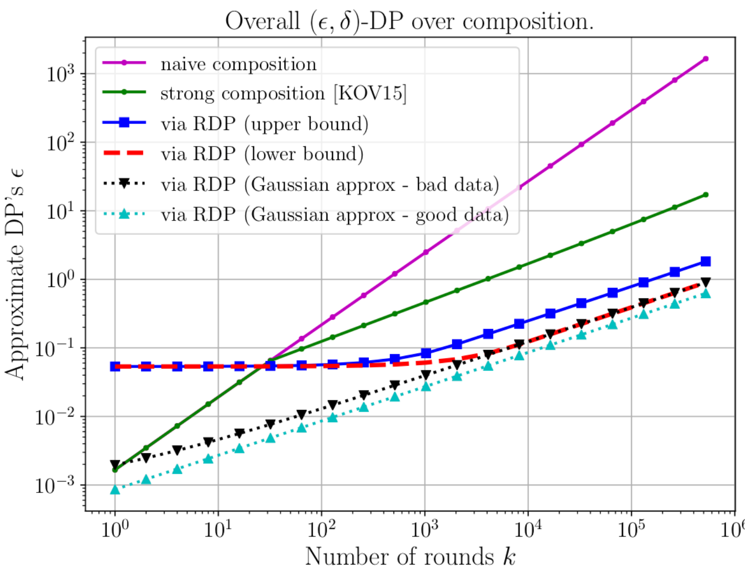
Given a sequence of DP-mechanisms, what is the privacy loss over composition?

- **Classical (advanced) composition:** Composing (ε, δ)-DP k times, return results from the optimal advanced composition.

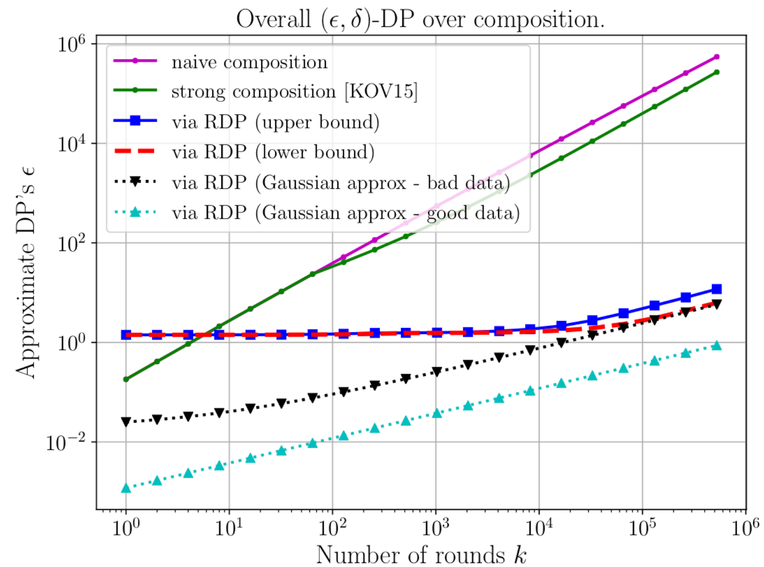
- **Moments accountant:** Compose "Subsampled-Gaussian" mechanism k times, compute (ε, δ) in the end.

Deep learning with differential privacy
M Abadi, A Chu, I Goodfellow, HB McMahan… - Proceedings of the …, 2016 - dl.acm.org
Machine learning techniques based on neural networks are achieving remarkable results in a wide variety of domains. Often, the training of models requires large, representative …
Cited by 2293 Related articles Import into BibTeX

The practical gains from moments accountant are significant



(a) Subsampled Gaussian with σ = 5



(a) Subsampled Gaussian with σ = 0.5

Figures from W., Balle, Kasiviswanathan (2018) “Subsampled Rényi Differential Privacy and Analytical Moments Accountant”

The dream of a general-purposed DP accounting tool

- Flexible mix & match of the DP building blocks
- Constant-tight mechanism-specific composition

Outline of the talk

- Mechanism-specific privacy accounting
  - Application to Differentially Private Deep Learning

- Limitation of RDP and existing theory of PLD

- Main results:
  - Dominating pairs
  - Composition and amplification by sampling
  - Characteristic function representation

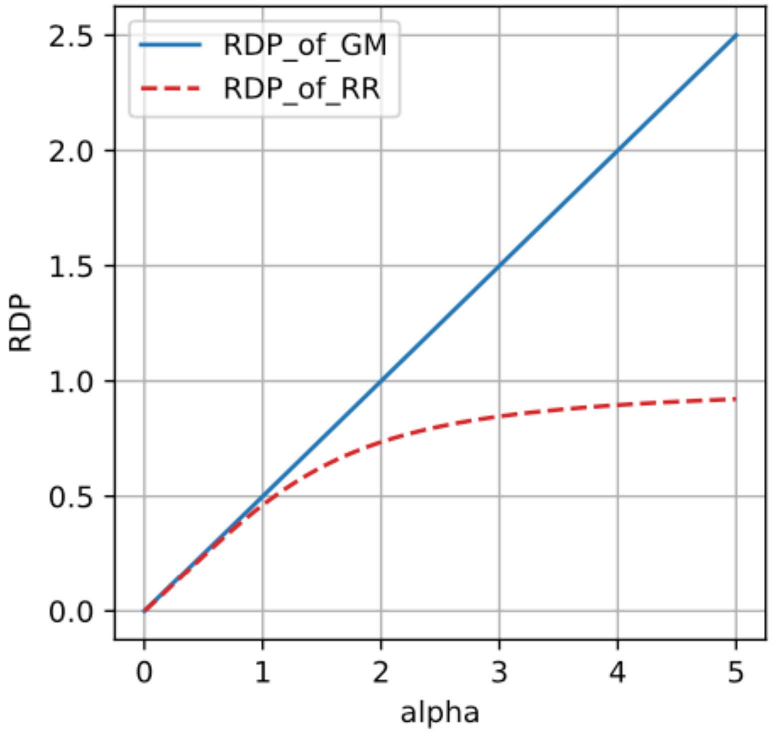
- Autodp -- a flexible tool for privacy accounting

Why aren’t we happy with RDP / moments accountant?

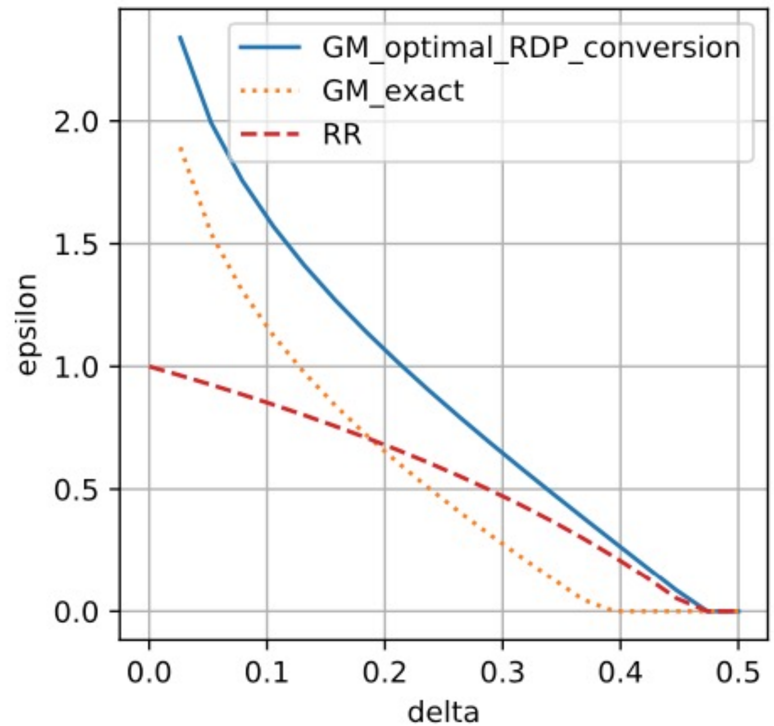
- Limitations of RDP
  1. Some mechanisms do not satisfy RDP
     - e.g. PTR, posterior sampling (even for linear regression).

  2. RDP is a lossy representation of a mechanism

The Conversion from RDP to (ε,δ)-DP is lossy



(a) RDP of RR and GM



(c) (ϵ, δ)-DP of RR and GM

GM with $\sigma = 1$ vs Rand. Resp. with $p = \frac{e}{1+e}$

The promising idea of Privacy Loss Distribution (or PLD) (Sommer et al.; Koskela et al)

- From classical DP theory, the privacy loss RV plays a central role.
  - $L_{P,Q} := \log \frac{p(o)}{q(o)}$, where $o \sim P$.
  - $L_{Q,P} := \log \frac{q(o)}{p(o)}$, where $o \sim Q$.

  where P=M(D), Q=M(D')

- If we keep track of the PLD, then it is tight!

Trouble with the PLD formalism

Challenge: To use PLD, the original authors "require the privacy analyst interested in applying our results (PLD formalism) to provide worst-case distributions."[Sommer et al., 2019]

- **Trouble 1**: The PLD formalism is defined for each pair of the neighboring datasets.
  - How to find the worst-case datasets?
  - Do they even necessarily exist?

- **Trouble 2:** Unclear what PLD to use when the mechanism of interest is "amplified" or "composed".
  - if we know the worst-case distribution for each mechanism, the composition of the individual PLDs may not correspond to the worst-case PLD of the composed mechanism.

Outline of the talk

- Mechanism-specific privacy accounting
  - Application to Differentially Private Deep Learning
- Limitation of RDP and existing theory of PLD
- Main results:
  - Dominating pairs
  - Composition and amplification by sampling
  - Characteristic function representation
- Autodp -- a flexible tool for privacy accounting

Recall: Equivalent definitions of DP via Hockey-Stick Divergences

- Recall: hockey-stick divergence (or privacy profile) is defined as

$$H_\alpha(P\|Q) = \int [p - \alpha q]_+, \forall \alpha > 0$$

- Known: M is (ε, δ)-DP iff

$$\sup_{D \sim D'} H_{e^\varepsilon}(M(D)\|M(D')) \leqslant \delta$$

There might not be a worst-case pair of datasets!

~~Worst case datasets~~ Dominating pairs and tight dominating pairs

Definition 7 (Dominating pair of distributions)**.** *We say that* (P, Q) *is a* dominating *pair of distributions for* $\mathcal{M}$ *(under neighboring relation* $\simeq$*) if for all* $\alpha \geq 0$[2]

$$\sup_{D \simeq D'} H_\alpha(\mathcal{M}(D) \| \mathcal{M}(D')) \leq H_\alpha(P \| Q). \quad (1)$$

- ***Tight*** **dominating pair** if "=" for all α

Proposition: A *tight* dominating pair *exists* for any mechanism.

Questions: How do we find them for each M? How do they work under composition / amplification?

Constructing a Dominating Pair from a Privacy Profile upper bound

1. Project H to the feasible space of privacy profiles

$$\mathcal{H} := \left\{ H : \mathbb{R}_{\geqslant 0} \to \mathbb{R} \middle| \begin{array}{l} H \ \ is\ \ convex,\ \ decreasing, \\ H(0) = 1 \ \ and\ \ H(x) \geqslant (1-x)_+ \end{array} \right\}.$$

2. Take the Fenchel conjugate of H

$$P \ \ has \ \ CDF \ 1 + H^*(x-1) \ \ in \ \ [0,1)$$

$$Q = \text{Uniform}([0,1])$$

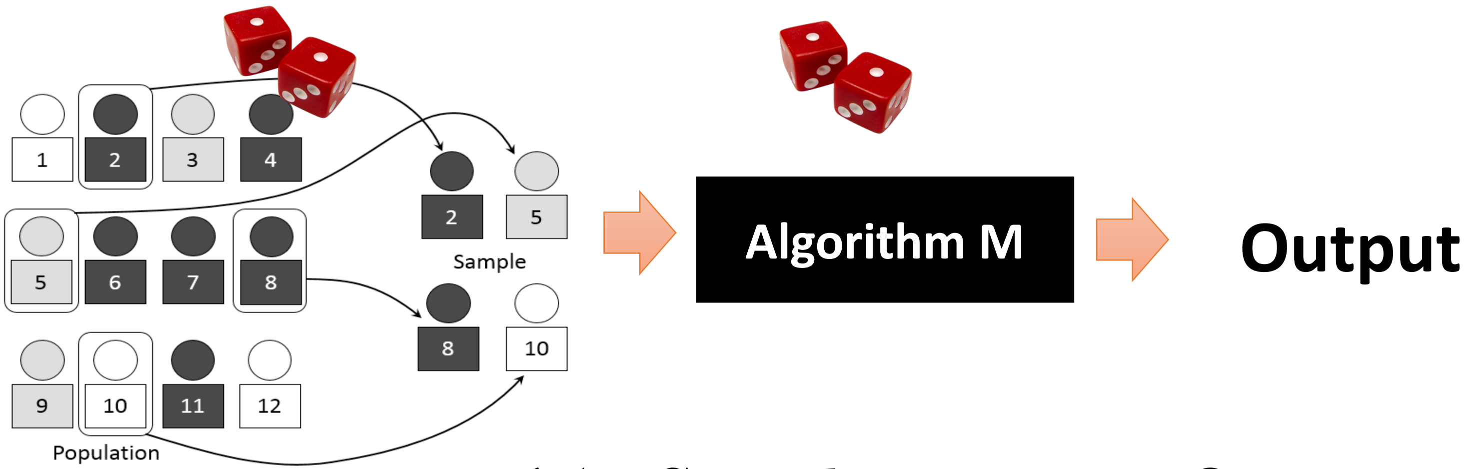
- The projection may improve the upper bound actually!
- No matter the output space of M, P,Q are univariate on [0,1].

Dominating pairs compose adaptively

Theorem (Adaptive Composition):
If (P, Q) *dominates* $\mathcal{M}$ *and* (P', Q') *dominates* $\mathcal{M}'$[3], *then* $(P \times P', Q \times Q')$ *dominates the composed mechanism* $(\mathcal{M}, \mathcal{M}')$.

(*M can depend on the output of M.)

Amplification by Sampling



$$\mathcal{M} \circ \text{Sample} : \text{Data} \to \text{Output}$$

A sensible conjecture:

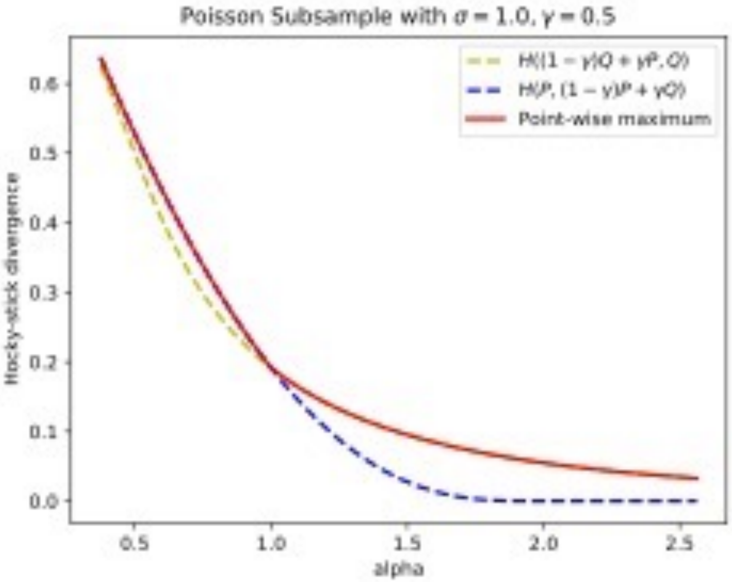
(P,Q)-dominates M => $((1-\gamma)Q + \gamma P, Q)$ dominates $M \circ \text{Sample}_\gamma$

Many published results / empirical work using PLD are implicitly relying on this conjecture.

Amplification by Sampling

A sensible conjecture: **False**

(P,Q)-dominates M => $((1-\gamma)Q + \gamma P, Q)$ dominates $M \circ \text{Sample}_\gamma$



False not just for Gaussian, but for any other mechanisms too, under

- Poisson-sampling + Add/Remove
- Random subset sampling + Replace One

If (P, Q) is a dominating pair of $\mathcal{M}$ under "Add/remove" Relation, then

$$\delta_{\mathcal{M}\circ S_{\textbf{Poisson}}}(\alpha) \leq \begin{cases} H_\alpha((1-\gamma)Q + \gamma P, Q) & \text{for } \alpha \geq 1; \\ H_\alpha(P, (1-\gamma)P + \gamma Q) & \text{for } 0 < \alpha < 1. \end{cases}$$

Our solution: Handling Add-Neighbor and Remove-Neighbor Separately!

Theorem 11. *Let $\mathcal{M}$ be a randomized algorithm.*

(1) *If (P, Q) dominates $\mathcal{M}$ for add neighbors then $(P, (1-\gamma)P + \gamma Q)$ dominates $\mathcal{M} \circ S_{\mathbf{Poisson}}$ for add neighbors and $((1-\gamma)Q + \gamma P, Q)$ dominates $\mathcal{M} \circ S_{\boldsymbol{Poisson}}$ for removal neighbors.*

(2) *If (P, Q) dominates $\mathcal{M}$ for replacing neighbors, then $(P, (1-\gamma)P + \gamma Q)$ dominates $\mathcal{M} \circ S_{\mathbf{Subset}}$ for add neighbors and $((1-\gamma)P + \gamma Q, P)$ dominates $\mathcal{M} \circ S_{\mathbf{Subset}}$ for removal neighbors.*

- For k-fold composition of the sampled algorithm, just do

$$\max\{H_{e^\epsilon}(P_1^k||Q_1^k), H_{e^\epsilon}(P_2^k||Q_2^k))\}$$

Checkpoint: Broader applicability of PLD

- Which pair of distributions to use for PLD to obtain valid DP bounds?
  - Our answers: Dominating pairs!
- How to find dominating pairs?
  - Case by case. But one can convert from existing analysis
- Composition and Subsampling
  - Useful for constructing complex mechanisms from basic building blocks

Outline of the talk

- Mechanism-specific privacy accounting
  - Application to Differentially Private Deep Learning
- Limitation of RDP and existing theory of PLD
- Main results:
  - Dominating pairs
  - Composition and amplification by sampling
  - Characteristic function representation
- Autodp -- a flexible tool for privacy accounting

How to represent PLD of a dominating pair and compose efficiently?

- Existing approach: Fourier Accountant

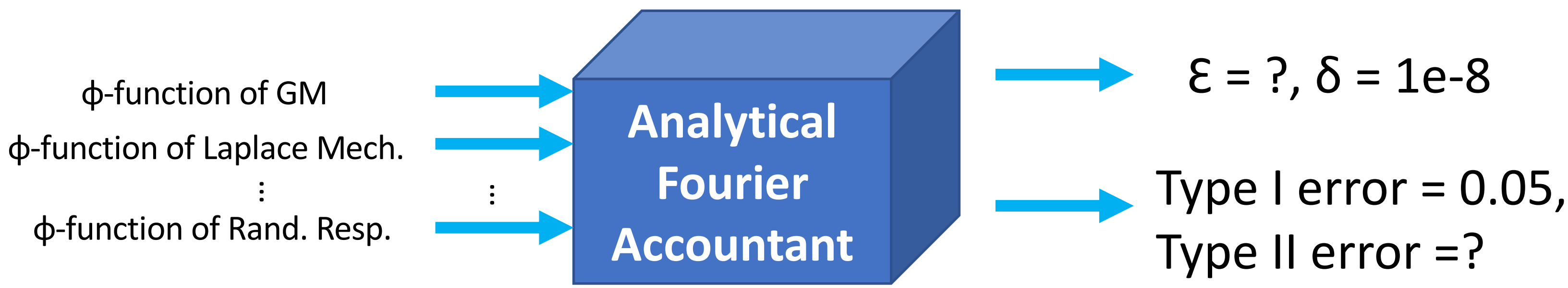
1. Truncate and discretize the density of PLRV
2. FFT to convert it to a Fourier domain representation
3. Compose in the Fourier domain. (Pointwise multiplication)
4. Inverse FFT back to the original space after composition

(Sommer et al. 2019; Koskela et al, 2020; 2021; Gopi et al, 2020)

Analytical Fourier accountant

Represent two characteristic functions of the dominating PLRV

$$\phi_{\mathcal{M}}(\alpha) := \mathbb{E}_P[e^{i\alpha \log(p/q)}],\ \phi'_{\mathcal{M}}(\alpha) := \mathbb{E}_Q[e^{i\alpha \log(q/p)}]$$



- **Natural Composition like RDP:** simply add up the (complex) log of ϕ-functions
- **Tight (Ɛ, δ)-DP Conversion**: via Levy's formula
- **Interpretable tradeoff function:** via duality.

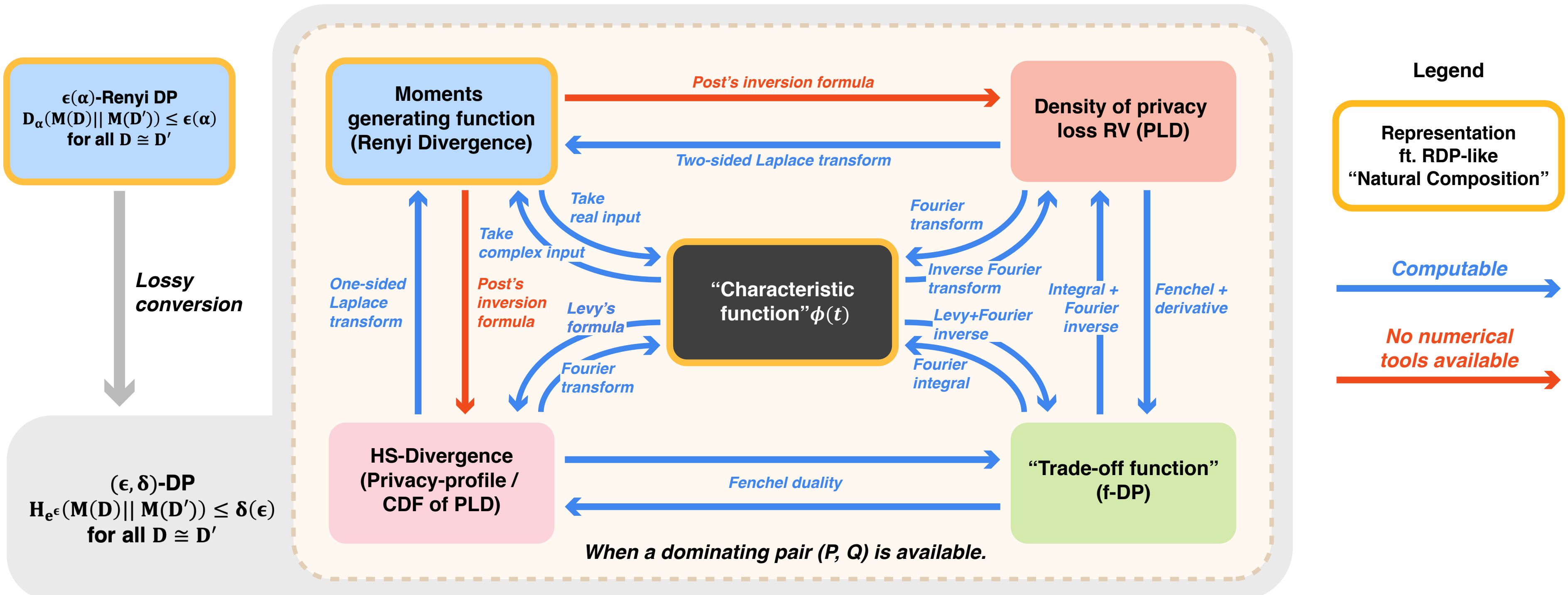
Examples of ϕ-function for common mechanisms

Mechanism	Dominating Pair	ϕ function
Randomized Response	$P : \Pr_P[0] = p; Q : \Pr_Q[1] = p$	$\phi_{\mathcal{M}}(\alpha) = \phi'_{\mathcal{M}}(\alpha) = pe^{\alpha i \log(\frac{p}{1-p})} + (1-p)e^{\alpha i \log(\frac{1-p}{p})}$
Laplace Mechanism	$P : p(x) = \frac{1}{2\lambda}e^{-\lvert x\rvert/\lambda}; Q : q(x) = \frac{1}{2\lambda}e^{-\lvert x-1\rvert/\lambda}$	$\phi_{\mathcal{M}}(\alpha) = \phi'_{\mathcal{M}}(\alpha) = \frac{1}{2}\left(e^{\frac{\alpha i}{\lambda}} + e^{\frac{-\alpha i-1}{\lambda}} + \frac{1}{2\alpha i+1}(e^{\frac{\alpha i}{\lambda}} - e^{\frac{-\alpha i-1}{\lambda}})\right)$
Gaussian Mechanism	$P : \mathcal{N}(1, \sigma^2); Q : \mathcal{N}(0, \sigma^2)$	$\phi_{\mathcal{M}}(\alpha) = \phi'_{\mathcal{M}}(\alpha) = e^{\frac{-1}{2\sigma^2}(\alpha^2 - i\alpha)}$

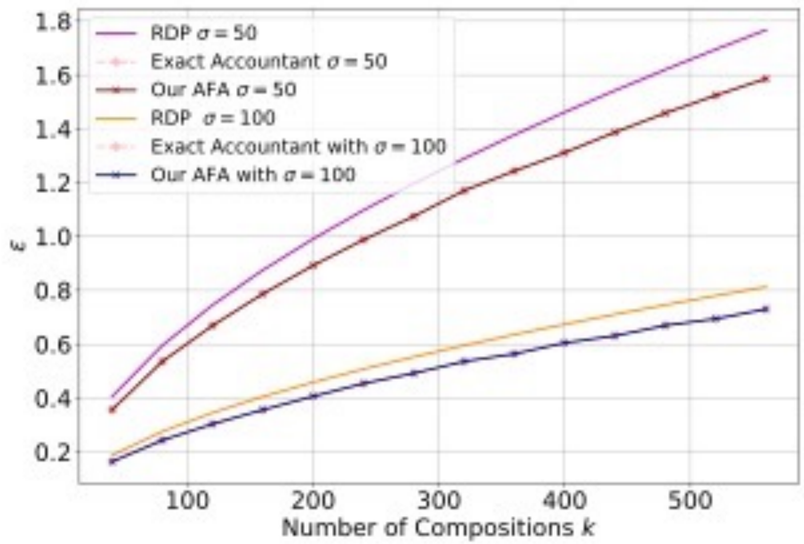
- Others that we know:
  - PureDP mechanisms are dominated by randomized response
  - ApproxDP mechanisms are dominated by leaky randomized response.
  - Exponential mechanism is dominated by two logistic distributions.
  - and so on …

- Research: expanding the list

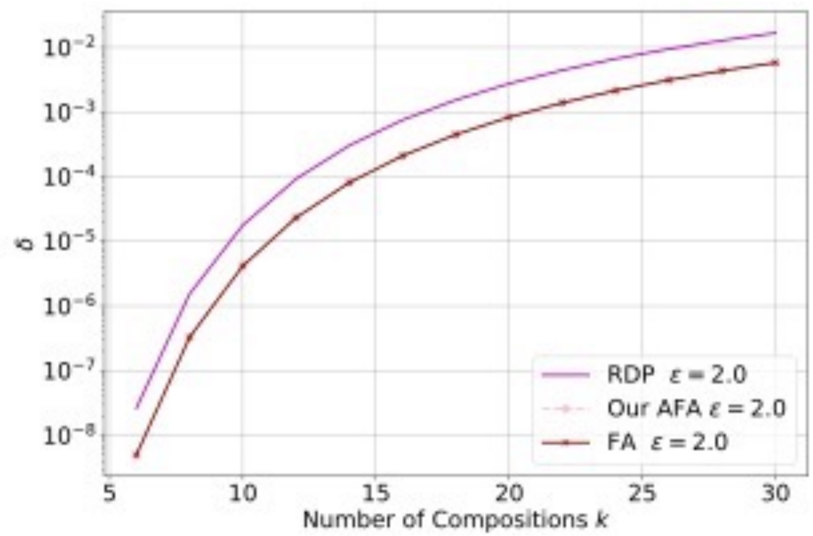
Connection between ϕ-function and other representations



It improves over RDP on the basic composition of building blocks.

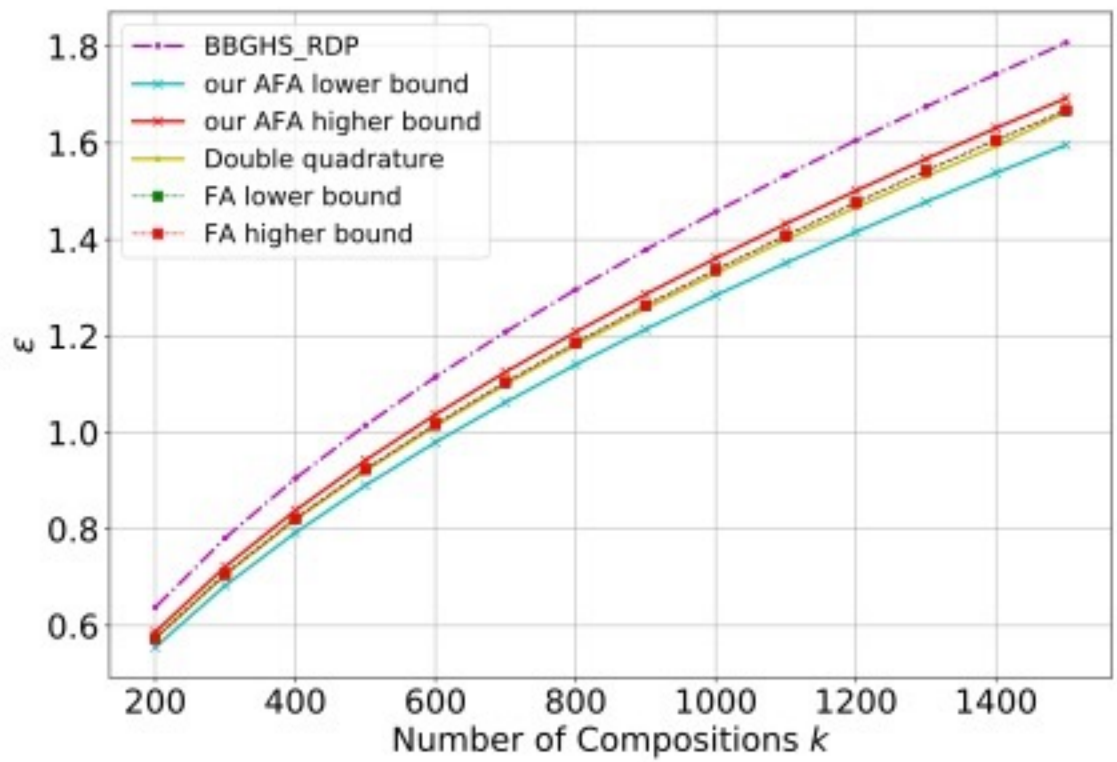


(a) Exp1 Gaussian mechanism



(b) Exp2 heterogeneous mechanisms

For sampled Gaussian, AFA (with quadrature methods) works like a charm.



(c) Exp3 Poisson Subsample

Our approach: error < 1e-14 with just 700 uneven spaced samples.
Koskela et al.: N = 1e5 evenly spaced points to obtain visually indistinguishable error.

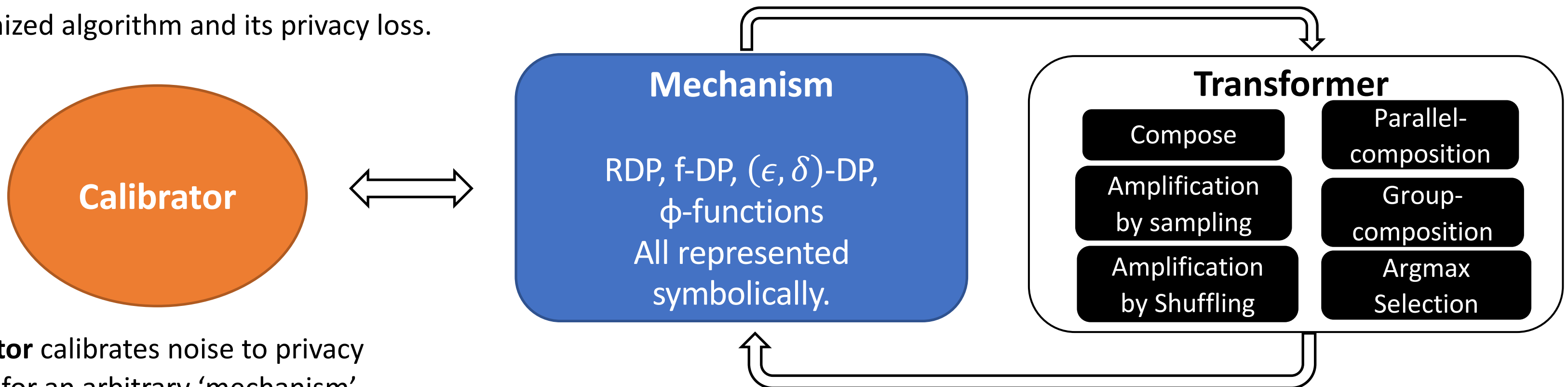
Outline of the talk

- Mechanism-specific privacy accounting
  - Application to Differentially Private Deep Learning
- Limitation of RDP and existing theory of PLD
- Main results:
  - Dominating pairs
  - Composition and amplification by sampling
  - Characteristic function representation
- Autodp -- a flexible tool for privacy accounting

autodp: a flexible and easy-to-use package for differential privacy

Mechanism is the base class that describes a randomized algorithm and its privacy loss.

Transformers manipulate functions (e.g., RDP) to create new **Mechanism**s.



Calibrator calibrates noise to privacy budget for an arbitrary 'mechanism'

1. You bring your mechanism.

2. Describe it in autodp as an Mechanism.

Then autodp takes care of

- Numerical computation of the privacy loss.
- Calibrating noise to privacy requirements.

Open source project:
https://github.com/yuxiangw/autodp

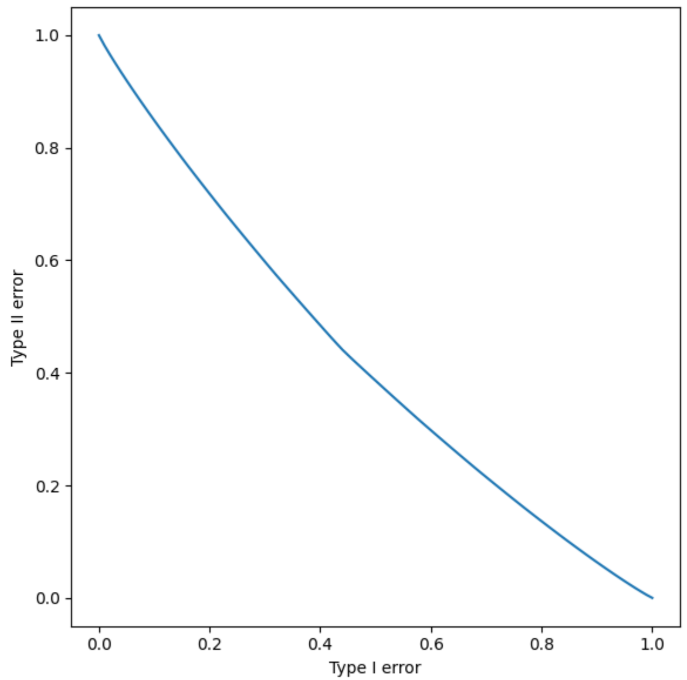
pip install autodp

Example autodp code: NoisySGD

```
from autodp.mechanism_zoo import GaussianMechanism
from autodp.transformer_zoo import AmplificationBySampling, Composition

subsample = AmplificationBySampling()
# by default this is using poisson sampling
mech = GaussianMechanism(sigma=5.0)
prob = 0.01

# Create subsampled Gaussian mechanism
# Gaussian mechanism qualifies for the tight bound
SubsampledGaussian_mech = subsample(mech,prob,improved_bound_flag=True)

# Now run this for 10000 iterations
compose = Composition()
noisysgd = compose([SubsampledGaussian_mech],[10000])
```

```python
import matplotlib.pyplot as plt

# Query for eps given delta

delta1 = 1e-6
eps1 = noisysgd.get_approxDP(delta1)
delta2 = 1e-4
eps2 = noisysgd.get_approxDP(delta2)
# Get name of the composed object, a structured
description of the mechanism generated automatically

print('Mechanism name is \"', noisysgd.name,'\"')
print('Parameters are: ',noisysgd.params)
print('epsilon(delta) = ', eps1, ', at delta = ', delta1)
print('epsilon(delta) = ', eps2, ', at delta = ', delta2)
# Get hypothesis testing interpretation so we can
directly plot it

fpr_list, fnr_list = noisysgd.plot_fDP()
plt.figure(figsize = (6,6))
plt.plot(fpr_list,fnr_list)
plt.xlabel('Type I error')
plt.ylabel('Type II error')
plt.show()
```

```
stdout:

Mechanism name is " Compose:{PoissonSample:Gaussian: 10000} "
Parameters are:  {'PoissonSample:Gaussian:sigma': 5.0,
'PoissonSample:Gaussian:PoissonSample': 0.01}
epsilon(delta) =  0.9141312880070975 , at delta =  1e-06
epsilon(delta) =  0.6843277003243384 , at delta =  0.0001
Process finished with exit code 0
```



Comparing to other DP open source library, you should use autodp

- Autodp decouples privacy accounting and DP mechanism implementation
  - A lot of research built into a simple straightforward API

- Autodp is the most flexible and among the tightest and easiest to use.
  - Very suitable for researchers developing new DP algorithms.
  - By default using RDP (mechanism specific analysis) for everything
  - Experimental support for Analytical Fourier Account

Take-home messages

- Compose mechanisms, not their privacy guarantee
- Dominating pairs fixes PLD formalism. If you want approx-DP in the end, you can retire RDP.
- Represent PLD using characteristic functions.
- Write your next DP paper with autodp!

Thank you for your attention!

Yuqing Zhu, Jinshuo Dong and W. (2021) "Optimal Accounting of Differential Privacy via Characteristic Function". AISTATS'2022: https://arxiv.org/abs/2106.08567

autodp: a flexible and easy-to-use package for differential privacy https://github.com/yuxiangw/autodp